Université Jean Monnet
Saint-Étienne

Habilitation à Diriger des Recherches
de l'Université Jean Monnet de St-Étienne

DSPT 9 : Sciences et technologies de l'information et de la communication

---

AVANCÉES EN THÉORIE PAC-BAYÉSIENNE
DE BORNES EN GÉNÉRALISATION À DES ALGORITHMES D'APPRENTISSAGE SUPERVISÉ ET DE TRANSFERT

Emilie MORVANT

Soutenue publiquement le 7 avril 2025

Devant le jury composé de

Rapporteurs	Marianne CLAUSEL	Professeure	Université de Lorraine
	Colin DE LA HIGUERA	Professeur	Université de Nantes
	Liva RALAIVOLA	VP Recherche	Criteo AI Lab
		Professeur	Aix-Marseille Université
Examinateurs	Stéphane CHRÉTIEN	Professeur	Université Lyon 2
	Rémi EMONET	MCF HDR	Université Jean Monnet
Tuteur	François JACQUENET	Professeur	Université Jean Monnet

Préparée au Laboratoire Hubert Curien - Équipe Data Intelligence
UMR CNSR 5516 CNRS - Université de Jean Monnet de St-Étienne - Institut d'Optique Graduate School

À mes élèves et doctorants, passés, présents et futurs,
qui, par leur soif d'apprendre et leur quête de savoir,
ont contribué à façonner mon parcours autant que j'ai cherché à guider le leur.

À mon père, mon Maître et mes enseignants,
qui, par leur générosité dans la transmission
ont marqué ma vie et nourri ma passion pour le partage.

≪ The one who sowed may sleep but the seeds should not rest. ≫

Sur la tombe du Grand Maître R. Shekhar (Manchuria Kung Fu)

Remerciements

Je tiens tout d'abord à adresser mes sincères remerciements aux membres de mon jury. Merci à vous d'avoir consacré du temps—une ressource précieuse dans notre métier—à l'évaluation de mon travail.

Merci à Liva Ralaivola, que j'ai eu la chance de côtoyer durant ma thèse. Sa générosité, ses conseils et sa présence ont été d'un grand soutien tout au long de ces trois années. C'est grâce à lui que j'ai découvert le monde du PAC-Bayes avec son fameux défi (réussi) : "Écrire un papier ICML en une semaine !". C'est également lui qui m'a présenté François Laviolette avec qui j'ai eu la chance de collaborer et auprès de qui j'ai passé un mois de recherche intense au GRAAL à Québec.

Merci à Marianne Clausel d'avoir accepté de consacrer du temps à mon HDR (connaissant les nombreuses sollicitations qu'implique le fait d'être une femme dans notre domaine). J'admire la pertinence de nos échanges, ainsi que sa bonne humeur scientifique contagieuse.

Merci à Colin De La Higuera, dont j'ai croisé la route pour la première fois sur les bancs de la faculté des sciences et techniques de l'université Jean Monnet, et qui fait partie des enseignants ayant marqué ma vie d'étudiante. Je suis sincèrement touchée qu'il ait accepté d'être rapporteur de mon HDR.

Merci à Stéphane Chrétien d'avoir accepté de présider ce jury. J'espère que "*APosTeriori*" nous aurons l'occasion de collaborer.

Merci à Rémi Emonet, avec qui je partage un bureau et une grande passion pour le rangement désorganisé. Je lui suis profondément reconnaissante, non seulement pour la richesse de nos échanges scientifiques et pédagogiques—toujours empreints de questions pertinentes et stimulantes—, mais aussi pour sa constante bienveillance à mon égard.

Enfin, merci à François Jacquenet, mon tuteur durant la préparation de cette HDR, mais bien plus que cela. Depuis ma Licence 1 Maths-Info à l'UJM en 2003 jusqu'à aujourd'hui, il a toujours su me prodiguer de précieux conseils et être une oreille très attentive. Sans lui, sans son soutien et sa bienveillance, mon parcours aurait été complètement différent. Je suis sincèrement honorée de l'avoir eu comme tuteur.

Je souhaite profiter de ce manuscrit pour remercier Florence Garrelie, directrice du Laboratoire Hubert Curien, pour sa bienveillance et son écoute. Je remercie également tous les membres de l'administration du LabHC qui font tourner (souvent dans l'ombre) le labo.

Merci à toutes les personnes qui, de près ou de loin, ont influencé mon parcours professionnel, que ce soit au sein de l'équipe Data Intelligence du LabHC ou du département informatique de la FST. Je pense tout particulièrement à Amaury Habrard, mon directeur de thèse, que j'ai eu la chance de retrouver à Saint-Étienne dès mon recrutement. C'est toujours un plaisir de pouvoir échanger scientifiquement avec lui.

Merci également à Stéphane Ayache, mon co-encadrant de thèse, même si nos chemins se croisent plus rarement désormais.

Une HDR ne se fait pas seule : les travaux présentés dans ce manuscrit sont le fruit d'échanges et de collaborations scientifiques ! Je remercie donc tous mes collaborateurs, en commençant par mes doctorants, qui sont au cœur de la plupart des avancées : Anil Goyal, Léo Gautheron, Paul Viallard, Hind Atbir et Julien Bastian. Ils m'ont toutes et tous permis, et me permettent encore, de grandir et d'avancer.

Un merci tout particulier à Paul, *a.k.a.* Polinou, avec qui j'ai partagé énormément pendant nos heures de travail—y compris plusieurs mois en distanciel durant le confinement—, mais

aussi en mode Manchuria Kung Fu. Je suis très fière de son parcours aujourd'hui (même si ce n'est que le début), et j'espère que nous continuerons à collaborer encore longtemps.

Merci à Pascal Germain, mon collaborateur "historique", avec qui j'ai construit le projet APRIORI. Et merci à tous les autres dont Aurélien Bellet, Cécile Capponi, Christine Largeron, Christoph Lampert, Farah Cherfaoui, Guillaume Metzler, Jean-Francis Roy, Jordan Patracone, Marc Sebban, Massih-Reza Amini, Mario Marchand, Rémi Eyraud et Valentina Zantedeschi.

En souvenir de ma thèse et de mes premiers pas dans le monde de la recherche, merci à l'équipe Qarma de Marseille.

En tant que Maître de Conférence, l'enseignement est très important pour moi. Je commence par une petite dédicace à des collègues du département informatique de la FST avec qui je partage diverses préoccupations et responsabilités en Licence : Émilie Samuel, Marc Bernard et Mathias Géry. Je remercie également l'ensemble de la scolarité de la FST pour l'accompagnement fournit !

Et, bien évidemment, je remercie tous ceux qui ont contribué à alimenter ma passion du partage et qui m'ont offert de belles opportunités pour enseigner très tôt, en commençant par mon père, Jean-Louis Morvant, puis une prof de maths d'exception, Chrystine Boureille, et plus tard mon Maître de Manchuria Kung Fu, Master Mathieu Derosière auprès de qui j'ai l'honneur d'apprendre encore chaque jour. Je souhaite lui exprimer toute ma gratitude et mon respect, il incarne pour moi la valeur essentielle du partage : celle qui passe par l'exemple, l'écoute et l'humilité de continuer à apprendre, toujours. Au passage, merci évidemment à toute la MKF Family (avec une pensée spéciale pour Aurélie et Neela) ! Oss.

Merci à Damien et Anne-Marlène. Enfin, merci à toute ma famille dont Michmich, les Torrus, les rFomages, la team à Lulu, les Grats, Jacqueline & Xavier, et à Cécile, ma plus grande supportrice dans tous les sens du terme (j'admire son courage, ah ah).

Une belle pensée pour mon père, JLM.

"Côté hublot dans le vaisseau
J'espère que t'es fier quand j'écris ces mots
Yes, papa, on s'captera tout là-haut
Cesse tes idées noires, poto
Stress donc retour au labo
Tu pars en quête comme commando Cousteau"

Table des matières

Première partie

Préliminaires

Curriculum Vitæ synthétique et résumé des contributions

1

1.1 Identification et CV Synthétique

Nom : Morvant **Prénom :** Emilie **Née le :** 12/03/1985 **À :** St-Étienne, France

Grade : MCF CN **Section CNU :** 27 **Page web :** `https://emorvant.github.io`

Établissement d'affectation : Université Jean Monnet de St-Étienne (UJM)

Composante d'affectation : Faculté des Sciences et Techniques

Unité de recherche : Laboratoire Hubert Curien, UMR CNRS 5516

Formation et Diplômes

— **2013** : **Doctorat en Informatique/Apprentissage automatique** **(mention très honorable)**
Titre : Apprentissage de vote de majorité pour la classification supervisée et l'adaptation de domaine : approches PAC-Bayésienne et combinaison de similarités
Encadrement : Amaury Habrard et Stéphane Ayache
Lieu : Univ. Aix-Marseille, Laboratoire d'Informatique Fondamentale, équipe Qarma
Prix de thèse d'Aix-Marseille Univ. 2013 & Accessit au prix de thèse en IA 2014 (délivré par l'AFIA)

— **2010** : **Master Recherche** Informatique Fondamentale — Univ. Aix-Marseille (mention bien)
Spécialité : apprentissage automatique et fouille de données

— **2008** : **Licence d'Informatique** — Université Jean Monnet (mention assez bien)

Parcours Professionnel

— **depuis 2014** : **Maître de conférences** (temps partiel 80% depuis 2022) à l'UJM et au Laboratoire Hubert Curien, UMR CNRS 5516, thématique **Data Intelligence**, équipe **Machine Learning**
Depuis 2020 : Bénéficiaire de la PEDR, puis de la RIPEC 3

— **2013-2014** : **Chercheur Post-doctoral** à l'*Institute of Science and Technology Austria*, Klosterneuburg, Autriche, dans le cadre du projet *ERC Starting Grant "Lifelong Learning of Visual Scene Understanding"* (thèmes : apprentissage automatique & vision par ordinateur)

— **2010-2013** : **Thèse** en apprentissage automatique dans l'équipe Qarma du Laboratoire d'Informatique Fondamentale de Marseille, financée par le projet ANR Videosense, avec un monitorat de 2011 à 2013.

— **2010** : **Stage de M2 Recherche** en apprentissage automatique dans l'équipe Qarma du Laboratoire d'Informatique Fondamentale de Marseille

1.2 Activité pédagogique

J'enseigne l'informatique depuis 2008 (année de mon M1). J'ai effectué divers enseignements d'informatique en post-bac, d'abord en tant que vacataire en prépa CPGE ECE au lycée St-Louis à St-Étienne, à l'université d'Aix-Marseille (vacations puis monitorat), à l'Institute of Science and Technology Austria (module doctoral) puis à l'UJM suite à mon recrutement. Je suis également **responsable la Licence 2 d'informatique depuis 2015 et responsable de la Licence 1 Informatique depuis 2024** à l'UJM.

En fait, mon goût pour de l'enseignement est même bien plus ancien. En 2000, j'ai commencé à animer des cours d'initiation à la communication et à l'audiovisuel pour des collégiens de l'établissement St-Louis à St-Étienne.

1.2.1 Principaux enseignements

Je réalise chaque année entre 250 et 300 heures par an, toutes en informatique à la Faculté des Sciences et Techniques (FST) de l'UJM à destination d'étudiants de niveau Licence (L1/L2) et de niveau Master (à noter que je n'interviens plus au niveau Master). La majorité de mes enseignements gravitent autour de la thématique "**programmation et algorithmique**". Je suis également intervenue en niveau Master pour initier les étudiants à la méthodologie de la recherche et à l'intelligence artificielle. J'encadre chaque année des stages de M1 et de M2 (à finalité recherche). Une des particularités de mon service est qu'il s'effectue principalement au niveau de la **L2 informatique dont je suis la responsable**. Cela me permet d'avoir **un suivi quasi quotidien des étudiants** et de leurs difficultés générales pour les accompagner au mieux non seulement dans les enseignements que j'effectue, mais également dans leur parcours universitaire (l'année de L2 est souvent cruciale pour la suite de leurs études et leur orientation professionnelle).

1.2.2 Pratiques pédagogiques

D'un point de vue général, en plus des exercices sur feuille, qui ont l'intérêt d'apprendre aux étudiants à formaliser leurs réflexions, je trouve crucial et important de leur proposer des mises en applications pratiques des notions étudiées en cours/TD.

Un premier exemple naturel est la programmation où la **pratique** sur machine est très importante. En plus des séances pratiques constituées d'exercices courts avec un objectif pédagogique identifié et clair, je mets en place des **projets de développement** avec un aspect plus ludique (ex. implémenter un jeu vidéo simple avec une interface graphique simple). Cependant, avec l'arrivée des agents conversationnels type ChatGPT, nous sommes confrontés à une évolution de la discipline informatique, dont notamment la programmation. Je travaille à une refonte progressive de mon approche pour évaluer les compétences pratiques en programmation des étudiants. Plus précisément, j'essaye dans un premier temps d'**accompagner les étudiants dans la valorisation de la plus-value qu'ils peuvent apporter face aux agents conversationnels**. Je les incite, en particulier, à toujours remettre en cause le code que j'écris au tableau, ce qui permet de stimuler leur esprit critique pour les aider non seulement à remettre en question leur propre code, mais également n'importe quel autre code fourni par une source extérieure. De plus, pour évaluer leurs compétences pratiques, je donne aujourd'hui une très forte importance à leur capacité d'analyse. Pour ce faire, je les accompagne dans l'élaboration d'un protocole expérimental afin de tester leur projet, puis j'évalue **leur capacité d'analyse et de critique** des résultats empiriques ob-

tenus ainsi que de leur code[1]. Dans un second temps, puisqu'ils seront très probablement amenés à utiliser ce genre d'agents conversationnels dans leur vie professionnelle, j'ai pour projet de mettre en place des exercices pour les sensibiliser à l'utilisation de ces agents. Une première piste concerne la mise en place d'exercices pour comparer les résultats générés par différents prompts, analyser ces résultats, les comprendre et évidemment les remettre en question.

Je tiens à mentionner ma forte implication, **durant la période du COVID-19**, pour assurer **la continuité pédagogique** durant la période de confinement. En effet, dès l'annonce du confinement, je me suis attelée à la mise en place de moyens de communication en distanciel avec les étudiants. Concrètement, j'ai mis en place des serveurs discord pour différentes promotions pour permettre non seulement l'échange entre les étudiants et les enseignants, mais également la possibilité de dispenser des cours en distanciel. Je me suis fortement impliquée dans **l'écoute et l'accompagnement des étudiants** durant cette période compliquée. Je me suis rendue disponible quasiment 24h/24 pour répondre à leurs difficultés scolaires et psychologiques ; je sais que cela a été beaucoup apprécié. Ce type de serveur est un outil généralement maîtrisé par les étudiants, ce qui facilite significativement l'échange et la communication : je continue aujourd'hui à utiliser cet outil pour certaines promotions en complément des outils plus classiques mis à disposition par l'université (type Moodle).

1.2.3 Mise en place d'un parcours en alternance

Je me suis impliquée dans la mise en place du parcours en alternance pour la Licence d'informatique à l'université Jean Monnet (dans ce parcours les étudiants suivent une partie des cours de la L2/L3 informatique "classique" et seule la L3 est effectuée en alternance). Ce parcours a ouvert en septembre 2020 et est très apprécié des étudiants qui souhaitent rejoindre le monde professionnel rapidement. Dans ce contexte, j'ai créé une UE spécifique à la Licence 2 d'informatique parcours alternance intitulée "Programmation spécifique alternant". Cette UE, majoritairement pratique, a pour objectifs *(i)* d'approfondir les compétences en programmation des étudiants qui réaliseront leur L3 en alternance, *(ii)* de commencer à les familiariser avec les bonnes pratiques à suivre en entreprise, pour faciliter leur insertion professionnelle.

1.2.4 Responsabilités pédagogiques

— Depuis 2015, je suis **responsable de la Licence 2** informatique (effectif entre 50 et 60 étudiants par an, dans un contexte difficile où nous ne pouvons pas dédoubler les TD, faute de ressources humaines). Les tâches liées à cette responsabilité sont les suivantes :
  — Échanges permanents avec les étudiants
  — Échanges avec l'équipe pédagogique et les responsables administratifs
  — Animation de plusieurs réunions durant l'année
  — Membre de la commission de recrutement L2/L3 sur e-candidat
  — Présidente du jury de la L2 pour les parcours classiques, alternants et LAS
  — Élaboration de l'emploi du temps

1. Par exemple, un projet que je propose régulièrement consiste à développer un petit logiciel de compression de données en utilisant le codage de Huffman. Lors de la présentation de leur projet, ils doivent alors me fournir une analyse empirique du taux de compression obtenu par leur programme.

— depuis 2024, je suis **responsable de la Licence 1** informatique. Les tâches liées à cette responsabilité sont moindres qu'en L2 et se résument principalement à être l'interlocutrice privilégiée entre les responsables administratifs de la Licence 1 (commune aux maths, à la physique et à la chimie) et le département informatique de la FST, à l'interaction avec les étudiants concernant le parcours informatique et à être membre du Jury de la L1.

— Membre des jurys L3 et Licence d'informatique (classique, parcours alternance et LAS)

— Je suis **actuellement responsable de 4 UEs** : 2 UEs de Programmation Impérative (au S3 et S4), UE Programmation spécifique alternants, UE Systèmes d'exploitation

— **Membre du conseil de perfectionnement** de la Licence d'informatique

1.3 Activité scientifique

1.3.1 Productions scientifiques et publications

Cette section fait état de mes publications depuis l'obtention de mon doctorat en 2013. La liste détaillée des publications, ainsi que les codes sources associés mis à disposition de la communauté scientifique, sont disponibles sur mon site : emorvant.github.io

Livre : 1

[L1] Ievgen Redko ; E. Morvant ; Amaury Habrard ; Marc Sebban ; Younès Bennani. **Domain Adaptation Theory : Available Theoretical Results**. *ISTE Press-Elsevier*, 2019, ISBN : 9781785482366

Articles dans des revues internationales avec comité de lecture : 7

[J1] Paul Viallard ; Pascal Germain ; Amaury Habrard ; E. Morvant. **A General Framework for the Practical Disintegration of PAC-Bayesian Bounds**, *Machine Learning Journal (MLJ)*, 113 :519–604, 2024, DOI : 10.1007/s10994-023-06391-0

[J2] Léo Gautheron ; Amaury Habrard ; E. Morvant ; Marc Sebban. **Metric Learning from Imbalanced Data with Generalization Guarantees**, *Pattern Recognition Letters (PRL)*, 133 :298-304, 2020, DOI : 10.1016/j.patrec.2020.03.008

[J3] Pascal Germain ; Amaury Habrard ; François Laviolette ; E. Morvant. **PAC-Bayes and Domain Adaptation**. *Neurocomputing*, 379 :379-397, 2020, DOI : 10.1016/j.neucom.2019.10.105

[J4] Anil Goyal ; E. Morvant ; Pascal Germain ; Massih-Reza Amini. **Multiview Boosting by Controlling the Diversity and the Accuracy of View-specific Voters**. *Neurocomputing*, 358 :81-92, 2019, DOI : 10.1016/j.neucom.2019.04.072

[J5] François Laviolette ; E. Morvant ; Liva Ralaivola ; Jean-Francis Roy. **Risk Upper Bounds for General Ensemble Methods with an application to Multiclass Classification**. *Neurocomputing*, 219 :15-25, 2017, DOI : 10.1016/j.neucom.2016.09.016

[J6] E. Morvant. **Domain Adaptation of Weighted Majority Votes via Perturbed Variation-Based Self-Labeling**, *Pattern Recognition Letters (PRL)*, 51(0) :37–43, 2015, DOI : 10.1016/j.patrec.2014.08.013

[J7] Aurélien Bellet ; Amaury Habrard ; E. Morvant ; Marc Sebban. **Learning A Priori Constrained Weighted Majority Votes**, *Machine Learning Journal (MLJ)*, 97(1-2) :129-154, 2014, DOI : 10.1007/s10994-014-5462-z

Articles de conférences internationales à comité de lecture : 12

[C1] Jordan Patracone ; Paul Viallard ; E. Morvant ; Gilles Gasso ; Amaury Habrard ; Stéphane Canu. **A Theoretically Grounded Extension of Universal Attacks from the Attacker's Viewpoint**. *European Conference on Machine Learning & Principles and Pratice of Knowledge Discovery in Databases (ECML-PKDD)*, 2024

[C2] Paul Viallard ; Rémi Emonet ; Amaury Habrard ; E. Morvant ; Valentina Zantedeschi. **Leveraging PAC-Bayes Theory and Gibbs Distributions for Generalization Bounds with Complexity Measures** *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2024

[C3] Valentina Zantedeschi ; Paul Viallard ; E. Morvant ; Rémi Emonet ; Amaury Habrard ; Pascal Germain ; Benjamin Guedj. **Learning Stochastic Majority Votes by Minimizing a PAC-Bayes Generalization Bound** *Advances on Neural Information Processing Systems (NeurIPS)*, 2021

[C4] Guillaume Vidot ; Paul Viallard ; Amaury Habrard ; E. Morvant. **A PAC-Bayes Analysis of Adversarial Robustness**. *Advances on Neural Information Processing Systems (NeurIPS)*, 2021

[C5] Paul Viallard ; Pascal Germain ; Amaury Habrard ; E. Morvant. **Self-Bounding Majority Vote Learning Algorithms by the Direct Minimization of a Tight PAC-Bayesian C-Bound** *European Conference on Machine Learning & Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD)*, 2021

[C6] Léo Gautheron ; Pascal Germain ; Amaury Habrard ; Guillaume Metzler ; E. Morvant ; Marc Sebban ; Valentina Zantedeschi. **Landmark-based Ensemble Learning with Random Fourier Features and Gradient Boosting**. *European Conference on Machine Learning & Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD)*, 2020

[C7] Gaël Letarte ; E. Morvant ; Pascal Germain. **Pseudo-Bayesian Learning with Kernel Fourier Transform as Prior**. *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2019

[C8] Léo Gautheron ; Amaury Habrard ; E. Morvant ; Marc Sebban. **Metric Learning from Imbalanced Data**. *The IEEE International Conference on Tools with Artificial Intelligence (ICTAI)*, 2019

[C9] Anil Goyal ; E. Morvant ; Massih-Reza Amini. **Multiview Learning of Weighted Majority Vote by Bregman Divergence Minimization**. *International Symposium on Intelligent Data Analysis (IDA)*, 2018

[C10] Anil Goyal ; E. Morvant ; Pascal Germain ; Massih-Reza Amini. **PAC-Bayesian Analysis for a two-step Hierarchical Mutliview Learning Approach**. *European Conference on Machine Learning & Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD)*, 2017

[C11] Pascal Germain ; Amaury Habrard ; François Laviolette ; E. Morvant. **A New PAC-Bayesian Perspective on Domain Adaptation**. *International Conference on Machine Learning (ICML)*, 2016

[C12] Mario Marchand ; Su Hongyu ; E. Morvant ; Juho Rousu ; John Shawe-Taylor. **Multilabel Structured Output Learning with Random Spanning Trees of Max-Margin Markov Networks**. *Advances on Neural Information Processing Systems (NeurIPS)*, 2014

Articles dans des workshops internationaux avec comité de lecture : 5

[W1] Paul Viallard ; Rémi Emonet ; Pascal Germain ; Amaury Habrard ; E. Morvant. **Interpreting Neural Networks as Majority Votes through the PAC-Bayesian Theory**. *NeurIPS 2019 Workshop on Machine Learning with guarantees*, 2019

[W2] Pascal Germain ; François Laviolette ; Amaury Habrard ; E. Morvant. **A New PAC-Bayesian View of Domain Adaptation**. *NeurIPS Workshop on Transfer and Multi-Task Learning : Trends and New Perspectives*, 2015

[W3] E. Morvant ; Amaury Habrard ; Stéphane Ayache. **Majority Vote of Diverse Classifiers for Late Fusion**. *IAPR International Workshops on Statistical Techniques in Pattern Recognition & Structural and Syntactic Pattern Recognition (S+SSPR)*, 2014

[W4] François Laviolette ; E. Morvant ; Liva Ralaivola ; Jean-Francis Roy. **On Generalizing the C-Bound to the Multiclass and Multi-label Settings**. *NeurIPS Workshop on Representation and Learning Methods for Complex Outputs*, 2014

[W5] Pascal Germain ; François Laviolette ; Amaury Habrard ; E. Morvant. **An Improvement to the Domain Adaptation Bound in a PAC-Bayesian context**. *NeurIPS Workshop on Transfer and Multi-task learning : Theory Meets Practice*, 2014

Articles dans des conférences nationales avec comité de lecture : 14

[N1] Hind Atbir ; Farah Cherfaoui ; Guillaume Metzler ; E. Morvant ; Paul Viallard. **Une borne PAC-Bayésienne sur une mesure de risque pour l'apprentissage équitable**, *Conférence Francophone sur l'Apprentissage Automatique (CAp)*, 2024

[N2] Paul Viallard ; Rémi Emonet ; Pascal Germain ; Amaury Habrard ; E. Morvant ; Valentina Zantedeschi. **Intérêt des bornes désintégrées pour la généralisation avec des mesures de complexité**. *CAp*, 2022

[N3] Valentina Zantedeschi ; Paul Viallard ; E. Morvant ; Rémi Emonet ; Amaury Habrard ; Pascal Germain ; Benjamin Guedj. **Learning Stochastic Majority Votes by Minimizing a PAC-Bayes Generalization Bound**. *CAp*, 2022

[N4] Paul Viallard ; E. Morvant ; Pascal Germain. **Apprentissage de Vote de Majorité par Minimisation d'une C-Borne**. *CAp*, 2021

[N5] Paul Viallard ; E. Morvant ; Pascal Germain. **Dérandomisation des Bornes PAC-Bayésiennes**. *CAp* 2021

[N6] Paul Viallard ; Guillaume Vidot ; E. Morvant. **Une Analyse PAC-Bayésienne de la Robustesse Adversariale**. *CAp* 2021

[N7] Paul Viallard ; Rémi Emonet ; Amaury Habrard ; E. Morvant ; Pascal Germain. **Théorie PAC-Bayésienne pour l'apprentissage en deux étapes de réseaux de neurones**. *CAp* 2020

[N8] Léo Gautheron ; Pascal Germain ; Amaury Habrard ; Guillaume Metzler ; E. Morvant ; Marc Sebban ; Valentina Zantedeschi. **Apprentissage d'ensemble basé sur des points de repère avec des caractéristiques de Fourier aléatoires et un renforcement du gradient**. *CAp* 2020

[N9] Léo Gautheron ; Pascal Germain, Amaury Habrard ; Gaël Letarte ; E. Morvant ; Marc Sebban ; Valentina Zantedeschi. **Revisite des "random Fourier features" basée sur l'apprentissage PAC-Bayésien via des points d'intérêts**. *CAp* 2019

[N10] Anil Goyal ; E. Morvant ; Massih-Reza Amini. **Apprentissage d'un vote de majorité hiérarchique pour l'apprentissage multi-vues**. *CAp* 2018

[N11] Léo Gautheron ; Amaury Habrard ; E. Morvant ; Marc Sebban. **Apprentissage de métrique pour la classification supervisée de données déséquilibrées**. *CAp* 2018

[N12] Anil Goyal, E. Morvant, Pascal Germain. **Une borne PAC-Bayésienne en espérance et son extension à l'apprentissage multi-vues**. *CAp* 2017

[N13] Anil Goyal ; E. Morvant ; Pascal Germain ; Massih-Reza Amini. **Théorèmes PAC-Bayésiens pour l'apprentissage multi-vues**. *CAp* 2016

[N14] E. Morvant. **Adaptation de domaine de vote de majorité par auto-étiquetage non itératif**. *CAp* 2014

Actes de conférence : 1

[P1] Vladimir Kolmogorov ; Christoph Lampert ; E. Morvant ; Rustem Takhanov. **Proceedings of the Annual Workshop of the Austrian Association for Pattern Recognition**, 2014

1.3.2 Encadrement doctoral et scientifique

1.3.2.1 Encadrements doctoraux

— Depuis 2024 : **Julien Bastian**
Titre : **Fair multi-view learning - Theory and algorithms**
Co-encadrement : Christine Largeron (LabHC), Guillaume Metzler (ERIC, Lyon 2)

— Depuis 2024 : **Hind Atbir**
Titre : **Learning fair & robust kernel-based models with generalization bounds**
Co-encadrement : Rémi Eyraud, Farah Cherfaoui (LabHC), Paul Viallard (Inria Rennes)

— 2019-2022 : **Paul Viallard**
Titre : **Beyond PAC-Bayesian Bounds : From Disintegration to Novel Bounds**
Co-encadrement : Amaury Habrard (LabHC), Pascal Germain (GRAAL, Québec)
(actuellement chercheur ISFP à l'Inria Rennes)

— 2017-2020 : **Léo Gautheron**
Titre : **Learning Tailored Data Representations from Few Labeled Examples**
Co-encadrement : Amaury Habrard, Marc Sebban (LabHC)
(actuellement ingénieur en sciences des données chez Synapse Défense)

— 2015-2018 : **Anil Goyal**
Titre : **Learning a Multiview Weighted Majority Vote Classifier : Using PAC-Bayesian Theory and Boosting**
Co-encadrement : Massih-Reza Amini (LIG, Grenoble)
(actuellement *Senior Research Scientist* chez Amazon)

1.3.2.2 Encadrements post-doctoraux

— 2022-2024 : **Marie-Ange Lèbre** (33%) — co-encadrée avec Amaury Habrard et Rémi Emonet (LabHC) sur l'apprentissage profond pour la détection et classification de microorganismes, financé par le projet ScanBioM en partenariat avec BioMérieux

1.3.2.3 Encadrements de stages de recherche, niveau Master

— Baptiste Mathevon (M1) : PAC-Bayesian Certifications for Multi-Armed Bandits
— Hind Atbir (M2) : PAC-Bayesian Fair Learning
— Julien Bastian (M2) : PAC-Bayesian Fairness Through Domain Generalization
— Mickaël Gault (M1) : Study of the relations between Domain Adaptation & Fairness
— Julien Bastian (M1) : Random Fourier Features, PAC-Bayes and Domain Adaptation
— Alexiane Fraisse (M1) : Random Fourier Features and Domain Adaptation
— Luiza Dzhidzhavadze (M1) : A Multiclass C-Bound-Based Algorithm
— Himanshu Pandey (M1) : A Multiclass C-Bound-Based Algorithm

— Paul Viallard (M2) : Deep Learning and PAC-Bayes
— Omar El-Sabrout (M1) : Active Learning for PAC-Bayesian Domain Adaptation
— Loujain Liekah (M1) : Experts Combination
— Luc Giffon (M2) : Efficient anomaly detection in data stream
— Arunava Maulik (M1) : Experts Combination
— Prem Prakash (M1) : Boosting and C-bound
— Léo Gautheron (M1) : Improving the bibliometry platform Labmetry
— Benjamin Sabot (M1) : Study of the C-bound as stopping criterion for neural networks
— Soroush Seifi (M1) : A PAC-Bayesian Multiview Study

1.3.3 Animation scientifique et investissement international

1.3.3.1 Invitations

— 2019 : **Conférencière invitée** lors des Journées de Statistique, Nancy, France.
Titre : When PAC-Bayesian Majority Vote meets Domain Adaptation
— 2018 : **Séminaire invité** à l'INRIA Lille - Nord Europe, France.
Titre : When PAC-Bayesian Majority Vote Meets Transfer Learning
— 2016 : **Séminaire invité**, LIVES workshop à Aix*Marseille Univ.
Titre : PAC-Bayesian Majority Vote & Domain Adaptation
— 2014 : **Séminaire invité** au LIG, Grenoble, France.
Titre : When PAC-Bayes meets Domain Adaptation
— 2014 : **Séminaire invité** au Laboratoire Hubert Curien, UJM
Titre : Domain Adaptation of Majority Votes via PV-based Label transfer

1.3.3.2 Implication dans des projets de recherche

Depuis la fin de ma thèse j'ai été impliquée dans les projets suivants :

— **Depuis 2024** : **Coordinatrice locale** du projet ANR FAMOUS
Fair Multi-modal Learning
En collaboration avec les laboratoires LIS, INT, LITIS et l'entreprise Euranova
— **2022-2024** : **Membre** du projet ScanBioM (dans le cadre de l'AAP "Grand defi biomédicament")
En collaboration avec l'entreprise Biomérieux.
— **Depuis 2021** : **Membre** du projet ANR TAUDOS
Theory and algorithms for the understanding of deep learning on sequential data
En collaboration avec les laboratoires LIS, MILA et l'entreprise Euranova
— **2019-2023** : **Porteuse et Coordinatrice** du projet ANR APRIORI
A PAC-Bayesian representation learning perspective
En collaboration avec Inria Lille - Nord Europe.
— **2016-2019** : **Membre** du projet ANR LIVES
Learning with interacting viewS
En collaboration avec les laboratoires LIS, INT, LIP6.

— **2018** : **Porteuse** du projet JCJC INS2I-CNRS PaRaFF
PAC-Bayesian Random Fourier Features
En collaboration avec Pascal Germain (Inria Lille - Nord Europe).
Ce projet d'un an a permis d'initier la rédaction du projet ANR APRIORI.
— **2015-2018** : **Porteuse** d'un projet région financé par le dispositif ARC
Financement de la thèse d'Anil Goyal
— **2013-2014** : **Membre** du projet ERC Starting Grant L3VISU à l'ISTA dont le porteur était Christoph Lampert et qui a financé mon postdoc.

1.3.3.3 Animation internationale et locale

— Participation régulière à des **comités de programmes** dans le cadre de conférences internationales (ex. ICML, ECML-PKDD, AISTATS, Neurips) et nationales (CAp)
— Participation à l'**organisation de deux conférences** (IDA 2015 et ÖAGM 2014)
— **Organisation d'un workshop** lors de la conférence ECML-PKDD 2014
— **Co-chair des démos** pour la conférence ECML-PKDD 2019
— **Publicity chair** de la conférence ECML-PKDD 2022
— **Experte externe** lors de l'évaluation de l'appel à candidature du programme postdoctoral "IST-BRIDGE [2] des *actions Marie Skłodowska-Curie* en 2022
— **Vice-présidente (2017 à 2020) et membre fondateur** de la Société Savante Francophone d'Apprentissage Machine (SFFAM, ssfam.org)
— **Membre du bureau (2018-2020) et membre fondateur** du groupe MAchine Learning et Intelligence Artificielle (MALIA) de la Société Française de Statistique
— Participation à **14 comités de sélection** dans le cadre de recrutements de Maître de Conférences en Informatique/Apprentissage automatique
— **Membre du CNU** en section 27 (2021-2022)
— En dehors des thèses que j'ai co-encadrée, j'ai été examinatrice pour le jury de thèse de Luxin Zhang et de celui de Tahar Allouche en 2022

1.3.3.4 Vulgarisation scientifique

— 2018 : **Les Universitaires retournent à l'École**, Lycée É. Mimard, St-Étienne
Titre : Apprentissage Automatique et Adaptation de Domaine
— 2018 : **Université pour tous**, Laboratoire Hubert Curien
Titre : Qu'est-ce que l'adaptation de domaine ?
— 2016 : **Visite d'étudiants au Laboratoire**, Laboratoire Hubert Curien
Titre : Qu'est-ce que l'adaptation de domaine ?

1.3.4 Implication dans la vie du Laboratoire

— **Membre du conseil du Laboratoire** Hubert Curien depuis 2015
— **Co-animatrice** (2017-2022) des comptes Twitter et Facebook du Laboratoire Hubert Curien : *Diffusion grand public et scientifique des faits marquants liés au laboratoire*
— **Sauveteur Secouriste du Travail** (titulaire du PSC1 depuis 2017).

2. Détails sur le programme IST-BRIDGE : `ista.ac.at/en/education/postdocs/ist-bridge`.

1.4 Résumé de mes contributions depuis ma thèse

Depuis ma thèse, mes travaux de recherche se placent dans le cadre l'apprentissage automatique *(machine learning)*, un sous domaine de l'intelligence artificielle. Ils s'articulent principalement autour de problématiques de l'**apprentissage statistique** : l'apprentissage supervisé, l'adaptation de domaine, l'apprentissage multi-vues (ou multimodal), l'apprentissage de représentation et, depuis plus récemment, l'apprentissage équitable et la robustesse. Bien que j'étudie ces problématiques d'un point de vue général, la majorité de mes contributions se basent sur la théorie PAC-Bayésienne que j'utilise pour dériver des garanties théoriques et de nouveaux algorithmes d'apprentissage. La Figure 1.1 représente de manière simplifiée le cadre général dans lequel se situe ma recherche. Dans la suite, je fais référence aux publications listées dans la Section 1.3.

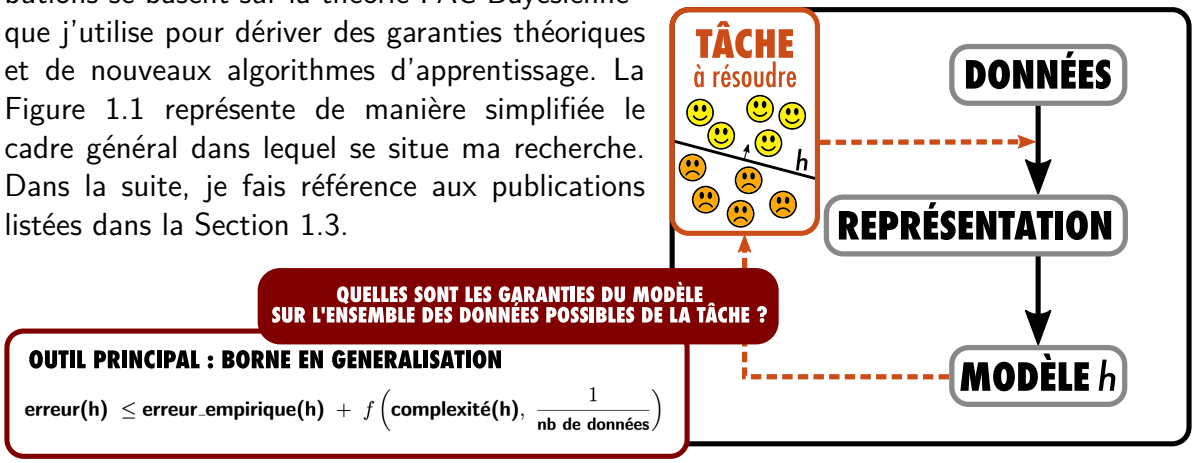


Figure 1.1. *Représentation schématique du cadre général dans lequel mes travaux se placent : l'apprentissage automatique. Étant donné une tâche à résoudre pour laquelle on dispose de données étiquetées, l'objectif est d'apprendre une modèle capable de répondre à la tâche sur des données non-observées. La question cœur de mes contributions est l'étude des garanties théoriques du modèle appris sur ces nouvelles données en utilisant des bornes en généralisation.*

1.4.1 Théorie PAC-Bayésienne

La **théorie PAC-Bayésienne**, introduite par Shawe-Taylor et Williamson (1997), est un outil issu de la théorie de l'apprentissage statistique pour dériver des bornes probabilistes (dîtes PAC pour *Probabilistic Approximately Correct*, Valiant (1984)) pour estimer, pour une tâche donnée, à quel point un modèle, appris à partir de données observées, est capable de généraliser sur de nouvelles données (pour la même tâche, *i.e.* des données tirées selon la même distribution de probabilité). Autrement dit, les garanties sont exprimées à l'aide de bornes probabilistes majorant l'espérance des erreurs commises par le modèle (appelée erreur en généralisation). Classiquement, ces bornes font intervenir un compromis entre les erreurs commises sur les données d'apprentissage (appelée erreur d'apprentissage) et une mesure de complexité de la famille de modèles considérée. L'idée sous-jacente de ce type de bornes est que pour apprendre un modèle performant, il faut que ce dernier soit capable de traiter correctement les données d'apprentissage tout en évitant le sur-apprentissage : pour cela, il faut limiter la complexité du modèle appris (la mesure de complexité a donc une influence forte sur le modèle appris). Les approches les plus classiques de la littérature telles que les bornes basées sur la dimension de Vapnik-Chervonenkis (Vapnik et Chervonenkis, 1968) ou la complexité de Rademacher (Bartlett et Mendelson, 2002) permettent d'obtenir des bornes "en pire cas" (*i.e.*, pour tous les modèles de la famille considérée).

La particularité de la théorie PAC-Bayésienne est que les bornes s'expriment en espérance sur la famille de modèles. Cela permet d'obtenir des garanties statistiques sur la qualité

de modèles pouvant être définis comme un **vote de majorité** pondéré[3] (autrement dit, une combinaison pondérée des modèles de la famille considérée), ou comme un modèle stochastique (souvent appelé classifieur de Gibbs en PAC-Bayes). Ces bornes ont l'avantage **(i)** d'être calculables à partir des données dont on dispose (les données d'apprentissage), **(ii)** d'être une source d'inspiration dans la conception d'algorithmes efficaces de minimisation des bornes, **(iii)** de pouvoir être directement optimisables pour obtenir des algorithmes auto-certifiés, et **(iv)** de pouvoir faire intervenir la diversité/complémentarité des fonctions combinées. En général, ces algorithmes construisent la combinaison de la manière suivante : étant donné une distribution (*i.e.*, des poids) *a priori* sur l'ensemble des fonctions intervenant dans la combinaison et un ensemble de données d'apprentissage, l'objectif est d'apprendre une distribution *a posteriori* (sur l'ensemble des fonctions) permettant de minimiser la borne, c'est-à-dire de minimiser l'espérance des erreurs commises par la combinaison. La majorité de mes contributions **[J3,J5,J6,J7,C2,C4,C7,C10, C11,W1,W2,W3,W4,W5]** se placent dans ce cadre et proposent des **analyses théoriques** et des **algorithmes** dans les problématiques résumées dans les Sections 1.4.2 à 1.4.6.

Il est important de préciser que la théorie PAC-Bayésienne a été peu étudiée jusqu'à l'avènement des techniques d'apprentissage profond au milieu des années 2010. En effet, bien que les bornes PAC-Bayésiennes s'expriment en termes d'espérance sur un ensemble de fonctions/modèles, elles permettent, en général, d'obtenir des bornes plus informatives et plus précises que les théories plus classiques telles que les bornes basées sur la dimension VC. Dans cette dynamique et dans le contexte de la thèse de Paul Viallard, nous avons obtenu, en plus des contributions citées ci-dessus, plusieurs avancées en théorie PAC-Bayésienne générale visant à s'affranchir de certains de ses défauts et à améliorer des bornes et algorithmes qui en découlent. Plus particulièrement, nous avons développé des **algorithmes auto-certifiés**[4] Freund, 1998 adaptés à l'apprentissage des votes de majorités **[C3,C5]**. Nous avons également proposé des méthodes pour s'affranchir de l'aspect stochastique des bornes PAC-Bayésiennes en proposant une nouvelle approche de **désintégration des bornes [J1]**. Cela a pour intérêt de permettre l'adaptation des résultats à des modèles plus généraux que les votes de majorité ou que les modèles stochastiques. Enfin, une contribution **[C2]** qui me semble majeure pour la communauté scientifique est la dérivation de bornes permettant d'incorporer une **mesure de complexité** définie par l'utilisateur. Cette dernière contribution ouvre de nombreuses pistes pour le développement, non seulement, de nouveau résultats théoriques en apprentissage automatique, mais également, de méthodes d'apprentissage.

1.4.2 Apprentissage de Représentation

Une étape clef, au cœur de la thèse de Léo Gautheron et du projet ANR APRIORI, qui détermine le succès de toute tâche d'apprentissage automatique (plus généralement en science des données) est la construction d'une représentation des données permettant de faciliter leur traitement. En effet, si cette représentation n'est pas suffisamment pertinente pour une tâche particulière, cette tâche ne pourra pas être résolue. Jusqu'à l'arrivée des méthodes d'apprentissage automatique de représentation, l'étape cruciale de la construction d'une représentation était réalisée "à la main". Aujourd'hui, une des approches les plus populaires en apprentissage de représentation repose sur les **réseaux de neurones profonds**

3. De nombreux modèles d'apprentissage automatique s'expriment comme un vote de majorité, c'est par exemple le cas des forêts aléatoires ou des SVM.

4. Un algorithme auto-certifié *(self-bounding algorithm)* optimise directement une borne en généralisation, garantissant que le modèle appris est accompagné de garantie théorique sur sa performance.

(*deep learning*), qui ont permis d'importantes avancées dans de nombreux domaines d'applications (vision par ordinateur, traitement automatique de la langue, bio-informatique, etc). Cependant, le succès de ces méthodes ne repose que sur peu de résultats théoriques. D'autres approches d'apprentissage, rassemblées sous le terme d'**apprentissage de métrique**, sont mieux comprises d'un point de vue théorique, comme les méthodes à noyaux, mais leurs succès en apprentissage de représentation sont actuellement moins retentissants.

J'ai amorcé, par le biais du projet JCJC INS2I-CNRS PaRaFF (2018), l'étude théorique de l'apprentissage de représentation sous l'angle de la théorie PAC-Bayésienne encore peu exploré jusqu'ici. C'est ce projet qui nous a permis de mettre en place les fondements du projet ANR APRIORI dont j'ai été la responsable scientifique coordinatrice. Cela a pris la forme d'une contribution **[C7]** dans laquelle nous avons proposé une ré-interprétation PAC-Bayésienne de la méthode d'approximation d'une fonction noyau connue sous le nom de **Random Fourier Features**. Dans le contexte de la thèse de Léo Gautheron, cette interprétation nous a permis de développer un nouvel algorithme d'apprentissage de métrique dans le cadre de la classification binaire **[C6]** qui a l'intérêt de pouvoir considérer très **peu de données** (contrairement aux réseaux de neurones profonds). Durant cette thèse, nous avons également proposé un algorithme d'apprentissage de métrique "classique" (en apprenant une métrique de type Mahalanobis) dans le cas de la classification de **données déséquilibrées [J2,C8]**. À noter également que les résultats obtenus dans le contexte de la thèse de Paul Viallard, cités dans la section précédente, ont pu être appliqués et testés efficacement sur des réseaux de neurones.

1.4.3 Apprentissage supervisé et vote de majorité

Dans le cadre de la classification supervisée où l'objectif est d'apprendre un modèle de classification à partir de données étiquetées, il existe de nombreux types de modèles. Depuis ma thèse, déjà en lien avec la théorie PAC-Bayésienne, je m'intéresse aux modèles s'exprimant comme une combinaison pondérée de fonctions (ce type de modèles fait partie des méthodes ensemblistes). Dans ce cadre, nous avons donc fait appel aux outils PAC-Bayésiens permettant d'apprendre des votes de majorité pondérés. Pour ce faire, nous avons étudié une borne sur l'erreur du vote de majorité qui fait intervenir un compromis entre performance et diversité des fonctions (avec l'idée qu'une combinaison n'a de sens que si les fonctions apportent de l'information suffisamment complémentaire). Cette borne est connue sous le nom de **C-borne** en théorie PAC-Bayésienne[5]. Dans ce contexte, MinCq (Roy et al., 2011) est le premier algorithme de classification binaire dont l'objectif est de minimiser la C-borne. Dans la continuité de ma thèse, nous avons proposé une adaptation de cet algorithme pour ajuster plus précisément les poids intervenant dans la pondération des fonctions **[J7]**. Nous avons appliqué cet algorithme à l'apprentissage d'une combinaison de fonctions de type k-NN (avec différentes valeurs de k), avec l'idée que la considération de différents voisinages apporte une information plus riche qu'un seul k-voisinage. Bien que justifiées par des bornes PAC-Bayésiennes, ces contributions sont plutôt **algorithmiques**, sans intervention de la borne en généralisation associée à la C-borne. Nous nous sommes intéressés à ce problème en dérivant de nouveaux résultats théoriques pour le vote de majorité pour adapter la C-borne à de la **classification multiclasse et multilabel [J5,W4]**, puis en développant des **algorithmes auto-certifiés** pour la minimiser **[C5]** (*i.e.*, des algorithmes qui optimisent

5. La "C-borne" a été introduite en apprentissage automatique par Breiman (2001) dans le cadre des *Random Forest*.

directement la borne en généralisation). L'ensemble de ces résultats peut non seulement s'appliquer à l'apprentissage supervisé, mais également à l'apprentissage multi-vues.

1.4.4 Apprentissage Multi-vues

L'objectif de l'**apprentissage multi-vues** (ou multimodal) est de permettre la prise en considération, lors de l'apprentissage, de différentes représentations des données et en particulier de leur complémentarité et diversité. Cette problématique peut être vue comme l'apprentissage d'une combinaison de modèles appris à partir de différentes représentations, vues ou modalités des données (on parle parfois de **fusion tardive**). La C-borne évoquée ci-dessus apparaît donc comme une solution naturelle à cette problématique, puisqu'elle permet de prendre en considération la diversité/complémentarité des fonctions impliquées dans la combinaison. À partir de la C-borne, une de mes premières contributions lors de mon post-doctorat a été d'adapter l'algorithme MinCq au cas spécifique de la fusion tardive pour du *ranking* d'images **[W3]**.

Cet intérêt pour l'apprentissage multi-vues, m'a permis d'intégrer, au moment de mon recrutement en tant que Maître de Conférences, le projet ANR LIVES dont l'apprentissage multi-vues était le fil conducteur. Durant ce projet et la thèse d'Anil Goyal, nous avons proposé la première **étude théorique PAC-Bayésienne de l'apprentissage multi-vues** lorsque l'on considère plus de deux vues **[C10]**. Cette étude théorique nous a permis de dériver plusieurs algorithmes d'apprentissage permettant de considérer différentes vues tout en considérant leur complémentarité **[J4,C9]**.

1.4.5 Adaptation de Domaine

En apprentissage supervisé, nous supposons généralement que le modèle appris sera appliqué sur de nouvelles données issues de la même tâche, c'est-à-dire de la même distribution sous-jacente des données. Cependant, cette hypothèse forte est difficile à vérifier pour des tâches réelles. En effet, de nombreuses applications requièrent que le modèle soit capable de s'adapter à une nouvelle distribution de données (par exemple suite à un changement de méthode ou de matériel d'acquisition). L'**adaptation de domaine**, un sous-domaine de l'apprentissage par transfert, a donc pour objectif de proposer des méthodes d'apprentissage capable d'apprendre, à partir de données issues d'une tâche source (le domaine source), un modèle efficace sur des données issues d'une tâche cible (le domaine cible, pour lequel nous ne disposons que de peu d'information). Dans ce cadre, durant ma thèse, nous avons développé la première **théorie PAC-Bayésienne pour l'adaptation de domaine** permettant, d'une part, d'obtenir des garanties théoriques sur la tâche cible et, d'autre part, de dériver des algorithmes. À la suite de ma thèse, nous avons amélioré ce résultat pour proposer une analyse théorique plus fine de l'adaptation de domaine en tirant parti des spécificités du PAC-Bayes **[J3,J6,C11,W2,W5]**. Cette analyse apporte une philosophie différente et originale sur la manière de s'attaquer à l'adaptation de domaine. En effet, la théorie classique de l'adaptation de domaine suggère qu'il faut apprendre une représentation dans laquelle les données sources et les données cibles seront indiscernables tout en gardant de bonnes garanties en généralisation pour la tâche source. Autrement dit, il faut trouver un bon compromis entre minimisation de l'erreur sur les données sources et minimisation d'une distance entre les deux domaines. Notre point de vue PAC-Bayésien suggère, quant à lui, que pour apprendre une bonne combinaison de fonctions, il faut trouver un bon compromis, pondéré par la dis-

tance entre les deux domaines, entre les erreurs jointes [6] des fonctions sur les données sources et la diversité des fonctions sur les données cibles. Il est important de noter que la théorie de l'adaptation de domaine est une problématique que j'étudie depuis mon stage de recherche en Master 2. De ce fait, j'ai pu acquérir une expertise ayant mené à la **co-rédaction d'un livre** sur l'état de l'art de la théorie de l'adaptation de domaine dans toute sa généralité **[L1]**.

1.4.6 Apprentissage Équitable et Robuste

Aujourd'hui, un des objectifs du projet ANR FAMOUS, qui a débuté en 2024 et dont je suis membre, est de s'attaquer à l'apprentissage multi-vues pour apprendre des modèles équitables et robustes. Les modèles issus des méthodes d'apprentissage automatique sont souvent fortement biaisés et amènent à des prises de décisions non équitables. Cet inconvénient majeur est une problématique très importante et la question de l'**équité** des méthodes d'apprentissage automatique fait partie de l'un des axes prioritaires dans la stratégie nationale du plan IA France 2030. Le comportement non équitable des méthodes est principalement dû aux biais présents dans les données utilisées lors de l'apprentissage. Bien que les bornes en généralisation classiques soient capables d'estimer dans quelles mesures l'erreur d'un modèle sur les données d'apprentissage est un bon estimateur de son erreur sur de nouvelles données (pour la même tâche), elles n'apportent pas nécessairement de garantie sur l'équité des modèles face aux biais intrinsèques des données ou de la structure du modèle, ni sur la robustesse des décisions face à des attaques extérieures. Un premier résultat, issu de la thèse de Paul Viallard, est la première **analyse théorique PAC-Bayésienne de la robustesse** face à des attaques **[C4]**. Un second résultat **[C1]** prend le point de vue de l'attaquant en proposant une extension des attaques dites universelles basées sur une analyse théorique (non PAC-Bayésienne).

Bien que n'ayant encore pas de publication dans le domaine de l'apprentissage équitable, une partie de mes travaux actuels se placent dans ce cadre avec, entre autres, le projet ANR FAMOUS et le début, en 2024, de deux thèses que je co-encadre. Une première, directement financée par le projet FAMOUS, s'intéresse plus particulièrement à l'apprentissage multi-vues, la seconde, financée par l'École Doctorale SIS, a pour objectif premier de dériver des bornes en généralisation PAC-Bayésienne pour l'apprentissage équitable et robuste.

1.5 Organisation de ce manuscrit

Il ressort du résumé précédent que la majorité de mes contributions s'inscrit dans le cadre de la théorie PAC-Bayésienne. La suite de ce manuscrit retrace, en quelque sorte, mon parcours scientifique au sein de cette théorie depuis la fin de ma thèse. Plus précisément :

— Le Chapitre 2 rappelle les notions nécessaires à la compréhension des bornes en généralisation avec un focus sur la théorie PAC-Bayésienne.

— La Partie II présente les analyses théoriques PAC-Bayésiennes qui ont servi de sources d'inspiration pour le développement d'algorithmes dans le cadre de l'apprentissage multi-vues, de l'adaptation de domaine et de l'apprentissage à partir de *Random Fourier Features*.

6. L'erreur jointe comptabilise une erreur lorsque si deux fonctions commettent une erreur sur une même donnée.

— La Partie III se concentre sur les résultats théoriques directement optimisables pour concevoir des algorithmes auto-certifiés, pour les votes de majorités et la robustesse adversaire.

— Enfin, la Partie IV explore l'obtention de nouvelles bornes, sortant des sentiers battus et ouvrant la voie à de nouveaux résultats et à de nouvelles perspectives pour le développement d'algorithmes auto-certifiés dans divers cadres de l'apprentissage automatique.

Ce manuscrit témoigne ainsi de l'évolution de mes travaux et des apports méthodologiques qu'ils offrent à la communauté scientifique, tout en soulignant leur potentiel pour enrichir les applications pratiques.

Avant-propos

Afin de simplifier la lecture et d'alléger ce manuscrit :

- aucune preuve des résultats théoriques n'est présentée ;
- seul un résumé des expériences menées est présenté.

Toutes les preuves et les résultats expérimentaux disponibles au moment de leur publication figurent dans les articles associés.

Généralités sur les bornes en généralisation et sur la théorie PAC-Bayésienne

2

Contexte

Ce chapitre introduit rapidement certaines notions essentielles à la compréhension des contributions présentées dans ce manuscrit.

2.1 Introduction

L'apprentissage statistique, introduit par Vapnik et Chervonenkis (1968, 1971), est lié à la théorie d'études statistiques sur les processus empiriques. L'ensemble de mes travaux s'intéresse au paradigme de la classification supervisée qui s'oppose à celui de la classification non supervisée (par exemple le *clustering*) pour laquelle nous ne disposons d'aucune supervision sur la classe des données observées. La classification supervisée se formalise comme l'apprentissage ou la construction d'une fonction (souvent appelée hypothèse, modèle, classifieur ou classificateur) à partir d'un ensemble d'apprentissage composé de données observées et étiquetées supposées indépendantes et identiquement distribuées selon une distribution de probabilité inconnue qui modélise la tâche étudiée. Dans ce cadre, le modèle appris doit non seulement bien s'adapter aux données observées, mais surtout être performant sur de nouvelles données. Autrement dit, il doit être capable de généraliser sur l'ensemble de la distribution des données. Cependant, puisque cette distribution est inconnue, il est impossible de mesurer directement et empiriquement la capacité en généralisation d'un modèle sur des données jamais observées. Une question fondamentale, qui est au cœur de ce manuscrit, est donc l'obtention de garanties théoriques concernant cette capacité en généralisation.

Ce genre de garanties prend souvent la forme d'une borne sur le risque du modèle sur la distribution, *i.e.* le risque réel ou l'erreur en généralisation, et fait intervenir des quantités mesurables ou estimable à partir d'un échantillon de données. Ces bornes sont appelées "bornes en généralisation."

2.2 La classification supervisée — Formalisation

Soit $\mathbb{X} \subseteq \mathbb{R}^{\mathrm{d}}$ un espace d'entrée de dimension d et $\mathbb{Y}$ un espace d'étiquetage ou de classes (en classification binaire $\mathbb{Y} = \{-1, +1\}$, en classification multiclasse $\mathbb{Y} = \{1, 2, \ldots, l\}$ avec $l \geq 2$. La tâche d'apprentissage est modélisée par une distribution de probabilité jointe fixée et inconnue $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$. Étant donné une famille d'hypothèses $\mathbb{H}$ définie comme un ensemble (fini ou infini) de fonctions $h : \mathbb{X} \to \mathbb{Y}$ que l'on appellera indistinctement hypothèses, modèles, votants ou classifieurs, l'objectif de l'apprenant est de trouver le modèle $h \in \mathbb{H}$ qui capture au mieux la relation entre $\mathbb{X}$ et $\mathbb{Y}$ modélisée par $\mathcal{D}$. En classification supervisée, l'apprenant dispose d'un ensemble d'apprentissage constitué d'exemples déjà étiquetés, défini ci-dessous.

Définition 2.2.1 (Ensemble d'apprentissage)**.** Un ensemble d'apprentissage $\mathbb{S}$ est un ensemble de m variables aléatoires indépendamment et identiquement distribuées (*i.i.d.*) selon une distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$. Il est défini par

$$\mathbb{S} = \bigcup_{i=1}^{m} \Big\{(\mathbf{x}_i, y_i)\Big\} = \Big\{(\mathbf{x}_i, y_i)\Big\}_{i=1}^{m} \in (\mathbb{X}\times\mathbb{Y})^m,$$
$$\text{où} \quad \forall i \in \{1, \ldots, m\}, \quad (\mathbf{x}_i, y_i) \sim \mathcal{D} \quad \text{et} \quad \mathbb{S} = \Big\{(\mathbf{x}_i, y_i)\Big\}_{i=1}^{m} \sim \mathcal{D}^m,$$

avec $\mathcal{D}^m$ la distribution de m exemples suivants $\mathcal{D}$. On a $\mathcal{D}^m(\mathbb{S}) = \prod_{i=1}^{m} \mathcal{D}\Big((\mathbf{x}_i, y_i)\Big)$.

À l'aide de l'information portée par l'ensemble d'apprentissage $\mathbb{S}$, l'apprenant doit choisir un modèle h dans $\mathbb{H}$, de telle sorte que ce modèle h décrive au mieux la relation entre $\mathbb{X}$ et $\mathbb{Y}$. Autrement dit, h doit avoir une bonne qualité en généralisation : étant donné un nouvel exemple $(\mathbf{x}, y)$ tiré aléatoirement selon $\mathcal{D}$, la prédiction $h(\mathbf{x})$ du modèle doit être la plus proche de la vraie valeur de y. Pour mesurer cette qualité, on fait généralement appel à une fonction perte $\ell : \mathbb{H} \times (\mathbb{X}\times\mathbb{Y}) \to [0, 1]$ qui associe un coût $\ell(h, (\mathbf{x}, y))$ à la prédiction de h sur l'exemple $(\mathbf{x}, y)$. Cette notion sert, en général, à évaluer une mauvaise réponse relativement à la fonction perte. L'espérance de ce coût sur les données tirées selon la distribution $\mathcal{D}$ est appelée le risque réel et est défini ci-dessous.

Définition 2.2.2 (Risque réel)**.** Soit une fonction perte $\ell : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y}) \to [0, 1]$, le risque réel (ou encore l'erreur en généralisation) $\mathrm{R}^{\ell}_{\mathcal{D}}(h)$ d'un modèle h de $\mathbb{H}$ sur une distribution de données $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$ est

$$\mathrm{R}^{\ell}_{\mathcal{D}}(h) = \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} \ell(h, (\mathbf{x}, y)).$$

Le meilleur modèle est donc celui qui minimise ce risque réel. Or, quel que soit le modèle h, son risque réel $\mathrm{R}^{\ell}_{\mathcal{D}}(h)$ ne peut pas être calculé car la loi de probabilité $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$ est inconnue. Les seules informations disponibles sont celles portées par l'ensemble $\mathbb{S}$. En pratique, le risque réel est donc "estimé" à l'aide du risque empirique évalué sur $\mathbb{S}$ et défini comme suit.

Définition 2.2.3 (Risque empirique)**.** Soit une fonction perte $\ell : \mathbb{H} \times (\mathbb{X} \times \mathbb{Y}) \to [0, 1]$, le risque empirique $\widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h)$ d'un modèle h de $\mathbb{H}$ sur un ensemble de m exemples $\mathbb{S} \sim \mathcal{D}^m$ est

$$\widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h) = \frac{1}{m} \sum_{i=1}^{m} \ell(h, (\mathbf{x}_i, y_i)).$$

Dans le meilleur des mondes, nous disposerions d'une infinité d'exemples d'apprentissage et trouver le modèle qui minimise le risque empirique serait la stratégie la plus pertinente. En réalité, le nombre d'exemples disponibles m est limité et il existe souvent un modèle h (parfois complexe) d'erreur empirique nulle. Face à de nouvelles données jamais observées, ce modèle, parfait en apparence, ne montrera pas toujours de bonnes performances et l'erreur réelle pourra s'avérer — beaucoup — plus élevée que l'erreur empirique : c'est le phénomène de sur-apprentissage. Pour éviter ce problème d'apprentissage "par cœur", une solution est de considérer le compromis biais/variance, se résumant principalement en un juste équilibre entre l'erreur empirique et la complexité de l'ensemble d'hypothèses : selon le principe du rasoir d'Occam, à performance égale, on préfère un modèle simple à un modèle complexe.

2.3 Les bornes en généralisation en quelques mots

Le principe énoncé ci-dessus peut se justifier avec des garanties théoriques appelées bornes en généralisation. La quantité d'intérêt pour dériver des bornes en généralisation est l'écart en généralisation *(generalization gap)* "$\mathsf{R}^{\ell}_{\mathcal{D}}(h) - \widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h)$" qui permet d'estimer à quel point le risque empirique d'un modèle peut être proche de son risque réel. Pour ce faire, les bornes en généralisation dîtes PAC *(Probably Approximately Correct)* (Valiant, 1984) majorent le risque réel avec grande probabilité sur le choix aléatoire de l'ensemble d'apprentissage $\mathbb{S} \sim \mathcal{D}^m$. Ce type de borne a la forme suivante :

$$\mathop{\mathbb{P}}_{\mathbb{S} \sim \mathcal{D}^m} \left[\mathsf{R}^{\ell}_{\mathcal{D}}(h) \leq \Phi \right] \geq 1 - \delta, \tag{2.1}$$

où $\Phi \geq 0$ et $\delta \in]0, 1]$. Ainsi, avec une probabilité d'au moins $1 - \delta$ (le *Probably* du terme PAC), le risque réel du modèle h est majoré par Φ (le *Approximately Correct* de PAC). À partir de cette formulation, Valiant (1984) a introduit la "PAC-apprenabilité" *(PAC-learnability)* : un ensemble d'hypothèses $\mathbb{H}$ est PAC-apprenable si l'Équation (2.1) reste vraie quand le nombre d'exemples m est polynomial en $\frac{1}{\delta}$ et $\frac{1}{\Phi}$. Pour obtenir Φ, en pratique, on utilise des inégalités de concentration qui permettent de borner une espérance (ici, le risque réel) par son estimation empirique (ici, le risque empirique).

2.3.1 Bornes en convergence uniforme

2.3.1.1 Définition générale

En fait, le premier cadre théorique pour dériver des bornes en généralisation PAC a été introduit par Vapnik et Chervonenkis (1968, 1971) : les bornes en convergence uniforme. Ce sont des bornes qui, étant donné un ensemble d'hypothèses $\mathbb{H}$, restent valables pour toutes les hypothèses de $\mathbb{H}$. Elles ont la forme suivante.

Définition 2.3.1 (Borne en convergence uniforme)**.** Soit $\phi : [0,1]^2 \to \mathbb{R}$ une mesure de l'écart en généralisation. Étant donné une distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, un ensemble d'hypothèses $\mathbb{H}$ et une fonction perte $\ell : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y}) \to [0,1]$, s'il existe une fonction $\Phi_{\mathrm{u}} :]0,1] \to \mathbb{R}$, telle que pour tout $\delta \in]0,1]$ on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\sup_{h\in\mathbb{H}} \phi\left(\mathsf{R}^{\ell}_{\mathcal{D}}(h), \widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h)\right) \leq \Phi_{\mathrm{u}}(\delta)\right] \geq 1-\delta, \tag{2.2}$$

alors l'Équation (2.2) est une borne en convergence uniforme.
En général, on peut considérer $\phi\left(\mathsf{R}^{\ell}_{\mathcal{D}}(h), \widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h)\right) = \left|\mathsf{R}^{\ell}_{\mathcal{D}}(h) - \widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h)\right|$.

Avec une probabilité d'au moins $1-\delta$ sur le choix aléatoire de l'ensemble $\mathbb{S}\sim\mathcal{D}^m$, une borne en convergence uniforme majore l'écart en généralisation pour toutes les hypothèses $h \in \mathbb{H}$ avec $\Phi_{\mathrm{u}}(\delta)$. On parle souvent de bornes en "pire cas" puisque considérer toutes les hypothèses revient à majorer l'écart en généralisation par une borne supérieure sur le plus grand écart en généralisation : l'hypothèse associée à ce plus grand écart est la "pire" hypothèse de $\mathbb{H}$. Ainsi, sachant que $\Phi_{\mathrm{u}}(\delta)$ dépend[1] également de de m, si $\lim_{m\to+\infty}\Phi_{\mathrm{u}}(\delta) = 0$, avec une probabilité d'au moins $1-\delta$ sur $\mathbb{S}\sim\mathcal{D}^m$, le risque empirique converge uniformément sur $\mathbb{H}$ vers le risque réel. Cette convergence uniforme implique que $\Phi_{\mathrm{u}}(\delta)$ dépend d'une mesure de complexité qui doit capturer la performance de la pire hypothèse de $\mathbb{H}$.

2.3.1.2 Convergence uniforme pour un ensemble fini d'hypothèses

Lorsque l'ensemble d'hypothèses est fini, une mesure de complexité simple et naturelle est le cardinal de $\mathbb{H}$ noté $\mathrm{card}(\mathbb{H})$. Cette mesure implique que plus on a d'hypothèses, plus la complexité est grande. La borne en généralisation ci-dessous est une instanciation de la Définition 2.3.1 (*e.g.*, Mohri et al. (2012) pour plus de détails).

Théorème 2.3.1 (Borne en généralisation pour $\mathbb{H}$ fini)**.** Étant donné une distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, un ensemble fini d'hypothèses $\mathbb{H}$ (*i.e.*, $\mathrm{card}(\mathbb{H}) < +\infty$) et une une fonction perte $\ell : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y}) \to [0,1]$, pour tout $\delta \in]0,1]$ on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\sup_{h\in\mathbb{H}}\left|\mathsf{R}^{\ell}_{\mathcal{D}}(h) - \widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h)\right| \leq \underbrace{\sqrt{\frac{\ln \mathrm{card}(\mathbb{H}) + \ln\frac{1}{\delta}}{2m}}}_{\Phi_{\mathrm{u}}(\delta)}\right] \geq 1-\delta. \tag{2.3}$$

L'Équation (2.3) implique la borne supérieure[2] suivante :

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\forall h \in \mathbb{H}, \quad \mathsf{R}^{\ell}_{\mathcal{D}}(h) \leq \widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h) + \sqrt{\frac{\ln \mathrm{card}(\mathbb{H}) + \ln\frac{1}{\delta}}{2m}}\right] \geq 1-\delta.$$

1. Afin de simplifier la lecture, nous utilisons la notation $\Phi_{\mathrm{u}}(\delta)$ au lieu de $\Phi_{\mathrm{u}}(\delta, m)$.
2. Plus précisément, l'Équation (2.3) est équivalente au résultat suivant qui combine la borne inférieure et la borne supérieure : $\mathbb{P}_{\mathbb{S}\sim\mathcal{D}^m}[\forall h \in \mathbb{H},\ \widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h) - \Phi_{\mathrm{u}}(\delta) \leq \mathsf{R}^{\ell}_{\mathcal{D}}(h) \leq \widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h) + \Phi_{\mathrm{u}}(\delta)] \geq 1-\delta$.

Puisque la complexité est ici le cardinal de $\mathbb{H}$, son calcul peut être facile. Cependant, cette borne ne converge pas quand $\mathrm{card}(\mathbb{H})$ est grand. Par exemple, si $\mathrm{card}(\mathbb{H}) \geq e^m$ pour $m \in \mathbb{N}$, la borne est $\sqrt{\frac{1}{2}} \leq \sqrt{\frac{1}{2m}(\ln \mathrm{card}(\mathbb{H}) + \ln \frac{1}{\delta})}$. En pratique, $\mathbb{H}$ est grand, voir infini, il est donc nécessaire d'avoir des bornes en généralisation pour de tels ensembles.

2.3.1.3 La dimension de Vapnik-Chervonenkis

La dimension VC (Vapnik et Chervonenkis, 1968, 1971), rappelée ci-dessous, permet d'obtenir des bornes en généralisation pour des ensembles infinis d'hypothèses avec la fonction perte 0-1 définie par $\ell_{01}(h, (\mathbf{x}, y)) = \mathrm{I}[h(\mathbf{x}) \neq y]$.

Définition 2.3.2 (Dimension VC)**.** La dimension VC, notée $\mathtt{vc}(\mathbb{H})$, d'un ensemble d'hypothèses $\mathbb{H}$ en classification binaire est définie comme le nombre maximum m d'exemples tel qu'on puisse toujours trouver une fonction $h \in \mathbb{H}$ qui classe parfaitement ces m exemples, quelle que soit leur étiquette :

$$\mathtt{vc}(\mathbb{H}) = \max \left\{ m \ : \ \forall \mathbb{S} \in (\mathbb{X} \times \mathbb{Y})^m, \exists h \in \mathbb{H} \text{ telle que } \widehat{\mathsf{R}}_{\mathbb{S}}(h) = 0 \right\},$$
$$\text{où} \quad \widehat{\mathsf{R}}_{\mathbb{S}}(h) = \frac{1}{m} \sum_{i=1}^{m} \ell_{01}(h, (\mathbf{x}, y)).$$

La dimension VC correspond donc à la quantité maximale de données telle qu'il existe une hypothèse de $\mathbb{H}$ qui soit consistante avec n'importe quel étiquetage. Cette définition permet de prouver la borne suivante (Mohri et al., 2012, Th 3.17, Cor 3.18-19).

Théorème 2.3.2 (Borne en généralisation basée sur la dimension VC)**.** Étant donné une distribution $\mathcal{D}$ sur $\mathbb{X} \times \mathbb{Y}$ et un ensemble $\mathbb{H}$ tel que $\forall h \in \mathbb{H}$, $h : \mathbb{X} \to \{-1, +1\}$, pour tout $\delta \in]0, 1]$, on a

$$\mathop{\mathbb{P}}_{\mathbb{S} \sim \mathcal{D}^m} \left[\sup_{h \in \mathbb{H}} \left[\mathsf{R}_{\mathcal{D}}(h) - \widehat{\mathsf{R}}_{\mathbb{S}}(h) \right] \leq \underbrace{\sqrt{\frac{2\mathtt{vc}(\mathbb{H}) \left(1 + \ln \frac{m}{\mathtt{vc}(\mathbb{H})}\right)}{m}} + \sqrt{\frac{\ln \frac{1}{\delta}}{2m}}}_{\Phi_{\mathrm{u}}(\delta)} \right] \geq 1 - \delta.$$

De manière équivalente, on a

$$\mathop{\mathbb{P}}_{\mathbb{S} \sim \mathcal{D}^m} \left[\forall h \in \mathbb{H}, \quad \mathsf{R}_{\mathcal{D}}(h) \leq \widehat{\mathsf{R}}_{\mathbb{S}}(h) + \sqrt{\frac{2\mathtt{vc}(\mathbb{H}) \left(1 + \ln \frac{m}{\mathtt{vc}(\mathbb{H})}\right)}{m}} + \sqrt{\frac{\ln \frac{1}{\delta}}{2m}} \right] \geq 1 - \delta.$$

Cette borne indique qu'avec une confiance de $1-\delta$, le risque empirique d'une hypothèse tend vers son risque réel lorsque la taille m de l'ensemble d'apprentissage augmente, et ce, d'autant plus "vite" que la dimension VC est faible. Quand la dimension VC de $\mathbb{H}$ est finie, la loi des grands nombres et la convergence uniforme impliquent que le risque empirique de l'hypothèse qui minimise le risque empirique et son risque réel convergent tous les deux en probabilité vers le minimum du risque sur $\mathbb{H}$. Lorsqu'un algorithme d'apprentissage permet

de vérifier cette propriété, on dit qu'il est consistant. Par exemple, on peut prouver que l'algorithme ERM est consistant (Vapnik, 1998) [3].

En pratique, bien que calculable lorsque $\mathtt{vc}(\mathbb{H})$ est connue, un des inconvénients de cette borne, est que $\mathtt{vc}(\mathbb{H})$ peut être difficile à calculer (ou même valoir l'infini). De plus, elle est définie uniquement pour la classification binaire avec la perte 0-1. La section suivante présente une extension basée sur la complexité de Rademacher (Bartlett et Mendelson, 2002) permettant de considérer le cas de la classification multiclasses.

2.3.1.4 La complexité de Rademacher

Intuitivement, la complexité de Rademacher (Bartlett et Mendelson, 2002) mesure la capacité d'un ensemble d'hypothèses à résister au bruit et peut amener à des bornes plus précises que celles basées sur la dimension VC.

Définition 2.3.3 (Complexité de Rademacher)**.** La complexité de Rademacher, notée $\mathtt{rad}(\mathbb{H})$, d'un ensemble d'hypothèses $\mathbb{H}$ pour une fonction perte $\ell : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y}) \to [0,1]$ est définie par

$$\mathtt{rad}(\mathbb{H}) = \mathop{\mathbb{E}}_{\mathbb{S}\sim\mathcal{D}^m} \mathop{\mathbb{E}}_{\{\kappa_1,\ldots,\kappa_m\}\sim\mathcal{K}^m} \sup_{h\in\mathbb{H}} \frac{1}{m}\sum_{i=1}^{m} \kappa_i\, \ell(h,(\mathbf{x}_i,y_i)),$$

où $\mathcal{K}$ est la distribution de Rademacher, *i.e.*, $\mathcal{K}(+1){=}\mathcal{K}(-1){=}\frac{1}{2}$.

Étant donné un ensemble d'apprentissage $\mathbb{S} \sim \mathcal{D}^m$ et les variables de Rademacher $\{\kappa_1,\ldots,\kappa_m\}\sim\mathcal{K}^m$, le supremum sur l'ensemble $\mathbb{H}$ est atteint quand la perte $\ell(h,(\mathbf{x}_i,y_i))$ est maximisée, *resp.* minimisée, pour $\kappa_i{=}+1$, *resp.* $\kappa_i{=}-1$. Tandis que la dimension VC considère l'étiquetage le plus difficile pour $\mathbb{H}$, la complexité de Rademacher peut être vue comme l'espérance sur tous les étiquetages possibles. Autrement dit, la complexité de Rademacher mesure la capacité à apprendre à partir d'étiquettes aléatoires. En utilisant l'inégalité (de concentration) de McDiarmid (1989), Bartlett et Mendelson (2002) ont dérivé la borne en généralisation suivante.

Théorème 2.3.3 (Borne en généralisation basée sur la complexité de Rademacher)**.** Étant donné une distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, un ensemble d'hypothèses $\mathbb{H}$ et une fonction perte $\ell : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y}) \to [0,1]$, pour tout $\delta \in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\sup_{h\in\mathbb{H}}\left|\mathsf{R}^{\ell}_{\mathcal{D}}(h) - \widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h)\right| \le \underbrace{2\,\mathtt{rad}(\mathbb{H}) + \sqrt{\frac{\ln\frac{1}{\delta}}{2m}}}_{\Phi_{\mathrm{u}}(\delta)}\right] \ge 1{-}\delta.$$

De manière équivalente, on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\forall h\in\mathbb{H},\quad \mathsf{R}^{\ell}_{\mathcal{D}}(h) \le \widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h) + 2\,\mathtt{rad}(\mathbb{H}) + \sqrt{\frac{\ln\frac{1}{\delta}}{2m}}\right] \ge 1{-}\delta.$$

3. Voir la Prop. 4.1 de Mohri et al. (2012) pour des détails sur la consistance de ERM.

L'écart en généralisation est ici majoré par un compromis entre la complexité de Rademacher de $\mathbb{H}$ et un terme qui tend à être faible lorsque m est grand. Comme avec la dimension VC, si $\texttt{rad}(\mathbb{H})$ (ou une borne supérieure) est connue, la borne en généralisation est calculable pour toutes les hypothèses de $\mathbb{H}$. Cependant, à l'instar des bornes basées sur la dimension VC, ces bornes peuvent être vues comme une analyse en pire cas. En effet, la valeur de la borne est la même pour toutes les hypothèses de $\mathbb{H}$, allant de la meilleure à la pire (associée au plus grand écart en généralisation).

Un inconvénient de ces analyses en pire cas est que l'obtention de bornes précises (*i.e.*, $\Phi_{\mathtt{u}}(\delta) < 1$) est difficile. Pour contourner ce problème, des bornes prenant en compte l'exploration de l'ensemble d'hypothèses par l'algorithme d'apprentissage ont été dérivées.

2.3.2 Bornes en généralisation dépendantes d'un algorithme

La manière dont l'ensemble d'hypothèses $\mathbb{H}$ est exploré dépend de l'algorithme d'apprentissage. Il est donc important de relier les capacités en généralisation aux spécificités de l'algorithme utilisé. Dans cette section, nous considérons un algorithme d'apprentissage qui prend en entrée un ensemble d'apprentissage $\mathbb{S} \in (\mathbb{X}\times\mathbb{Y})^m$ et qui renvoie une hypothèse $h_{\mathbb{S}}$. L'ensemble des hypothèses est alors noté $\mathbb{H} = \{h_{\mathbb{S}}\}_{\mathbb{S}\in(\mathbb{X}\times\mathbb{Y})^m}$. Une borne en généralisation, qui dépend d'un tel algorithme, a la forme suivante.

Définition 2.3.4 (Borne dépendante de l'algorithme)**.** Soit $\phi : [0,1]^2 \to \mathbb{R}$ une mesure de l'écart en généralisation. Étant donné une distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, un ensemble d'hypothèses $\mathbb{H}$ et une fonction perte $\ell : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y}) \to [0,1]$, s'il existe une fonction $\Phi_{\mathtt{a}} :]0,1] \to \mathbb{R}$, telle que pour tout $\delta \in]0,1]$ on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\phi\Big(\mathsf{R}^{\ell}_{\mathcal{D}}(h_{\mathbb{S}}), \widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h_{\mathbb{S}})\Big) \leq \Phi_{\mathtt{a}}(\delta)\right] \geq 1-\delta, \tag{2.4}$$

alors l'Équation (2.4) est une borne dépendante de l'algorithme considéré.
En général, $\phi(\mathsf{R}^{\ell}_{\mathcal{D}}(h_{\mathbb{S}}), \widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h_{\mathbb{S}})) = \mathsf{R}^{\ell}_{\mathcal{D}}(h_{\mathbb{S}}) - \widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h_{\mathbb{S}})$ et $h_{\mathbb{S}}$ est l'hypothèse apprise par l'algorithme avec $\mathbb{S}\sim\mathcal{D}^m$.

Pour obtenir une valeur de borne $\Phi_{\mathtt{a}}(\delta)$, il faut prendre en compte une propriété de l'algorithme. Nous rappelons les propriétés de stabilité uniforme (Bousquet et Elisseeff, 2002) et de robustesse algorithmique (Xu et Mannor, 2010) dans les Sections 2.3.2.1 et 2.3.2.2.

2.3.2.1 La stabilité uniforme

Les résultats principaux de Bousquet et Elisseeff (2002) exploitent la capacité d'un algorithme à produire un résultat suffisamment stable face à de légères modifications de l'ensemble d'apprentissage.

Définition 2.3.5 (Stabilité uniforme)**.** Soit $\mathbb{H}=\{h_{\mathbb{S}}\}_{\mathbb{S}\in(\mathbb{X}\times\mathbb{Y})^m}$ un ensemble d'hypothèses et $\ell : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y}) \to [0,1]$ une fonction perte, un algorithme admet une stabilité uniforme β_{S} si

$$\sup_{\substack{\mathbb{S},\mathbb{S}'\in(\mathbb{X}\times\mathbb{Y})^m \\ \text{t.q. } \Delta(\mathbb{S},\mathbb{S}')=1}} \sup_{(\mathbf{x},y)\in\mathbb{X}\times\mathbb{Y}} \Big|\ell(h_{\mathbb{S}},(\mathbf{x},y)) - \ell(h_{\mathbb{S}'},(\mathbf{x},y))\Big| \leq \beta_{\mathsf{S}},$$

où $\Delta(\mathbb{S},\mathbb{S}') = \sum_{i=1}^{m} \mathrm{I}\left[(\mathbf{x}_i,y_i) \neq (\mathbf{x}'_i,y'_i)\right]$ est la distance de Hamming entre $\mathbb{S}$ et $\mathbb{S}'$.

Un algorithme est stable, si pour deux ensembles $\mathbb{S}$ et $\mathbb{S}'$ qui ne diffèrent que d'un seul exemple, la différence entre $h_{\mathbb{S}}$ et $h_{\mathbb{S}'}$ est faible pour tous les exemples $(\mathbf{x}, y) \in \mathbb{X}\times\mathbb{Y}$. Cette différence est majorée par le terme $\beta_{\mathbb{S}}$ relié à la fonction perte et à la régularisation utilisée par l'algorithme. En fait, $\beta_{\mathbb{S}}$ peut être vu comme une complexité qui dépend du nombre d'exemples m. Lorsque $\beta_{\mathbb{S}} = O(\frac{1}{\sqrt{m}})$ ou $\beta_{\mathbb{S}} = O(\frac{1}{m})$, on parle d'algorithme stable uniformément (plus m augmente, plus l'algorithme est stable). À partir de cette définition de stabilité uniforme et en utilisant l'inégalité de McDiarmid (1989), Bousquet et Elisseeff (2002) ont prouvé la borne en généralisation suivante.

Théorème 2.3.4 (Borne en généralisation basée sur la stabilité uniforme)**.** Soit un algorithme de stabilité uniforme $\beta_{\mathbb{S}}$, un ensemble d'hypothèses $\mathbb{H} = \{h_{\mathbb{S}}\}_{\mathbb{S}\in(\mathbb{X}\times\mathbb{Y})^m}$ et une fonction perte $\ell : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y}) \to [0,1]$. Étant donné une distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout $\delta \in]0,1]$ on a

$$\underset{\mathbb{S}\sim\mathcal{D}^m}{\mathbb{P}}\left[\mathrm{R}^{\ell}_{\mathcal{D}}(h_{\mathbb{S}}) - \widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h_{\mathbb{S}}) \leq 2\,\beta_{\mathbb{S}} + (4\,m\,\beta_{\mathbb{S}}+1)\sqrt{\frac{\ln\frac{1}{\delta}}{2m}}\right] \geq 1-\delta.$$

De manière équivalente, on a

$$\underset{\mathbb{S}\sim\mathcal{D}^m}{\mathbb{P}}\left[\mathrm{R}^{\ell}_{\mathcal{D}}(h_{\mathbb{S}}) \leq \widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h_{\mathbb{S}}) + 2\,\beta_{\mathbb{S}} + (4\,m\,\beta_{\mathbb{S}}+1)\sqrt{\frac{\ln\frac{1}{\delta}}{2m}}\right] \geq 1-\delta.$$

Si l'algorithme a une stabilité uniforme en $O(\frac{1}{\sqrt{m}})$ alors le taux de convergence est en $O(\frac{1}{\sqrt{m}})$. Si la stabilité est en $O(1)$, la borne ne converge pas. Comme pour la convergence uniforme, le terme $\beta_{\mathbb{S}}$ doit être connu (ou majoré) pour calculer la borne.

2.3.2.2 La robustesse algorithmique

Xu et Mannor (2010, 2012) ont démontré que certains algorithmes (dont ceux construisant un modèle parcimonieux) ne sont pas stables face à une faible modification de l'ensemble d'apprentissage. Pour contrer ce problème, ils ont proposé la notion de robustesse algorithmique. Un algorithme est dit robuste s'il obtient des performances similaires sur l'ensemble d'apprentissage $\mathbb{S}$ et sur un ensemble $\mathbb{T}$ "similaire" à $\mathbb{S}$; cette similarité est mesurée à l'aide d'un partitionnement de $\mathbb{X}\times\mathbb{Y}$ construit pour que deux exemples proches et de même classe appartiennent à la même partition.

Définition 2.3.6 (Robustesse algorithmique)**.** Soit un ensemble d'hypothèses $\mathbb{H}=\{h_{\mathbb{S}}\}_{\mathbb{S}\in(\mathbb{X}\times\mathbb{Y})^m}$, une fonction perte $\ell : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y}) \to [0,1]$ et N ensembles disjoints tels que $(\mathbb{X}\times\mathbb{Y}) = \bigcup_{i=1}^{N}\mathbb{Z}_i$, un algorithme est $(\{\mathbb{Z}_i\}_{i=1}^{N}, \beta_{\mathrm{R}})$-robuste si

$$\sup_{\mathbb{S}\in(\mathbb{X}\times\mathbb{Y})^m}\ \sup_{i\in\{1,\ldots,N\}}\ \sup_{(\mathbf{x},y),(\mathbf{x}',y')\in\mathbb{Z}_i}\left|\ell(h_{\mathbb{S}},(\mathbf{x},y)) - \ell(h_{\mathbb{S}},(\mathbf{x}',y'))\right| \leq \beta_{\mathrm{R}}.$$

Pour chaque $\mathbb{Z}_i \subseteq \mathbb{X}\times\mathbb{Y}$, la différence de la perte entre deux exemples $(\mathbf{x}, y) \in \mathbb{Z}_i$ et $(\mathbf{x}', y') \in \mathbb{Z}_i$ doit être majorée par β_{R} (qui peut dépendre de $\mathbb{S}$). Si un algorithme est robuste, alors la borne suivante peut être dérivée en utilisant l'inégalité de Breteganolle-Huber-Carol (Vaart et Wellner, 1996, Prop A.6.6).

Théorème 2.3.5 (Borne en généralisation basée sur la robustesse algorithmique)**.** Soit un algorithme $(\{\mathbb{Z}_i\}_{i=1}^N, \beta_R)$-robuste, un ensemble d'hypothèses $\mathbb{H} = \{h_S\}_{S\in(\mathbb{X}\times\mathbb{Y})^m}$ et une fonction perte $\ell : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y}) \to [0,1]$. Étant donné une distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout $\delta \in]0,1]$ on a

$$\mathop{\mathbb{P}}_{S\sim\mathcal{D}^m}\left[\mathrm{R}^{\ell}_{\mathcal{D}}(h_S) - \widehat{\mathrm{R}}^{\ell}_{S}(h_S) \leq \beta_R + \sqrt{\frac{2N\ln 2 + 2\ln\frac{1}{\delta}}{2m}}\right] \geq 1-\delta.$$

De manière équivalente, on a

$$\mathop{\mathbb{P}}_{S\sim\mathcal{D}^m}\left[\mathrm{R}^{\ell}_{\mathcal{D}}(h_S) \leq \widehat{\mathrm{R}}^{\ell}_{S}(h_S) + \beta_R + \sqrt{\frac{2N\ln 2 + 2\ln\frac{1}{\delta}}{2m}}\right] \geq 1-\delta.$$

Dans l'idéal, le paramètre β_R doit dépendre de m pour que la borne converge. On peut remarquer qu'il y a un compromis à trouver entre le nombre de partitions N et β_R. En effet, plus N est grand, plus β_R doit être petit. Or, lorsque N est trop grand, par exemple si $N \geq m$, alors la borne est supérieure à $\sqrt{\ln(2)}$ et ne converge pas vers 0 quand $m \to +\infty$.

Un inconvénient des bornes qui dépendent de propriétés algorithmiques telles que la stabilité uniforme ou la robustesse algorithmique est que le terme β_S ou β_R doit être calculé pour chaque algorithme. Cela rend la dérivation des bornes fastidieuse. Un autre type de bornes — les bornes PAC-Bayésiennes qui sont au cœur de ce manuscrit — n'ont pas cet inconvénient. Elles ont, en outre, l'intérêt de faciliter la dérivation d'algorithmes.

2.3.3 Bornes en généralisation PAC-Bayésiennes classiques

Nous présentons la forme générale des bornes PAC-Bayésiennes introduites par Shawe-Taylor et Williamson (1997) et McAllester (1999) (et détaillées dans la Section 2.4). Ces bornes diffèrent de celles présentées dans les Sections 2.3.1 et 2.3.2, car elles considèrent une distribution, notée ρ, sur l'ensemble d'hypothèses $\mathbb{H}$. Cette distribution attribue un poids $\rho(h)$ à chaque hypothèse $h \in \mathbb{H}$. L'idée est de construire ρ de telle sorte que plus l'hypothèse h généralise bien, plus son poids $\rho(h)$ soit grand. Dans ce contexte, les bornes PAC-Bayésiennes s'intéressent non plus au risque individuel $\widehat{\mathrm{R}}^{\ell}_{S}(h)$ d'une hypothèse $h \in \mathbb{H}$, mais à l'espérance sur $\mathbb{H}$ selon la distribution ρ des risques individuels $\mathbb{E}_{h\sim\rho}\,\widehat{\mathrm{R}}^{\ell}_{S}(h)$. Cette moyenne des erreurs peut être interprétée comme l'erreur du classifieur stochastique associée à ρ. Pour prédire l'étiquette d'une entrée $\mathbf{x} \in \mathbb{X}$, le classifieur stochastique *(i)* tire aléatoirement une hypothèse $h \in \mathbb{H}$ selon ρ, puis *(ii)* renvoie la valeur prédite $h(\mathbf{x})$. Le modèle d'intérêt n'est donc plus une hypothèse de $\mathbb{H}$, mais est ce classifieur stochastique, appelé classifieur de Gibbs en théorie PAC-Bayésienne (voir Section 2.4.1).

Définition 2.3.7 (Borne en généralisation PAC-Bayésienne)**.** Soit une mesure de l'écart en généralisation $\phi : [0,1]^2 \to \mathbb{R}$. Étant donné une distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, un ensemble d'hypothèses $\mathbb{H}$ et une fonction perte $\ell : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y}) \to [0,1]$, s'il existe une fonction $\Phi_{\mathrm{pb}} :]0,1] \to \mathbb{R}$, telle que pour tout $\delta \in]0,1]$ on a

$$\mathop{\mathbb{P}}_{S\sim\mathcal{D}^m}\left[\begin{array}{c}\text{Pour toute distribution } \rho \text{ sur } \mathbb{H},\\ \phi\left(\mathop{\mathbb{E}}_{h\sim\rho}\mathrm{R}^{\ell}_{\mathcal{D}}(h), \mathop{\mathbb{E}}_{h\sim\rho}\widehat{\mathrm{R}}^{\ell}_{S}(h)\right) \leq \Phi_{\mathrm{pb}}(\delta)\end{array}\right] \geq 1-\delta, \tag{2.5}$$

alors l'Équation (2.5) est une borne PAC-Bayésienne.

Un point clé est que la fonction $\Phi_{\text{pb}}(\delta)$ majore l'espérance des risques de toutes les hypothèses de $\mathbb{H}$ pondérés selon la distribution *posterior* ρ. Ainsi, contrairement aux bornes en convergence uniforme qui sont considérées comme des bornes en "pire cas", les bornes PAC-Bayésiennes permettent d'obtenir des bornes en "moyenne" sur l'ensemble $\mathbb{H}$.

Dans sa forme la plus simple, une borne en généralisation PAC-Bayésienne s'intéresse à l'écart en généralisation mesuré par $\phi(\mathbb{E}_{h\sim\rho}\, \mathrm{R}^{\ell}_{\mathcal{D}}(h), \mathbb{E}_{h\sim\rho}\, \widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h)) = \mathbb{E}_{h\sim\rho}\, \mathrm{R}^{\ell}_{\mathcal{D}}(h) - \mathbb{E}_{h\sim\rho}\, \widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h)$. Cette forme a été démontrée par Maurer (2004) et est rappelée dans le théorème suivant.

Théorème 2.3.6 (Borne PAC-Bayésienne de Maurer (2004))**.** Étant donné une distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, un ensemble d'hypothèses $\mathbb{H}$ et une fonction perte $\ell : \mathbb{H} \times (\mathbb{X}\times\mathbb{Y}) \to [0,1]$, pour toute distribution π sur $\mathbb{H}$ (définie *a priori*), pour tout $\delta \in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\begin{array}{l}\text{pour toute distribution } \rho \text{ on } \mathbb{H},\\[2ex] \mathop{\mathbb{E}}_{h\sim\rho} \mathrm{R}^{\ell}_{\mathcal{D}}(h) - \mathop{\mathbb{E}}_{h\sim\rho} \widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h) \leq \sqrt{\dfrac{1}{2m}\left[\mathrm{KL}(\rho\|\pi) + \ln\dfrac{2\sqrt{m}}{\delta}\right]}\end{array}\right] \geq 1-\delta,$$

où $\mathrm{KL}(\rho\|\pi) = \mathbb{E}_{h\sim\rho} \ln \frac{\rho(h)}{\pi(h)}$ est la divergence de Kullback-Leibler (KL) entre ρ et π. De manière équivalente, on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\begin{array}{l}\text{pour toute distribution } \rho \text{ on } \mathbb{H},\\[2ex] \mathop{\mathbb{E}}_{h\sim\rho} \mathrm{R}^{\ell}_{\mathcal{D}}(h) \leq \mathop{\mathbb{E}}_{h\sim\rho} \widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h) + \sqrt{\dfrac{1}{2m}\left[\mathrm{KL}(\rho\|\pi) + \ln\dfrac{2\sqrt{m}}{\delta}\right]}\end{array}\right] \geq 1-\delta.$$

Dans ce théorème, la borne dépend de la KL-divergence entre ρ et π, qui peut être vue comme mesure de complexité du classifieur stochastique. En effet, $\mathrm{KL}(\rho\|\pi)$ mesure à quel point la distribution ρ s'éloigne d'une distribution π qui a été fixée avant l'observation de l'ensemble d'apprentissage $\mathbb{S}\sim\mathcal{D}^m$, ou autrement dit, à quel point ρ dépend de l'ensemble $\mathbb{S}$. Plus la valeur $\mathrm{KL}(\rho\|\pi)$ est grande, plus les distributions ρ et π sont différentes.

2.4 La théorie PAC-Bayésienne en détails

Les bornes en généralisation PAC-Bayésiennes sont au cœur des contributions de ce manuscrit. Afin d'appréhender les notions nécessaires à leur compréhension, nous détaillons ces bornes ainsi que certaines notions clés qui gravitent autour.

2.4.1 Le vote de majorité PAC-Bayésien

Une caractéristique du classifieur stochastique de Gibbs selon une distribution ρ sur un ensemble d'hypothèses $\mathbb{H}$ est qu'il est étroitement lié au vote de majorité où chaque hypothèse de $\mathbb{H}$ est pondérée par sa probabilité selon ρ. Ce vote de majorité, parfois appelé classifieur de Bayes en théorie PAC-Bayésienne, est défini comme suit.

Définition 2.4.1 (Vote de majorité)**.** En classification binaire avec $\mathbb{Y} = \{-1, +1\}$, étant donné $\mathbb{H}$ un ensemble d'hypothèses $h : \mathbb{X} \to [-1, +1]$, aussi appelées *votants* en PAC-Bayes, et une distribution ρ sur $\mathbb{H}$, le vote de majorité est défini par

$$\forall \mathbf{x} \in \mathbb{X}, \quad \mathrm{MV}_\rho(\mathbf{x}) = \operatorname{sign}\left[\mathop{\mathbb{E}}_{h\sim\rho} h(\mathbf{x})\right].$$

En classification multiclasse avec $\mathbb{Y} = \{1, 2, \ldots, l\}$, étant donné $\mathbb{H}$ un ensemble d'hypothèses $h : \mathbb{X} \to \mathbb{Y}$ et une distribution ρ sur $\mathbb{H}$, le vote de majorité est défini par

$$\forall \mathbf{x} \in \mathbb{X}, \quad \mathrm{MV}_\rho(\mathbf{x}) = \operatorname*{argmax}_{y'\in\mathbb{Y}} \mathop{\mathbb{P}}_{h\sim\rho}(h(\mathbf{x}) = y') = \operatorname*{argmax}_{y'\in\mathbb{Y}} \mathop{\mathbb{E}}_{h\sim\rho} \mathrm{I}\left[h(\mathbf{x}) = y'\right].$$

Il est important de préciser que de nombreux modèles de classification peuvent s'exprimer comme un tel vote de majorité. C'est le cas, par exemple, des SVM (Cortes et Vapnik, 1995) où chaque votant dépend d'un exemple (Graepel et al., 2005), ou des réseaux de neurones (Kawaguchi et al., 2017). En outre, l'apprentissage de modèles basés sur un vote de majorité fait partie d'un type de méthodes d'apprentissage connues sous le nom de méthodes ensemblistes comme le *bagging* (Breiman, 1996), les forêts aléatoires (Breiman, 2001) ou le *boosting* (Freund et Schapire, 1996).

Étant donné une distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, l'objectif est alors d'apprendre un vote de majorité MV_ρ tel qu'il commette le moins d'erreur possible sur $\mathcal{D}$. Le risque $\mathrm{R}_\mathcal{D}(\mathrm{MV}_\rho)$ du vote de majorité, évalué avec la fonction perte 0-1, est défini comme suit.

Définition 2.4.2 (Risque du vote de majorité)**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble de votants $\mathbb{H}$, pour toute distribution ρ sur $\mathbb{H}$, le risque réel du vote de majorité est défini par

$$\mathrm{R}_\mathcal{D}(\mathrm{MV}_\rho) = \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} \mathrm{I}\left[\mathrm{MV}_\rho(\mathbf{x}) \neq y\right] = \mathop{\mathbb{P}}_{(\mathbf{x},y)\sim\mathcal{D}}\left(\mathrm{MV}_\rho(\mathbf{x}) \neq y\right).$$

Le risque empirique calculé sur un ensemble $\mathbb{S} = (\mathbf{x}_i, y_i) \sim \mathcal{D}^m$ est

$$\widehat{\mathrm{R}}_\mathbb{S}(\mathrm{MV}_\rho) = \frac{1}{m}\sum_{i=1}^{m} \mathrm{I}\left[\mathrm{MV}_\rho(\mathbf{x}_i) \neq y_i\right].$$

La marge du vote de majorité, définie ci-dessous, permet de mieux comprendre la décision du vote en capturant à quel point il se trompe.

Définition 2.4.3 (Marge du vote de majorité)**.** Pour tout ensemble de votants $\mathbb{H}$, pour toute distribution ρ sur $\mathbb{H}$, la marge du vote de majorité sur un exemple $(\mathbf{x}, y)$ est définie par

$$m_\rho(\mathbf{x}, y) = \mathop{\mathbb{P}}_{h\sim\rho}\left[h(\mathbf{x}) = y\right] - \max_{y'\in\mathbb{Y}, y'\neq y} \mathop{\mathbb{P}}_{h\sim\rho}\left[h(\mathbf{x}) = y'\right].$$

La marge est positive lorsque le score $\mathbb{P}_{h\sim\rho}\left[h(\mathbf{x}) = y\right]$ est supérieur au score associé aux autres étiquettes $y' \neq y$ et négative sinon. Elle capture si un exemple $(\mathbf{x}, y)$ est mal classé.

En effet, grâce à la marge, le risque du vote de majorité $\mathrm{R}_{\mathcal{D}}(\mathrm{MV}_\rho)$ se ré-écrit

$$\begin{aligned}\mathrm{R}_{\mathcal{D}}(\mathrm{MV}_\rho) &= \mathop{\mathbb{P}}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\mathop{\mathbb{P}}_{h\sim\rho}[h(\mathbf{x})=y] \leq \max_{y'\in\mathbb{Y},y'\neq y}\mathop{\mathbb{P}}_{h\sim\rho}[h(\mathbf{x})=y']\right]\\ &= \mathop{\mathbb{P}}_{(\mathbf{x},y)\sim\mathcal{D}}\left[m_\rho(\mathbf{x},y)\leq 0\right].\end{aligned}$$

Le risque correspond donc à la probabilité qu'un des scores soit supérieur à celui de la vraie étiquette. Cependant, la marge est non convexe par rapport à la distribution ρ (en raison de la fonction $\max$), ce qui peut rendre difficile la dérivation d'un algorithme d'apprentissage visant à optimiser la marge. Pour contourner ce problème dans le cadre PAC-Bayésien, nous avons proposé (Laviolette et al., 2017) de considérer une borne inférieure convexe sur la marge réelle appelée la $\frac{1}{2}$-*marge*.

Définition 2.4.4 (La $\frac{1}{2}$-marge du vote de majorité)**.** Pour tout ensemble de votants $\mathbb{H}$, pour toute distribution ρ sur $\mathbb{H}$, la $\frac{1}{2}$-marge est définie par

$$\widehat{m_\rho}(\mathbf{x},y) = 2\left[\mathop{\mathbb{P}}_{h\sim\rho}[h(\mathbf{x})=y] - \tfrac{1}{2}\right].$$

Lorsque le score $\mathbb{P}_{h\sim\rho}[h(\mathbf{x})=y]$ dépasse $\frac{1}{2}$, le vote de majorité classe correctement avec certitude l'exemple $(\mathbf{x},y)$. L'idée de cette relaxation est de calculer la différence entre le score et $\frac{1}{2}$: lorsque la marge est positive, l'exemple est correctement classé. La Figure 2.1 illustre ce comportement. En classification binaire, la $\frac{1}{2}$-marge se réduit à $\widehat{m_\rho}(\mathbf{x},y) = y\,\mathbb{E}_{h\sim\rho}\,h(\mathbf{x})$. À partir de la Définition 2.4.4, nous en déduisons la borne supérieure suivante sur le risque du vote de majorité

$$\mathrm{R}_{\mathcal{D}}(\mathrm{MV}_\rho) \leq \mathop{\mathbb{P}}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\mathop{\mathbb{P}}_{h\sim\rho}[h(\mathbf{x})=y]\leq\tfrac{1}{2}\right] = \mathop{\mathbb{P}}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\widehat{m_\rho}(\mathbf{x},y)\leq 0\right].$$

En général, le risque du vote de majorité, qui dépend de la perte 0-1, est difficile à optimiser et il est fréquent d'utiliser des relaxations du risque. Nous rappelons dans la section suivante, plusieurs relaxations du vote de majorité en PAC-Bayes.

2.4.2 Relaxations du risque du vote de majorité

Nous rappelons 3 bornes supérieures sur le risque du vote de majorité en PAC-Bayes : le risque de Gibbs (Langford et Shawe-Taylor, 2002 ; McAllester, 2003), l'erreur jointe (Lacasse et al., 2006 ; Germain et al., 2015 ; Masegosa et al., 2020) et la C-borne (Breiman, 2001 ; Lacasse et al., 2006 ; Roy et al., 2011).

2.4.2.1 Le risque de Gibbs, l'erreur jointe et le désaccord

Avant de présenter les bornes supérieures, nous rappelons une notion centrale en PAC-Bayes et étroitement lié au risque du vote de majorité : le risque de Gibbs. Ce dernier, évoqué précédemment, correspond en fait à la performance moyenne des votants intervenant dans le vote de majorité et est défini ci-dessous.

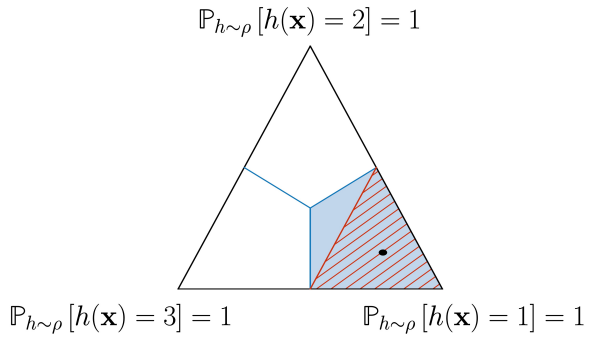


Figure 2.1. *Illustration de la marge du vote de majorité avec* 3 *classes,* $\mathbb{Y}=\{1,2,3\}$*. Le triangle représente la combinaison convexe des* 3 *scores où chaque sommet est le score maximal possible avec* $\forall i \in \mathbb{Y}$*,* $\mathbb{P}_{h\sim\rho}[h(\mathbf{x})=i]=1$*. Le point noir est la prédiction du vote sur un exemple* $(\mathbf{x},y)\in\mathbb{X}\times\mathbb{Y}$ *: les scores sont* $\mathbb{P}_{h\sim\rho}[h(\mathbf{x})=1]=0.7$ *et* $\mathbb{P}_{h\sim\rho}[h(\mathbf{x})=2]=\mathbb{P}_{h\sim\rho}[h(\mathbf{x})=3]=0.15$*. La zone bleue est la zone où le vote prédit la classe* y *pour* $\mathbf{x}$ *et où la marge* $m_\rho(\mathbf{x},y)$ *est positive. La zone rouge représente les prédictions où* $\mathbb{P}_{h\sim\rho}[h(\mathbf{x})=1]\geq\frac{1}{2}$ *(où la* $\frac{1}{2}$*-marge* $\widehat{m_\rho}(\mathbf{x},y)$ *est positive).*

Définition 2.4.5 (Risque de Gibbs)**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble de votants $\mathbb{H}$, pour toute distribution ρ sur $\mathbb{H}$, le risque de Gibbs est

$$\begin{aligned}\mathrm{R}_{\mathcal{D}}(G_\rho) &= \mathop{\mathbb{E}}_{h\sim\rho}\mathrm{R}_{\mathcal{D}}(h) = \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}}\mathop{\mathbb{E}}_{h\sim\rho}\mathrm{I}\left[h(\mathbf{x})\neq y\right] = \mathop{\mathbb{P}}_{(\mathbf{x},y)\sim\mathcal{D},h\sim\rho}\left[h(\mathbf{x})\neq y\right]\\ &= \frac{1}{2}\left[1-\mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}}\widehat{m_\rho}(\mathbf{x},y)\right], \qquad (2.6)\end{aligned}$$

où G_ρ est le classifieur de Gibbs.

Le risque de Gibbs est donc la moyenne pondérée par ρ des risques des votants, ce qui correspond exactement à la moyenne des erreurs qui intervient dans les bornes en généralisation PAC-Bayésiennes classiques (rappelées en Section 2.3.3). Autrement dit, les bornes PAC-Bayésiennes classiques sont des bornes en généralisation sur le risque de Gibbs. Il convient de rappeler que le classifieur de Gibbs agit comme un prédicteur stochastique en opposition au vote de majorité qui, lui, est un prédicteur déterministe.

De plus, Lacasse et al. (2006) ont démontré que le risque de Gibbs peut se réécrire comme la somme de deux quantités importantes : l'erreur jointe et la moitié du désaccord (rappelées ci-dessous). Ces deux notions prennent explicitement en compte la corrélation et la diversité entre les votants qui sont des quantités importantes pour apprendre une combinaison de votants efficace (Dietterich, 2000 ; Kuncheva, 2014).

Définition 2.4.6 (Erreur jointe et désaccord)**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble de votants $\mathbb{H}$, pour toute distribution ρ sur $\mathbb{H}$, l'erreur jointe associée à une distribution ρ est définie par

$$\begin{aligned}\mathrm{e}_{\mathcal{D}}(\rho) &= \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}}\mathop{\mathbb{E}}_{h\sim\rho}\mathop{\mathbb{E}}_{h'\sim\rho}\mathrm{I}\left[h(\mathbf{x})\neq y\right]\mathrm{I}\left[h'(\mathbf{x})\neq y\right]\\ &= \mathop{\mathbb{P}}_{(\mathbf{x},y)\sim\mathcal{D},h\sim\rho,h'\sim\rho}\left[h(\mathbf{x})\neq y, h'(\mathbf{x})\neq y\right]\\ &= \frac{1}{4}\left[1-2\mathop{\mathbb{E}}_{h\sim\rho}\widehat{m_\rho}(\mathbf{x},y)+\mathop{\mathbb{E}}_{h'\sim\rho}\widehat{m_\rho}(\mathbf{x},y)^2\right] \qquad (2.7)\end{aligned}$$

Le désaccord associé à une distribution ρ est définie par

$$\begin{aligned} \mathrm{d}_{\mathcal{D}}(\rho) &= 2 \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} \mathop{\mathbb{E}}_{h\sim\rho} \mathop{\mathbb{E}}_{h'\sim\rho} \mathrm{I}\left[h(\mathbf{x}) \neq y\right] \mathrm{I}\left[h'(\mathbf{x}) = y\right] \\ &= 2 \mathop{\mathbb{P}}_{(\mathbf{x},y)\sim\mathcal{D},h\sim\rho,h'\sim\rho} \left[h(\mathbf{x}) \neq y, h'(\mathbf{x}) = y\right] \\ &= \frac{1}{2}\left[1 - \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} \widehat{m_\rho}(\mathbf{x},y)^2\right]. \end{aligned} \tag{2.8}$$

À partir de ces définitions, Lacasse et al. (2006) ont démontré l'égalité suivante :

$$\mathrm{R}_{\mathcal{D}}(G_\rho) = \mathrm{e}_{\mathcal{D}}(\rho) + \frac{1}{2}\mathrm{d}_{\mathcal{D}}(\rho) \tag{2.9}$$

$$\iff \quad \mathrm{d}_{\mathcal{D}}(\rho) = 2\left[\mathrm{R}_{\mathcal{D}}(G_\rho) - \mathrm{e}_{\mathcal{D}}(\rho)\right]. \tag{2.10}$$

Cette égalité met en évidence des faits importants : *(i)* plus le risque de Gibbs est faible, plus le désaccord sera faible, et *(ii)* le désaccord augmente proportionnellement à la diminution de l'erreur jointe. En d'autres termes, dès lors que l'on considère plusieurs votants, la diversité des votants joue un rôle important dans la performance globale (que ce soit pour le vote de majorité ou pour le classifieur de Gibbs).

2.4.2.2 Trois bornes sur le risque du vote de majorité

Langford et Shawe-Taylor (2002) ont introduit la première relaxation du risque du vote de majorité en le majorant par 2 fois le risque de Gibbs.

Théorème 2.4.1 (Relaxation basée sur le risque de Gibbs)**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble de votants $\mathbb{H}$, pour toute distribution ρ sur $\mathbb{H}$, on a

$$\mathrm{R}_{\mathcal{D}}(\mathrm{MV}_\rho) \leq 2\,\mathrm{R}_{\mathcal{D}}(G_\rho). \tag{2.11}$$

Ce résultat est important : il implique qu'une borne en généralisation PAC-Bayésienne sur la moyenne des risques implique une borne en généralisation pour le vote de majorité. Malgré cette relation, le risque de Gibbs n'est pas suffisamment précis puisque dès que $\mathrm{R}_{\mathcal{D}}(G_\rho)$ dépasse $\frac{1}{2}$, la borne dépasse 1 et devient non informative.

Une manière de faire intervenir un peu plus précisément la diversité des votants est de faire appel à l'erreur jointe (Définition 2.4.6). Masegosa et al. (2020) ont ainsi démontré que l'erreur du vote de majorité peut être majorée par 4 fois l'erreur jointe.

Théorème 2.4.2 (Relaxation basée sur l'erreur jointe)**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble de votants $\mathbb{H}$, pour toute distribution ρ sur $\mathbb{H}$, on a

$$\mathrm{R}_{\mathcal{D}}(\mathrm{MV}_\rho) \leq 4e_{\mathcal{D}}(\rho). \tag{2.12}$$

Cette inégalité reflète l'idée que, pour qu'un vote de majorité soit performant, les votants doivent être suffisamment diversifiés et, s'ils commettent des erreurs, celles-ci doivent se produire sur des exemples différents. Or, si l'erreur jointe $\mathrm{e}_{\mathcal{D}}(\rho)$ dépasse $\frac{1}{4}$, la borne dépasse 1 et est non informative. Il est alors préférable, dans certains cas, de faire appel à une autre borne

appelée la C-borne en PAC-Bayes [4] (Breiman, 2001 ; Lacasse et al., 2006 ; Laviolette et al., 2017). La C-borne est une borne supérieure du risque du vote de majorité faisant intervenir à la fois la marge du vote et son second moment statistique, permettant de prendre en compte la diversité/complémentarité des votants (Morvant et al., 2014). Cette borne découle de l'inégalité de Cantelli-Chebyshev.

Théorème 2.4.3 (La C-borne)**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble de votants $\mathbb{H}$, pour toute distribution ρ sur $\mathbb{H}$, si

$$\mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} \widehat{m_\rho}(\mathbf{x},y) > 0 \iff \mathrm{R}_\mathcal{D}(G_\rho) < \tfrac{1}{2} \iff 2e_\mathcal{D}(\rho) + d_\mathcal{D}(\rho) < 1,$$

$$\text{alors on a} \quad \mathrm{R}_\mathcal{D}(\mathrm{MV}_\rho) \le 1 - \frac{\left(\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\widehat{m_\rho}(\mathbf{x},y)\right]\right)^2}{\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left(\widehat{m_\rho}(\mathbf{x},y)\right)^2} \tag{2.13}$$

$$= 1 - \frac{\left(1-2\mathrm{R}_\mathcal{D}(G_\rho)\right)^2}{1-2d_\mathcal{D}(\rho)} \tag{2.14}$$

$$= 1 - \frac{\left(1-\left[2e_\mathcal{D}(\rho)+d_\mathcal{D}(\rho)\right]\right)^2}{1-2d_\mathcal{D}(\rho)} \tag{2.15}$$

$$= C_\mathcal{D}(\rho).$$

Le théorème suivant énonce les relations d'ordre entre les relaxations du risque du vote de majorité, notamment dues à Germain et al. (2015) et Masegosa et al. (2020).

Théorème 2.4.4 (Relations entre les Théorèmes 2.4.1, 2.4.2 et 2.4.3)**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble de votants $\mathbb{H}$, pour toute distribution ρ sur $\mathbb{H}$, si $\mathrm{R}_\mathcal{D}(G_\rho) < \frac{1}{2}$, on a

(i) $\mathrm{R}_\mathcal{D}(\mathrm{MV}_\rho) \le C_\mathcal{D}(\rho) \le 4e_\mathcal{D}(\rho) \le 2\mathrm{R}_\mathcal{D}(G_\rho)$, si $\mathrm{R}_\mathcal{D}(G_\rho) \le d_\mathcal{D}(\rho)$,

(ii) $\mathrm{R}_\mathcal{D}(\mathrm{MV}_\rho) \le 2\mathrm{R}_\mathcal{D}(G_\rho) \le C_\mathcal{D}(\rho) \le 4e_\mathcal{D}(\rho)$, sinon.

En résumé, La C-borne est un bon compromis entre le risque de Gibbs et le désaccord. En effet, la C-borne $C_\mathcal{D}(\rho)$ est plus précise que $4\mathrm{e}_\mathcal{D}(\rho)$ dans tous les cas. D'ailleurs, lorsque $e_\mathcal{D}(\rho)$ est proche de $\frac{1}{4}$, alors la C-borne peut, quant à elle, être proche de 0 en fonction de la valeur du désaccord. De plus, quand $\mathrm{R}_\mathcal{D}(G_\rho) \le \mathrm{d}_\mathcal{D}(\rho)$, la C-borne est plus petite que $2\mathrm{R}_\mathcal{D}(G_\rho)$.

2.4.3 Bornes en généralisation PAC-Bayésienne

La théorie PAC-Bayésienne (Shawe-Taylor et Williamson, 1997 ; McAllester, 1999) a été motivée par l'apport de borne en généralisation de type PAC pour des approches s'inspirant des méthodes Bayésienne (voir Bishop, 2007, pour plus de détails sur l'inférence Bayésienne). Dans ce genre de méthodes, on suppose qu'on dispose d'une distribution définie *a priori* sur l'ensemble d'hypothèses $\mathbb{H}$, puis en utilisant le théorème de Bayes et l'ensemble d'apprentissage $\mathbb{S}$, on obtient une distribution *a posteriori* sur $\mathbb{H}$. Contrairement à l'inférence Bayésienne classique où la distribution *a posteriori* doit être proportionnelle au produit de la distribution *a priori* et de la vraisemblance des données, la théorie PAC-Bayésienne permet

4. Le terme "C-borne" est spécifique au PAC-Bayes et a été introduit par Lacasse et al. (2006).

de considérer une distribution *a priori* arbitraire appelée distribution *prior*. En fait, le terme "Bayésien" en théorie PAC-Bayésienne vient du fait que dans les résultats classiques, nous majorons l'écart en généralisation $\left|\mathbb{E}_{h\sim\rho_{\mathbb{S}}} \mathrm{R}^{\ell}_{\mathcal{D}}(h) - \mathbb{E}_{h\sim\rho_{\mathbb{S}}} \widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h)\right|$ où $h \in \mathbb{H}$ est échantillonné à partir d'une distribution dépendante des données $\rho_{\mathbb{S}}$ appelée la distribution *posterior*.

Pour définir plus formellement les bornes en généralisation PAC-Bayésiennes, nous donnons plus de détails sur la distribution *posterior* sur $\mathbb{H}$ notée $\rho_{\mathbb{S}}$ ou ρ et sur la distribution *prior* sur $\mathbb{H}$ notée π. Une distribution de probabilité ρ est définie par sa fonction de densité de probabilité $h \mapsto \rho(h)$ par rapport à une mesure de référence[5] sur $\mathbb{H}$; nous notons $\mathbb{M}(\mathbb{H})$ l'ensemble des fonctions de densité de probabilité sur $\mathbb{H}$. Ainsi, la distribution $\rho_{\mathbb{S}} \in \mathbb{M}(\mathbb{H})$ est la dérivée de Radon-Nikodym d'une mesure de probabilité par rapport à la mesure de référence. Nous notons également $\mathbb{M}^*(\mathbb{H}) \subseteq \mathbb{M}(\mathbb{H})$ l'ensemble des densités de probabilité strictement positives sur $\mathbb{H}$. Pour simplifier, nous supposons que le support du *posterior* $\rho_{\mathbb{S}}$ est inclus dans le support du *prior* π, *i.e.*, si $\pi(h) = 0$ alors $\rho_{\mathbb{S}}(h) = 0$ (continuité absolue) ; ainsi, nous avons $\pi \in \mathbb{M}^*(\mathbb{H})$.

Nous pouvons maintenant re-définir de manière un peu plus précise que la forme générale d'une borne en généralisation PAC-Bayésienne.

Définition 2.4.7 (Borne en généralisation PAC-Bayésienne)**.** Soit une mesure de l'écart en généralisation $\phi : [0,1]^2 \to \mathbb{R}$. Étant donné une distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, un ensemble d'hypothèses $\mathbb{H}$, une fonction perte $\ell : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y}) \to [0,1]$ et une distribution *prior* $\pi \in \mathbb{M}^*(\mathbb{H})$ sur $\mathbb{H}$, s'il existe une fonction $\Phi : \mathbb{M}(\mathbb{H})\times\mathbb{M}^*(\mathbb{H})\times]0,1] \to \mathbb{R}$, telle que pour tout $\delta \in]0,1]$ on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\forall \rho \in \mathbb{M}(\mathbb{H}),\ \phi\left(\mathop{\mathbb{E}}_{h\sim\rho_{\mathbb{S}}} \mathrm{R}^{\ell}_{\mathcal{D}}(h), \mathop{\mathbb{E}}_{h\sim\rho_{\mathbb{S}}} \widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h)\right) \leq \Phi\left(\rho_{\mathbb{S}}, \pi, \delta\right)\right] \geq 1-\delta, \tag{2.16}$$

alors l'Équation (2.16) est une borne PAC-Bayésienne.

La Définition 2.4.7 est une définition générale, car elle dépend d'une fonction $\phi()$ qui mesure l'écart en généralisation. Cet écart est majoré avec une probabilité d'au moins $1-\delta$ par une fonction $\Phi()$ qui dépend de δ et des distributions $\rho_{\mathbb{S}}$ et π. En général, plus δ est petit, plus la borne $\Phi()$ est grande, *i.e.*, la fonction $\Phi()$ est décroissante par rapport à δ. De plus, $\Phi()$ dépend de la distribution *posterior* $\rho_{\mathbb{S}} \in \mathbb{M}(\mathbb{H})$ qui, elle, dépend des données et de la distribution *prior* $\pi \in \mathbb{M}^*(\mathbb{H})$. La distribution *prior* ne dépend pas des données et peut encoder une connaissance *a priori*, *e.g.*, venant d'un expert ou d'un ensemble d'apprentissage supplémentaire différent de $\mathbb{S}$ (Parrado-Hernández et al., 2012b ; Dziugaite et al., 2021). Nous rappelons ci-dessous les instanciations du Théorème 2.4.7 les plus classiques de la littérature (*e.g.*, Seeger, 2002 ; McAllester, 2003 ; Catoni, 2007), associées chacune à une mesure $\phi()$ différente.

2.4.4 Borne PAC-Bayésienne générale de Germain et al.

Plusieurs bornes PAC-Bayésiennes classiques peuvent être englobées dans le théorème général suivant proposé par Germain et al. (2009).

5. Par exemple, si $\mathbb{H} = \mathbb{R}^d$, alors la mesure de référence est la mesure de Lebesgue.

Théorème 2.4.5 (Borne PAC-Bayésienne générale de GERMAIN et al. (2009))**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout espace d'hypothèses $\mathbb{H}$, pour toute distribution $\pi \in \mathbb{M}^*(\mathbb{H})$ sur $\mathbb{H}$, pour toute fonction mesurable $\varphi : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y})^m \to \mathbb{R}$, on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\forall \rho \in \mathbb{M}(\mathbb{H}),\ \mathop{\mathbb{E}}_{h\sim\rho}\varphi(h,\mathbb{S}) \le \mathrm{KL}(\rho\|\pi) + \ln\left(\frac{1}{\delta}\mathop{\mathbb{E}}_{\mathbb{S}'\sim\mathcal{D}^m}\mathop{\mathbb{E}}_{h'\sim\pi} e^{\varphi(h',\mathbb{S}')}\right)\right] \ge 1-\delta,$$

où $\mathrm{KL}(\rho\|\pi) = \mathop{\mathbb{E}}_{h\sim\rho}\ln\dfrac{\rho(h)}{\pi(h)}$ est la KL-divergence entre ρ et π.

Notons que cette borne est valide pour toute distribution *posterior* $\rho \in \mathbb{M}(\mathbb{H})$ dont la distribution *prior* π ou n'importe quelle distribution *posterior* $\rho_{\mathbb{S}}$ dépendante des données. De plus, cette borne est pénalisée par la KL-divergence entre ρ et π : plus ρ est proche de π, plus la divergence et donc la borne seront faibles. En outre, la borne est valide pour une fonction φ : $\mathbb{H}\times(\mathbb{X}\times\mathbb{Y})^m \to \mathbb{R}$ qui capture l'écart entre le risque réel $\mathrm{R}^{\ell}_{\mathcal{D}}(h)$ et le risque empirique $\widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h)$. Par exemple, avec $\varphi(h,\mathbb{S}) = m\,\phi(\mathrm{R}^{\ell}_{\mathcal{D}}(h), \widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h))$ où $\phi()$ est convexe, il est possible de retrouver la borne de la Définition 2.4.7. Comme nous le rappelons ci-dessous, en fixant $\phi()$, nous sommes en mesure de retrouver des bornes PAC-Bayésiennes classiques de la littérature.

2.4.4.1 Borne de la forme de McAllester

En fixant $\varphi(h,\mathbb{S}) = 2m\left[\mathrm{R}^{\ell}_{\mathcal{D}}(h) - \widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h)\right]^2$ dans le Théorème 2.4.5, on peut retrouver la borne du Théorème 2.3.6 (plus précise que la borne de MCALLESTER (2003, Th 1)).

Théorème 2.4.6 (Borne à la MCALLESTER (2003))**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout espace d'hypothèses $\mathbb{H}$, pour toute distribution $\pi \in \mathbb{M}^*(\mathbb{H})$, pour toute fonction perte $\ell : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y})^m \to [0,1]$, pour tout $\delta \in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\forall \rho \in \mathbb{M}(\mathbb{H}),\ \left|\mathop{\mathbb{E}}_{h\sim\rho}\mathrm{R}^{\ell}_{\mathcal{D}}(h) - \mathop{\mathbb{E}}_{h\sim\rho}\widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h)\right| \le \sqrt{\frac{1}{2m}\left[\mathrm{KL}(\rho\|\pi) + \ln\tfrac{2\sqrt{m}}{\delta}\right]}\right] \ge 1-\delta. \quad (2.17)$$

D'après ce théorème, l'écart $|\,\mathbb{E}_{h\sim\rho}\mathrm{R}^{\ell}_{\mathcal{D}}(h) - \mathbb{E}_{h\sim\rho}\widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h)|$ tend vers 0 lorsque le nombre d'exemples m augmente. En effet, plus il y a d'exemples, plus l'espérance des risques empiriques $\mathbb{E}_{h\sim\rho}\mathrm{R}^{\ell}_{\mathcal{D}}(h)$ est proche de l'espérance des risques réels $\mathbb{E}_{h\sim\rho}\widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h)$ quelle que soit la distribution $\rho \in \mathbb{M}(\mathbb{H})$. En fait, borner cet écart permet d'obtenir une borne supérieure et une borne inférieure sur l'espérance des risques réels : avec une probabilité d'au moins $1-\delta$ sur le choix aléatoire de $\mathbb{S}\sim\mathcal{D}^m$, on a

$$\forall \rho \in \mathbb{M}(\mathbb{H}),\quad \mathop{\mathbb{E}}_{h\sim\rho}\mathrm{R}^{\ell}_{\mathcal{D}}(h) \le \mathop{\mathbb{E}}_{h\sim\rho}\widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h) + \sqrt{\frac{1}{2m}\left[\mathrm{KL}(\rho\|\pi) + \ln\tfrac{2\sqrt{m}}{\delta}\right]}, \quad (2.18)$$

$$\text{et}\quad \mathop{\mathbb{E}}_{h\sim\rho}\mathrm{R}^{\ell}_{\mathcal{D}}(h) \ge \mathop{\mathbb{E}}_{h\sim\rho}\widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h) - \sqrt{\frac{1}{2m}\left[\mathrm{KL}(\rho\|\pi) + \ln\tfrac{2\sqrt{m}}{\delta}\right]}. \quad (2.19)$$

Si l'objectif d'apprentissage est de trouver $\rho\in\mathbb{M}(\mathbb{H})$ qui minimise l'espérance des risques réels $\mathbb{E}_{h\sim\rho}\mathrm{R}^{\ell}_{\mathcal{D}}(h)$, alors une solution consiste à trouver ρ qui minimise la borne via le problème de

minimisation suivant :

$$\min_{\rho\in\mathbb{M}(\mathbb{H})}\left\{\mathop{\mathbb{E}}_{h\sim\rho}\widehat{\mathsf{R}}_{\mathbb{S}}^{\ell}(h)+\sqrt{\frac{1}{2m}\Big[\mathrm{KL}(\rho\|\pi)+\ln\tfrac{2\sqrt{m}}{\delta}\Big]}\right\}.$$

Puisque la borne est valide pour tout $\rho\in\mathbb{M}(\mathbb{H})$ (avec grande probabilité), elle est également valide pour la solution optimale. Néanmoins, cette borne qui a l'intérêt d'être facilement interprétable, n'est pas la plus précise.

2.4.4.2 Borne de la forme de Catoni

Catoni (2007, Th. 1.2.1) a proposé une borne qui peut être plus précise que celle du Théorème 2.4.6 en utilisant un paramètre $c>0$. Cette borne, rappelée dans le Théorème 2.4.7, peut être retrouvée en instanciant le Théorème 2.4.5 avec $\varphi(h,\mathbb{S})=m\,\phi(\mathsf{R}_{\mathcal{D}}^{\ell}(h),\widehat{\mathsf{R}}_{\mathbb{S}}^{\ell}(h))$ où $\phi(\mathsf{R}_{\mathcal{D}}^{\ell}(h),\widehat{\mathsf{R}}_{\mathbb{S}}^{\ell}(h))=-\ln\Big(1-\big[1-e^{-c}\big]\,\mathsf{R}_{\mathcal{D}}^{\ell}(h)\Big)-c\widehat{\mathsf{R}}_{\mathbb{S}}^{\ell}(h)$.

Théorème 2.4.7 (Borne à la Catoni (2007))**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout espace d'hypothèses $\mathbb{H}$, pour toute distribution $\pi\in\mathbb{M}^*(\mathbb{H})$, pour toute fonction perte $\ell:\mathbb{H}\times(\mathbb{X}\times\mathbb{Y})^m\to[0,1]$, pour tout $c>0$, pour tout $\delta\in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\begin{array}{l}\forall\rho\in\mathbb{M}(\mathbb{H}),\\ -\ln\Big(1-\Big[1-e^{-c}\Big]\mathop{\mathbb{E}}_{h\sim\rho}\mathsf{R}_{\mathcal{D}}^{\ell}(h)\Big)-c\mathop{\mathbb{E}}_{h\sim\rho}\widehat{\mathsf{R}}_{\mathbb{S}}^{\ell}(h)\le\dfrac{1}{m}\Big[\mathrm{KL}(\rho\|\pi)+\ln\tfrac{1}{\delta}\Big]\end{array}\right]\ge 1-\delta. \tag{2.20}$$

Ce résultat est plus difficile à interpréter du fait de la mesure d'écart. En l'écrivant comme une borne supérieure sur l'espérance des risques réels $\mathbb{E}_{h\sim\rho}\mathsf{R}_{\mathcal{D}}^{\ell}(h)$, on a, avec une probabilité d'au moins $1-\delta$ sur $\mathbb{S}\sim\mathcal{D}^m$,

$$\forall\rho\in\mathbb{M}(\mathbb{H}),\ \mathop{\mathbb{E}}_{h\sim\rho}\mathsf{R}_{\mathcal{D}}^{\ell}(h)\le\frac{1}{1-e^{-c}}\left[1-\exp\left(-c\mathop{\mathbb{E}}_{h\sim\rho}\widehat{\mathsf{R}}_{\mathbb{S}}^{\ell}(h)-\frac{1}{m}\left[\mathrm{KL}(\rho\|\pi)+\ln\frac{1}{\delta}\right]\right)\right].$$

En d'autres termes, l'espérance des risques réels $\mathbb{E}_{h\sim\rho}\mathsf{R}_{\mathcal{D}}^{\ell}(h)$ est majorée par un compromis, contrôlé par c, entre l'espérance des risques empiriques $\mathbb{E}_{h\sim\rho}\widehat{\mathsf{R}}_{\mathbb{S}}^{\ell}(h)$ et le terme $\frac{1}{m}\Big[\mathrm{KL}(\rho\|\pi)+\ln\frac{1}{\delta}\Big]$. Ce résultat est en contraste avec les bornes des Équations (2.17), (2.18) et (2.19) puisque la présence du paramètre c permet d'ajuster la précision de la borne. En pratique, trouver la valeur de c est difficile, car la borne est valide avec une grande probabilité sur le choix de $\mathbb{S}\sim\mathcal{D}^m$ pour toutes les valeurs $c>0$. Une solution pour s'affranchir de cette contrainte est d'appliquer une borne de l'union pour obtenir une borne valide pour tout c qui appartient à un ensemble fini.

2.4.4.3 Borne de la forme de Seeger

Une des bornes les plus précises en PAC-Bayes (sans paramètre c) est la borne démontrée par Seeger (2002). Cette borne dépend de la KL-divergence entre deux distributions de Bernoulli définie comme suit.

Définition 2.4.8 (KL-divergence entre distributions de Bernoulli)**.** Pour toute distribution $q \in [0,1]$ et $p \in]0,1[$, la "petite" kl est définie par

$$\mathrm{kl}(q\|p) = \mathrm{KL}(\mathcal{B}(q)\|\mathcal{B}(p)) = q\ln\frac{q}{p} + (1-q)\ln\frac{1-q}{1-p},$$

où $\mathcal{B}(q)$ et $\mathcal{B}(p)$ sont des distributions de Bernoulli de biais respectivement q et p.

La première borne PAC-Bayésienne basée sur la divergence $\mathrm{kl}()$ a été démontrée par Seeger (2002). Nous rappelons ci-dessous une version plus précise de cette borne qui correspond à l'instanciation proposée par Germain et al. (2009) du Théorème 2.4.5 avec $\varphi(h,\mathbb{S}) = m\ \mathrm{kl}(\mathsf{R}^\ell_{\mathcal{D}}(h)\|\widehat{\mathsf{R}}^\ell_{\mathbb{S}}(h))$.

Théorème 2.4.8 (Borne à la Seeger (2002))**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout espace d'hypothèses $\mathbb{H}$, pour toute distribution $\pi \in \mathbb{M}^*(\mathbb{H})$, pour toute fonction perte $\ell : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y})^m \to [0,1]$, pour tout $\delta \in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\forall\rho\in\mathbb{M}(\mathbb{H}),\ \mathrm{kl}\Big(\mathop{\mathbb{E}}_{h\sim\rho}\mathsf{R}^\ell_{\mathcal{D}}(h)\Big\|\mathop{\mathbb{E}}_{h\sim\rho}\widehat{\mathsf{R}}^\ell_{\mathbb{S}}(h)\Big) \le \frac{1}{m}\Big[\mathrm{KL}(\rho\|\pi)+\ln\tfrac{2\sqrt{m}}{\delta}\Big]\right] \ge 1-\delta. \tag{2.21}$$

Cette borne majore l'écart entre $\mathbb{E}_{h\sim\rho}\mathsf{R}^\ell_{\mathcal{D}}(h)$ et $\mathbb{E}_{h\sim\rho}\widehat{\mathsf{R}}^\ell_{\mathbb{S}}(h)$ avec $\mathrm{kl}()$, rendant cette borne plus difficile à interpréter. Notons que l'inégalité de Pinsker, *i.e.*, $\forall(p,q)\in[0,1]^2$, $2(q-p)^2 \le \mathrm{kl}(q\|p)$, permet de retrouver la borne de Maurer (2004). En outre, Germain et al. (2009, Prop ? 2.1) et Lacasse (2010, Prop. 6.2.2) permettent de relier les Théorèmes 2.4.7 et 2.4.8 grâce à l'égalité suivante :

$$\max_{c>0}\Big\{-\ln\Big(1-\big[1-e^{-c}\big]p\Big)-c\,q\Big\} = \mathrm{kl}(q\|p).$$

En d'autres termes, étant donné $p \in]0,1[$ et $q \in [0,1]$, la valeur de $\mathrm{kl}(q\|p)$ coïncide avec la fonction $-\ln(1-[1-e^{-c}]p)-c\,q$ lorsque $c \ge 0$ est la valeur optimale. Telle quelle, l'Équation (2.21) du Théorème 2.4.8 ne permet pas de majorer ou minorer l'espérance des risques $\mathbb{E}_{h\sim\rho}\mathsf{R}^\ell_{\mathcal{D}}(h)$ contrairement aux bornes des Théorèmes 2.4.6 et 2.4.7. Il est cependant possible de faire appel aux fonctions $\overline{\mathrm{kl}}$ et $\underline{\mathrm{kl}}$ suivantes.

Définition 2.4.9 ($\overline{\mathrm{kl}}$ et $\underline{\mathrm{kl}}$)**.** Soit $\tau \ge 0$, pour tout $q \in [0,1]$, on définit

$$\overline{\mathrm{kl}}(q|\tau) = \max\Big\{p\in]0,1[\ \Big|\ \mathrm{kl}(q\|p)\le\tau\Big\},\ \text{et}\ \underline{\mathrm{kl}}(q|\tau) = \min\Big\{p\in]0,1[\ \Big|\ \mathrm{kl}(q\|p)\le\tau\Big\}.$$

La fonction $\overline{\mathrm{kl}}()$ (*resp.* $\underline{\mathrm{kl}}$) correspond à la valeur maximale (*resp.* minimale) $p \in]0,1[$ telle que l'inégalité $\mathrm{kl}(q\|p) \le \tau$ soit valide. Une approximation des valeurs associées à ces fonctions peut être calculée en utilisant l'inégalité de Pinsker. En effet, on a

$$\overline{\mathrm{kl}}(q|\tau) \le q + \sqrt{\frac{1}{2}\tau}, \qquad \text{et} \qquad q - \sqrt{\frac{1}{2}\tau} \le \underline{\mathrm{kl}}(q|\tau). \tag{2.22}$$

Pour calculer la valeur exacte de $\underline{\mathrm{kl}}()$ et de $\overline{\mathrm{kl}}()$, il faut résoudre deux problèmes d'optimisation. Reeb et al. (2018) ont proposé l'Algorithme 2.1 basé sur la méthode de la dichotomie : on affine itérativement l'intervalle $[p_{\mathsf{min}}, p_{\mathsf{max}}]$ auquel appartient $p \in]0,1[$. Si l'égalité $\mathrm{kl}(q\|p) = \tau$ est atteinte ou si l'intervalle $[p_{\mathsf{min}}, p_{\mathsf{max}}]$ est suffisamment petit, la valeur de p est la valeur courante.

Algorithme 2.1 Calcul de $\overline{\mathrm{kl}}(q|\tau)$ *resp.* $\underline{\mathrm{kl}}(q|\tau)$

Entrées : $q \in [0,1]$ (le risque empirique), la valeur $\tau \geq 0$, seuil de tolérance ϵ, nombre maximum d'itérations T_{max}

1: $p_{\mathsf{max}} \leftarrow 1$ et $p_{\mathsf{min}} \leftarrow q$ (*resp.* $p_{\mathsf{max}} \leftarrow q$ et $p_{\mathsf{min}} \leftarrow 0$)
2: **pour** $t \leftarrow 1$ à T_{max} **faire**
3: $\quad p = \frac{1}{2}\left[p_{\mathsf{min}} + p_{\mathsf{max}}\right]$
4: $\quad$ **if** $\mathrm{kl}(q\|p) = \tau$ **ou** $(p_{\mathsf{min}} - p_{\mathsf{max}}) < \epsilon$ **alors retourner** p
5: $\quad$ **if** $\mathrm{kl}(q\|p) > \tau$ **alors** $p_{\mathsf{max}} = p$ (*resp.* $p_{\mathsf{min}} = p$)
6: $\quad$ **if** $\mathrm{kl}(q\|p) < \tau$ **alors** $p_{\mathsf{min}} = p$ (*resp.* $p_{\mathsf{max}} = p$)
7: **retourner** p

De plus, Reeb et al. (2018) ont démontré l'expression des dérivées par rapport à q et τ :

$$\frac{\partial \mathrm{k}(q|\psi)}{\partial q} = \frac{\ln \frac{1-q}{1-\mathrm{k}(q|\psi)} - \ln \frac{q}{\mathrm{k}(q|\psi)}}{\frac{1-q}{1-\mathrm{k}(q|\psi)} - \frac{q}{\mathrm{k}(q|\psi)}}, \quad \text{et} \quad \frac{\partial \mathrm{k}(q|\psi)}{\partial \psi} = \frac{1}{\frac{1-q}{1-\mathrm{k}(q|\psi)} - \frac{q}{\mathrm{k}(q|\psi)}}, \tag{2.23}$$

où $\mathrm{k}()$ est soit $\underline{\mathrm{kl}}()$, soit $\overline{\mathrm{kl}}()$. Ces dérivés leur ont permis d'obtenir des algorithmes de minimisation de bornes basés sur la descente de gradient.

La Définition 2.4.9 permet de réécrire la borne du Théorème 2.4.8 pour majorer l'espérance des risques $\mathbb{E}_{h\sim\rho}\, \mathsf{R}^{\ell}_{\mathcal{D}}(h)$. Avec une probabilité d'au moins $1-\delta$ sur le choix aléatoire de $\mathbb{S} \sim \mathcal{D}^m$, on a pour tout $\rho \in \mathbb{M}(\mathbb{H})$

$$\mathop{\mathbb{E}}_{h\sim\rho} \mathsf{R}^{\ell}_{\mathcal{D}}(h) \leq \overline{\mathrm{kl}}\left(\mathop{\mathbb{E}}_{h\sim\rho} \widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h) \;\middle|\; \frac{1}{m}\Big[\mathrm{KL}(\rho\|\pi) + \ln \tfrac{2\sqrt{m}}{\delta}\Big] \right) \tag{2.24}$$

$$\text{et} \quad \mathop{\mathbb{E}}_{h\sim\rho} \mathsf{R}^{\ell}_{\mathcal{D}}(h) \geq \underline{\mathrm{kl}}\left(\mathop{\mathbb{E}}_{h\sim\rho} \widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h) \;\middle|\; \frac{1}{m}\Big[\mathrm{KL}(\rho\|\pi) + \ln \tfrac{2\sqrt{m}}{\delta}\Big] \right). \tag{2.25}$$

L'approximation de l'Équation (2.22) permet de prouver que la borne du Théorème 2.4.8 est plus précise que celle du Théorème 2.4.6. En effet, en appliquant l'Équation (2.22) aux Équations (2.24) et (2.25), on peut retrouver les Équations (2.18) et (2.19).

2.4.5 Borne PAC-Bayésienne générale de Bégin et al.

La KL-divergence $\mathrm{KL}(\rho\|\pi)$ entre les distributions *posterior* et *prior* est omniprésente dans les bornes PAC-Bayes. En fait, $\mathrm{KL}(\rho\|\pi)$ quantifie la différence entre les termes $\mathbb{E}_{h\sim\rho}\, \varphi(h, \mathbb{S})$ et $\mathbb{E}_{h\sim\pi} \exp\left[\varphi(h, \mathbb{S})\right]$ et fait apparaître le terme constant $\mathbb{E}_{\mathbb{S}'}\, \mathbb{E}_{h'\sim\pi}\, e^{\varphi(h', \mathbb{S}')}$ (Donsker et Varadhan, 1976). Plus concrètement, l'outil utilisé pour quantifier la différence entre ces deux espérances est appelé "inégalité de changement de mesure" *(change of measure inequality)*. Dans le cas classique, cette inégalité est la suivante.

Proposition 2.4.1 (Représentation variationnelle de Donsker-Varadhan)**.** Pour tout ensemble d'hypothèses $\mathbb{H}$, pour toute distribution $\pi \in \mathbb{M}^*(\mathbb{H})$, pour toute fonction mesurable $\varphi : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y})^m \to \mathbb{R}$ telle que $\mathbb{E}_{h'\sim\pi}\, e^{\varphi(h',\mathbb{S})} < +\infty$, on a

$$\forall \mathbb{S} \in (\mathbb{X}\times\mathbb{Y})^m, \quad \forall \rho \in \mathbb{M}(\mathbb{H}), \qquad \mathop{\mathbb{E}}_{h\sim\rho} \varphi(h,\mathbb{S}) - \ln\left(\mathop{\mathbb{E}}_{h\sim\pi} e^{\varphi(h,\mathbb{S})}\right) \leq \mathrm{KL}(\rho\|\pi)$$
$$\iff \mathop{\mathbb{E}}_{h\sim\rho} \varphi(h,\mathbb{S}) \leq \mathrm{KL}(\rho\|\pi) + \ln\left(\mathop{\mathbb{E}}_{h\sim\pi} e^{\varphi(h,\mathbb{S})}\right).$$

Si la distribution ρ est définie par $\rho(h) = \pi(h)\frac{e^{\varphi(h,\mathbb{S})}}{\mathbb{E}_{h'\sim\pi}\, e^{\varphi(h',\mathbb{S})}}$, on a

$$\forall \mathbb{S} \in (\mathbb{X}\times\mathbb{Y})^m, \qquad \mathop{\mathbb{E}}_{h\sim\rho} \varphi(h,\mathbb{S}) - \ln\left(\mathop{\mathbb{E}}_{h\sim\pi} e^{\varphi(h,\mathbb{S})}\right) = \mathrm{KL}(\rho\|\pi),$$
$$\iff \mathop{\mathbb{E}}_{h\sim\rho} \varphi(h,\mathbb{S}) = \mathrm{KL}(\rho\|\pi) + \ln\left(\mathop{\mathbb{E}}_{h\sim\pi} e^{\varphi(h,\mathbb{S})}\right).$$

Nous remarquons que cette inégalité ressemble à la borne générale de Germain et al. (2009) (du Théorème 2.4.5), faisant apparaître le terme constant $\mathbb{E}_{\mathbb{S}'\sim\mathcal{D}^m}\mathbb{E}_{h'\sim\pi}\, e^{\varphi(h',\mathbb{S}')}$. En considérant d'autres divergences, il est possible d'obtenir d'autres termes constants. Par exemple, Bégin et al. (2016) ont dérivé la borne suivante, qui est une borne PAC-Bayésienne générale avec la divergence de Rényi définie par $\forall\lambda>1,\ \mathrm{D}_\lambda(\rho\|\pi) = \frac{1}{\lambda-1}\ln\left(\mathbb{E}_{h\sim\pi}\left[\frac{\rho(h)}{\pi(h)}\right]^\lambda\right)$.

Théorème 2.4.9 (Borne PAC-Bayésienne générale de Bégin et al. (2016))**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble d'hypothèses $\mathbb{H}$, pour toute distribution $\pi \in \mathbb{M}^*(\mathbb{H})$, pour toute fonction mesurable $\varphi : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y})^m \to \mathbb{R}_*^+$, pour tout $\lambda > 1$, pour tout $\delta \in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\begin{array}{l}\forall\rho\in\mathbb{M}(\mathbb{H}),\\ \dfrac{\lambda}{\lambda-1}\ln\left(\mathop{\mathbb{E}}_{h\sim\rho}\varphi(h,\mathbb{S})\right) \leq \mathrm{D}_\lambda(\rho\|\pi) + \ln\left(\dfrac{1}{\delta}\mathop{\mathbb{E}}_{\mathbb{S}'\sim\mathcal{D}^m}\mathop{\mathbb{E}}_{h'\sim\pi}\varphi(h',\mathbb{S}')^{\frac{\lambda}{\lambda-1}}\right)\end{array}\right] \geq 1-\delta.$$

La fonction $\varphi(h,\mathbb{S}) \mapsto \frac{\lambda}{\lambda-1}\ln\left[\mathbb{E}_{h\sim\rho}\varphi(h,\mathbb{S})\right]$ représente l'écart en généralisation et est majorée par la divergence de Rényi. Bien que les Théorèmes 2.4.9 et 2.4.5 semblent différents, ils sont reliés : en remplaçant $\varphi(h,\mathbb{S})$ par $\exp\left(\frac{\lambda-1}{\lambda}\varphi(h,\mathbb{S})\right)$ et en appliquant l'inégalité de Jensen à la partie gauche de la borne, on obtient

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\forall\rho\in\mathbb{M}(\mathbb{H}), \quad \mathop{\mathbb{E}}_{h\sim\rho}\varphi(h,\mathbb{S}) \leq \mathrm{D}_\lambda(\rho\|\pi) + \ln\left(\frac{1}{\delta}\mathop{\mathbb{E}}_{\mathbb{S}'\sim\mathcal{D}^m}\mathop{\mathbb{E}}_{h'\sim\pi} e^{\varphi(h',\mathbb{S}')}\right)\right] \geq 1-\delta.$$

Cette borne est un peu moins précise que celle du Théorème 2.4.5 : pour tout $\lambda > 1$ et pour toutes distributions ρ et π, on a $\mathrm{KL}(\rho\|\pi) \leq \mathrm{D}_\lambda(\rho\|\pi)$ et $\lim_{\lambda\to1^+}\mathrm{D}_\lambda(\rho\|\pi) = \mathrm{KL}(\rho\|\pi)$ (Erven et Harremoës, 2014). Comme pour le Théorème 2.4.5, fixer la fonction $\varphi()$ et majorer $\mathbb{E}_{\mathbb{S}\sim\mathcal{D}^m}\mathbb{E}_{h\sim\pi}\,\varphi(h,\mathbb{S})^{\frac{\lambda}{\lambda-1}}$ permet d'obtenir une borne calculable. Par exemple, avec le Théorème 2.4.9 on peut retrouver des bornes à la McAllester, à la Catoni ou à la Seeger basée sur la divergence de Rényi.

D'un point de vue général, les bornes en généralisation PAC-Bayésiennes s'intéressent donc à l'espérance de $\varphi()$ selon la distribution *posterior* ρ. Comme mentionnée en Section 2.3.3, cela produit des bornes en espérance sur l'ensemble d'hypothèses $\mathbb{H}$ (ou pour le vote de

majorité sur $\mathbb{H}$), au lieu de bornes valables pour une hypothèse de l'ensemble $\mathbb{H}$. Les bornes PAC-Bayésiennes désintégrées énoncées dans la section suivante ont pour but de trouver une majoration de $\varphi()$ pour une unique hypothèse $h \in \mathbb{H}$.

2.5 Bornes PAC-Bayésiennes désintégrées

Les bornes PAC-Bayésiennes sont de nature stochastique sur $\mathbb{H}$ et l'obtention de bornes pour une hypothèse de $\mathbb{H}$ n'est pas une tâche simple. Sous certaines conditions, il est possible d'obtenir de telles bornes (*e.g.*, Langford et Shawe-Taylor, 2002 ; Langford, 2005 ; Germain et al., 2009). La solution qui nous intéresse consiste à tirer une hypothèse $h \in \mathbb{H}$ selon la distribution *posterior* $\rho_{\mathbb{S}} \in \mathbb{M}(\mathbb{H})$ pour obtenir une borne en généralisation pour cette hypothèse. Cette astuce permet de majorer l'écart $\left|\mathrm{R}_{\mathcal{D}}^{\ell}(h) - \widehat{\mathrm{R}}_{\mathbb{S}}^{\ell}(h)\right|$ de la manière suivante.

Définition 2.5.1 (Borne en généralisation PAC-Bayésienne désintégrée)**.** Soit une mesure de l'écart en généralisation $\phi : [0,1]^2 \to [0,1]$. Une borne PAC-Bayésienne désintégrée est définie telle que pour toute distribution $\mathcal{D}$ sur $\mathbb{X} \times \mathbb{Y}$, pour tout ensemble d'hypothèses $\mathbb{H}$ avec $\ell : \mathbb{H} \times (\mathbb{X} \times \mathbb{Y}) \to [0,1]$ une fonction perte, pour toute distribution *prior* $\pi \in \mathbb{M}^*(\mathbb{H})$, pour tout algorithme $A : (\mathbb{X} \times \mathbb{Y})^m \times \mathbb{M}^*(\mathbb{H}) \to \mathbb{M}(\mathbb{H})$, il existe une fonction $\Phi : \mathbb{M}(\mathbb{H}) \times \mathbb{M}^*(\mathbb{H}) \times]0,1] \to \mathbb{R}$ telle que pour tout $\delta \in]0,1]$ on a

$$\mathop{\mathbb{P}}_{\substack{\mathbb{S}\sim\mathcal{D}^m \\ h\sim\rho_{\mathbb{S}}}} \left[\phi(\mathrm{R}_{\mathcal{D}}^{\ell}(h), \widehat{\mathrm{R}}_{\mathbb{S}}^{\ell}(h)) \;\leq\; \Phi\Big(\rho_{\mathbb{S}}, \pi, \delta\Big) \right] \;\geq\; 1 - \delta,$$

où $\rho_{\mathbb{S}} = A(\mathbb{S}, \pi)$ est la sortie de l'algorithme déterministe A et $\phi()$ est, par exemple, $\phi(\mathrm{R}_{\mathcal{D}}^{\ell}(h), \widehat{\mathrm{R}}_{\mathbb{S}}^{\ell}(h)) = |\mathrm{R}_{\mathcal{D}}^{\ell}(h) - \widehat{\mathrm{R}}_{\mathbb{S}}^{\ell}(h)|$.

Le point important à remarquer en comparaison des bornes PAC-Bayésiennes de la section précédente est que l'espérance $\mathbb{E}_{h\sim\rho_{\mathbb{S}}}[\cdot]$ est à l'extérieur de la fonction indicatrice : c'est ce que l'on appelle la "désintégration" des bornes PAC-Bayes. De plus, la distribution *posterior* $\rho_{\mathbb{S}}$ est ici obtenue grâce à un algorithme qui dépend de la distribution *prior* $\pi \in \mathbb{M}^*(\mathbb{H})$ et de l'ensemble d'apprentissage $\mathbb{S}$. Ce type de borne a été introduit par Catoni (2007, Th. 1.2.7) et Blanchard et Fleuret (2007).

2.5.1 Borne désintégrée générale de Rivasplata et al.

Comme pour les bornes classiques, il existe une forme générale des bornes désintégrées. Le théorème ci-dessous a été proposé par Rivasplata et al. (2020, Th. 1(i)).

Théorème 2.5.1 (Borne désintégrée générale de Rivasplata et al. (2020))**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X} \times \mathbb{Y}$, pour tout ensemble d'hypothèses $\mathbb{H}$, pour toute distribution *prior* $\pi \in \mathbb{M}^*(\mathbb{H})$, pour toute fonction mesurable $\varphi : \mathbb{H} \times (\mathbb{X} \times \mathbb{Y})^m \to \mathbb{R}$, pour tout $\delta \in]0,1]$, pour tout algorithme $A : (\mathbb{X} \times \mathbb{Y})^m \times \mathbb{M}^*(\mathbb{H}) \to \mathbb{M}(\mathbb{H})$, on a

$$\mathop{\mathbb{P}}_{\substack{\mathbb{S}\sim\mathcal{D}^m \\ h\sim\rho_{\mathbb{S}}}} \left[\varphi(h, \mathbb{S}) \leq \underbrace{\ln\left[\frac{\rho_{\mathbb{S}}(h)}{\pi(h)}\right] + \ln\left[\frac{1}{\delta} \mathop{\mathbb{E}}_{\mathbb{S}'\sim\mathcal{D}^m} \mathop{\mathbb{E}}_{h'\sim\pi} \exp\left(\varphi(h', \mathbb{S}')\right)\right]}_{\Phi(\rho_{\mathbb{S}}, \pi, \delta)} \right] \geq 1 - \delta,$$

où $\rho_{\mathbb{S}} = A(\mathbb{S}, \pi)$ est la sortie de l'algorithme déterministe A.

Cette borne est valide avec grande probabilité sur le choix aléatoire non seulement de l'ensemble d'apprentissage $\mathbb{S}\sim\mathcal{D}^m$ mais également de $h\sim\rho_{\mathbb{S}}$. De plus, au lieu de faire intervenir la KL-divergence, cette borne dépend d'une version "désintégrée" de la KL-divergence : $\ln\frac{\rho_{\mathbb{S}}(h)}{\pi(h)}$. Ce terme est le rapport logarithmique de la densité de la distribution *posterior* $\rho_{\mathbb{S}}(h)$ et de la distribution *prior* $\pi(h)$ pour l'hypothèse tirée $h\sim\rho_{\mathbb{S}}$. Intuitivement, plus la densité *posterior* $\rho_{\mathbb{S}}(h)$ est proche de la densité *prior* $\pi(h)$ pour $h\sim\rho_{\mathbb{S}}$, plus la KL-divergence désintégrée sera faible. Comme pour la borne générale de Germain et al. (2009), il faut fixer $\varphi()$ puis de majorer le terme $\mathbb{E}_{\mathbb{S}'\sim\mathcal{D}^m}\mathbb{E}_{h'\sim\pi}\,\mathrm{e}^{\varphi(h',\mathbb{S}')}$ pour obtenir une borne calculable. Ce théorème général permet de retrouver la borne dérivée par Catoni (2007, Th. 1.2.7).

2.5.2 Borne désintégrée de Catoni

La borne de Catoni (2007, Th. 1.2.7), rappelée ci-dessous, est l'une des premières bornes désintégrées introduites dans la littérature.

Théorème 2.5.2 (Borne désintégrée de Catoni (2007))**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble d'hypothèses $\mathbb{H}$, pour toute distribution *prior* $\pi\in\mathbb{M}^*(\mathbb{H})$, pour toute fonction perte, $\ell:\mathbb{H}\times(\mathbb{X}\times\mathbb{Y})^m\to[0,1]$, pour tout $c>0$, pour tout $\delta\in]0,1]$, pour tout algorithme $A:(\mathbb{X}\times\mathbb{Y})^m\times\mathbb{M}^*(\mathbb{H})\to\mathbb{M}(\mathbb{H})$, on a

$$\mathop{\mathbb{P}}_{\substack{\mathbb{S}\sim\mathcal{D}^m\\ h\sim\rho_{\mathbb{S}}}}\left[-\ln\Big(1-[1-e^{-c}]\mathop{\mathbb{E}}_{h\sim\rho}\mathsf{R}_{\mathcal{D}}^{\ell}(h)\Big)-c\mathop{\mathbb{E}}_{h\sim\rho}\widehat{\mathsf{R}}_{\mathbb{S}}^{\ell}(h)\le\frac{1}{m}\left[\ln\frac{\rho_{\mathbb{S}}(h)}{\pi(h)}+\ln\frac{1}{\delta}\right]\right]\ge1-\delta,$$

où $\rho_{\mathbb{S}}=A(\mathbb{S},\pi)$ est la sortie de l'algorithme déterministe A.

Après le tirage de l'ensemble d'apprentissage $\mathbb{S}\sim\mathcal{D}^m$ et de l'hypothèse $h\sim\rho_{\mathbb{S}}$, la borne obtenue est similaire à la borne PAC-Bayésienne classique du Théorème 2.4.7 : seule la KL-divergence est remplacée par sa version désintégrée. D'autres mesures d'écart entre le risque réel $\mathsf{R}_{\mathcal{D}}^{\ell}(h)$ et le risque empirique $\widehat{\mathsf{R}}_{\mathbb{S}}^{\ell}(\rho)$ peuvent être considérées. Par exemple, Blanchard et Fleuret (2007) ont proposé une borne avec $\mathrm{kl}(\widehat{\mathsf{R}}_{\mathbb{S}}^{\ell}(h)\|\mathsf{R}_{\mathcal{D}}^{\ell}(h))$.

2.5.3 Borne désintégrée de Blanchard et Fleuret

La technique de preuve suivie par Blanchard et Fleuret (2007) est différente et se base sur une technique appelée *Occam's hammer*. Par exemple, ils ont prouvé la borne PAC-Bayésienne "à la Seeger (2002)" suivante.

Théorème 2.5.3 (Borne désintégrée de Blanchard et Fleuret (2007))**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble d'hypothèses $\mathbb{H}$, pour toute distribution *prior* $\pi\in\mathbb{M}^*(\mathbb{H})$, pour toute fonction perte $\ell:\mathbb{H}\times(\mathbb{X}\times\mathbb{Y})^m\to[0,1]$, pour tout $k>0$, pour tout $\delta\in]0,1]$, pour tout algorithme $A:(\mathbb{X}\times\mathbb{Y})^m\times\mathbb{M}^*(\mathbb{H})\to\mathbb{M}(\mathbb{H})$, on a

$$\mathop{\mathbb{P}}_{\substack{\mathbb{S}\sim\mathcal{D}^m\\ h\sim\rho_{\mathbb{S}}}}\left[\mathrm{kl}_+(\widehat{\mathsf{R}}_{\mathbb{S}}^{\ell}(h)\|\mathsf{R}_{\mathcal{D}}^{\ell}(h))\le\frac{1}{m}\left[\ln\frac{k+1}{\delta}+\left(1+\frac{1}{k}\right)\ln_+\frac{\rho_{\mathbb{S}}(h)}{\pi(h)}\right]\right]\ge1-\delta,$$

où $\rho_{\mathbb{S}}=A(\mathbb{S},\pi)$ est la sortie de l'algorithme A et $\ln_+(\cdot)=\max(\ln(\cdot),0)$,
et $\mathrm{kl}_+(\widehat{\mathsf{R}}_{\mathbb{S}}^{\ell}(h)\|\mathsf{R}_{\mathcal{D}}^{\ell}(h))=\mathrm{kl}(\widehat{\mathsf{R}}_{\mathbb{S}}^{\ell}(h)\|\mathsf{R}_{\mathcal{D}}^{\ell}(h))$ si $\widehat{\mathsf{R}}_{\mathbb{S}}^{\ell}(h)<\mathsf{R}_{\mathcal{D}}^{\ell}(h)$ ou 0 sinon.

Comme pour la borne de Catoni (2007), cette borne est paramétrée et sa précision dépend du paramètre $k > 1$. Cependant, le paramètre optimal dépend du rapport logarithmique $\ln_+ \frac{\rho_{\mathbb{S}}(h)}{\pi(h)}$ et ne peut pas être fixé l'ensemble d'apprentissage tiré $\mathbb{S} \sim \mathcal{D}^m$ n'est pas connu.

2.6 Conclusion

Ce chapitre introduit les bornes en généralisation avec un focus sur les bornes PAC-Bayésiennes classiques. Ces bornes permettent d'obtenir des garanties théoriques, en particulier pour les modèles pouvant s'exprimer comme un vote de majorité pondéré. Comme les contributions de ce manuscrit l'illustrent, ces bornes sont particulièrement utiles pour dériver des algorithmes d'apprentissage garantissant que le modèle n'est pas trop sensible au sur-apprentissage. Par exemple, dans le Chapitre 6 et dans le Chapitre 7, nous présentons des algorithmes dits auto-certifiés qui permettent d'optimiser directement une borne PAC-Bayésienne pour minimiser le risque du vote de majorité respectivement dans le cadre de la classification et dans le cadre de la robustesse adversaire. Dans le cas où la minimisation directe des bornes est plus difficile, il est également possible d'utiliser les bornes PAC-Bayésiennes comme source d'inspiration pour développer de nouvelles méthodes d'apprentissage fondée théoriquement, comme c'est le cas des contributions du Chapitre 4 (pour l'adaptation de domaine), du Chapitre 5 (où les votants sont exprimés comme des *Random Fourier Features*) ou du Chapitre 6 (pour l'apprentissage multi-vues).

Bien que les bornes PAC-Bayésiennes aient un intérêt pratique certain, leur principal inconvénient est que nous bornons $\mathbb{E}_{h\sim\rho}\,\varphi(h, \mathbb{S})$ au lieu de $\varphi(h, \mathbb{S})$. En revanche, les bornes désintégrées permettent de borner le terme $\varphi(h, \mathbb{S})$, ce qui est plus pertinent si l'on souhaite travailler avec une hypothèse unique $h \sim \rho$. Le Chapitre 8 illustre le potentiel de ces bornes en énonçant une application pratique de ces bornes, notamment avec des modèles surparamétrés ; cela mène également à la dérivation de nouvelles bornes désintégrées plus adaptées à l'optimisation. Enfin, le Chapitre 9 introduit de nouvelles perspectives, basées sur ces bornes, pour obtenir des bornes en généralisation qui ne dépendent pas des mesures classiques de complexité telles que la dimension VC ou la complexité de Rademacher, mais qui dépendent de mesure de complexité pouvant être définies par l'utilisateur en fonction de la tâche considérée.

Deuxième partie

Les bornes PAC-Bayésiennes comme source d'inspiration d'algorithmes d'apprentissage

Apprentissage multi-vues PAC-Bayésien 3

Contexte

Ce chapitre étend les bornes "historiques" PAC-Bayésiennes des Sections 2.4.4.1 à 2.4.4.3 au cadre de l'apprentissage multi-vues avec plus de 2 vues. En particulier, il présente une partie des travaux réalisés lors de la thèse de Anil Goyal sur l'apprentissage multi-vues à deux niveaux (parfois appelé fusion tardive) avec plus de 2 vues dans le cadre théorique PAC-Bayésien. Ces travaux ont été publiés à ECML-PKDD 2017 (Goyal et al., 2017) et dans le journal Neurocomputing (Goyal et al., 2019) et font suite à une partie de mes travaux de thèse et de post-doctorat qui ont donné lieu à une publication dans le workshop S+SSPR (Morvant et al., 2014).

3.1 Introduction

Avec l'explosion des données disponibles, il est courant que les observations soient décrites à travers plusieurs vues ou modalités. Dans ce chapitre, nous étudions le problème d'apprentissage d'un classifieur binaire à partir de plusieurs sources d'information décrivant les observations. Cette problématique est appelée apprentissage multi-vues ou multimodal (Atrey et al., 2010 ; Sun et al., 2019). L'objectif est de proposer un critère théoriquement fondé pour "combiner correctement" différentes vues tout en prenant en compte la diversité/complémentarité entre ces vues. Généralement, cela se fait soit par concaténation directe des représentations (*early fusion*), soit par combinaison des prédictions des classifieurs spécifiques à chaque vue (*late fusion*) (Snoek et al., 2005 ; Morvant et al., 2014). Ici, nous nous plaçons dans ce second cadre. Nous proposons une stratégie d'apprentissage multi-vues à deux niveaux, basée sur une analyse PAC-Bayésienne qui permet de dériver des bornes en généralisation pour des modèles exprimés comme une combinaison pondérée sur un ensemble de classifieurs ou de vues dans notre cas. Dans ce cadre, étant donné un ensemble de votants spécifiques à chaque vue, nous proposons de définir une hiérarchie de distributions *a posteriori* et *a priori* sur les vues, de sorte que *(i)* pour chaque vue v, nous considérons une distribution *prior* P_v et apprenons une distribution *posterior* Q_v sur l'ensemble des votants spécifiques à cette vue, et *(ii)* nous considérons une distribution *prior* π et apprenons une distribution *posterior* ρ sur l'ensemble des vues (voir la Figure 3.1), respectivement appelées *hyper-prior* et *hyper-posterior*. En suivant cette hiérarchie, nous définissons un vote de majorité multi-vues où les classifieurs spécifiques à chaque vue sont pondérés selon les distributions *posterior* et *hyper-posterior*. Ainsi, nous étendons la théorie PAC-Bayésienne classique à l'apprentissage multi-vues avec plus de 2 vues et dérivons une borne de généralisation PAC-Bayésienne pour

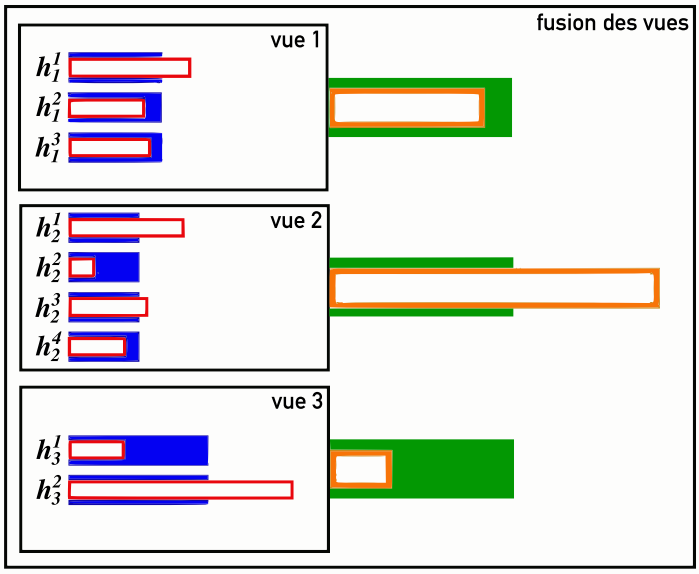


Figure 3.1. *Exemple de distributions hiérarchiques dans le cadre multi-vues avec 3 vues. Pour toutes les vues $v \in \{1,2,3\}$, on a un ensemble de n_v votants $\mathbb{H}_v = \{h_v^1, \ldots, h_v^{n_v}\}$ et une distribution prior P_v sur $\mathbb{H}_v$ (en bleu). De plus, on a une distribution hyper-prior π (en vert) sur l'ensemble des 3 vues. L'objectif est d'apprendre un ensemble de distributions posterior $\{Q_v\}_{v=1}^3$ (en rouge) et une distribution hyper-posterior ρ (en orange). Dans cet exemple, la longueur d'un rectangle représente le poids (ou la probabilité) assigné à un votant ou à une vue.*

notre vote de majorité multi-vues. Notre approche englobe celle de Amini et al. (2009) qui considère une distribution uniforme pour combiner les prédictions des classifieurs spécifiques à chaque vue. En outre, comparé au travail PAC-Bayésien de Sun et al. (2017), nous nous focalisons sur le cas plus général d'apprentissage multi-vues avec plus de 2 vues.

Sur le plan pratique, nous proposons un algorithme, appelé PB-MVBoost, inspiré de l'idée du *boosting* (Freund, 1995 ; Freund et Schapire, 1997). PB-MVBoost est une méthode ensembliste qui apprend un vote de majorité multi-vues en combinant des votants spécifiques à chaque vue. Il est connu que contrôler la diversité entre les classifieurs spécifiques à une vue ou entre les vues est un élément clé de l'apprentissage multi-vues (Kuncheva, 2014 ; Morvant et al., 2014) Ainsi, pour apprendre les poids associés aux vues en prenant en compte un compromis entre la diversité et la précision, nous utilisons à la C-borne multi-vues (Germain et al., 2015 ; Roy et al., 2016) qui, comme en mono-vue, est une relaxation du risque du vote de majorité. Concrètement, à chaque itération de notre algorithme, nous apprenons : *(i)* les poids des votants spécifiques à une vue en fonction de leur capacité à traiter les exemples de la vue correspondante (capturant des informations propres à chaque vue), *(ii)* les poids des vues en minimisant la C-borne multi-vues.

3.2 Notations et contexte

Nous considérons des tâches de classification binaire où une donnée $\mathbf{x} = (x^1, \ldots, x^V)$ est une observation décrite selon $V \geq 2$ vues (ou modalités), *i.e.*, l'espace d'entrée est $\mathbb{X} = \mathbb{X}_1 \times \ldots \times \mathbb{X}_V$ (les vues ne sont pas nécessairement de la même dimension). L'ensemble des V vues est noté $\mathbf{V}$. L'espace d'étiquetage binaire est $\mathbb{Y} = \{-1, +1\}$. Nous supposons que les exemples $(\mathbf{x}, y)$ sont tirés selon une distribution inconnue $\mathcal{D}$ sur $\mathbb{X} \times \mathbb{Y}$. Pour modéliser l'approche multi-vues à deux niveaux, nous adoptons le cadre suivant. Pour chaque vue $v \in \mathbf{V}$, nous considérons un ensemble de votants spécifique à la vue $\mathbb{H}_v$ de votants $h_v : \mathbb{X}_v \to \mathbb{Y}$,

ainsi qu'une distribution *prior* P_v sur $\mathbb{H}_v$. Étant donné une distribution *hyper-prior* π sur l'ensemble des vues $\mathbf{V}$ et un échantillon d'apprentissage multi-vues $\mathbb{S} = \{(\mathbf{x}_i, y_i)\}_{i=1}^m \sim \mathcal{D}^m$, l'objectif est double : *(i)* trouver une distribution *posterior* Q_v sur $\mathbb{H}_v$ pour chaque vue $v \in \mathbf{V}$, et *(ii)* trouver une distribution *hyper-posterior* ρ sur l'ensemble des vues $\mathbf{V}$. Nous cherchons donc une hiérarchie de distribution comme illustrée sur la Figure 3.1. Les distributions apprises définissent alors un vote de majorité multi-vues $\mathrm{MV}_\rho^{\mathbf{V}}$ défini par

$$\mathrm{MV}_\rho^{\mathbf{V}}(\mathbf{x}) = \operatorname{sign}\left[\mathop{\mathbb{E}}_{v\sim\rho}\mathop{\mathbb{E}}_{h_v\sim Q_v} h\left(x^v\right)\right].$$

Le but est alors de trouver les distributions *posterior* et *hyper-posterior* qui minimisent le risque réel $\mathrm{R}_{\mathcal{D}}(\mathrm{MV}_\rho^{\mathbf{V}})$ du vote défini par

$$\mathrm{R}_{\mathcal{D}}\left(\mathrm{MV}_\rho^{\mathbf{V}}\right) = \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} \mathrm{I}\left[\mathrm{MV}_\rho^{\mathbf{V}}(\mathbf{x}) \neq y\right].$$

Le risque de Gibbs associé est alors

$$\mathrm{R}_{\mathcal{D}}\left(G_\rho^{\mathbf{V}}\right) = \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}}\mathop{\mathbb{E}}_{v\sim\rho}\mathop{\mathbb{E}}_{h_v\sim Q_v} \mathrm{I}\left[h_v\left(x^v\right) \neq y\right],$$

où $G_\rho^{\mathbf{V}}$ est le classifieur stochastique de Gibbs. Le risque de Gibbs peut être réécrit en termes de désaccord multi-vues $\mathrm{d}_{\mathcal{D}}^{\mathbf{V}}(\rho)$ et d'erreur jointe multi-vues $\mathrm{e}_{\mathcal{D}}^{\mathbf{V}}(\rho)$ comme suit :

$$\begin{aligned}
\mathrm{R}_{\mathcal{D}}\left(G_\rho^{\mathbf{V}}\right) &= \frac{1}{2}\mathrm{d}_{\mathcal{D}}^{\mathbf{V}}(\rho) + \mathrm{e}_{\mathcal{D}}^{\mathbf{V}}(\rho),\\
\text{où}\quad \mathrm{d}_{\mathcal{D}}^{\mathbf{V}}(\rho) &= \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}_{\mathbb{X}}}\mathop{\mathbb{E}}_{v\sim\rho'}\mathop{\mathbb{E}}_{v'\sim\rho}\mathop{\mathbb{E}}_{h_v\sim Q_v}\mathop{\mathbb{E}}_{h'_v\sim Q_{v'}} \mathrm{I}\left[h_v\left(x^v\right) \neq h'_v\left(x^{v'}\right)\right],\\
\text{et}\quad \mathrm{e}_{\mathcal{D}}^{\mathbf{V}}(\rho) &= \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}}\mathop{\mathbb{E}}_{v\sim\rho}\mathop{\mathbb{E}}_{v'\sim\rho}\mathop{\mathbb{E}}_{h_v\sim Q_v}\mathop{\mathbb{E}}_{h_{v'}\sim Q_{v'}} \mathrm{I}\left[h_v\left(x^v\right) \neq y\right]\mathrm{I}\left[h'_v\left(x^{v'}\right) \neq y\right].
\end{aligned}$$

La contrepartie empirique du risque de Gibbs est

$$\begin{aligned}
\widehat{\mathrm{R}}_{\mathbb{S}}\left(G_\rho^{\mathbf{V}}\right) &= \frac{1}{m}\sum_{i=1}^m \mathop{\mathbb{E}}_{v\sim\rho}\mathop{\mathbb{E}}_{h_v\sim Q_v} \mathrm{I}\left[h_v\left(x_i^v\right) \neq y_i\right] &(3.1)\\
&= \frac{1}{2}\,\widehat{\mathrm{d}}_{\mathbb{S}}^{\mathbf{V}}(\rho) + \widehat{\mathrm{e}}_{\mathbb{S}}^{\mathbf{V}}(\rho), &(3.2)
\end{aligned}$$

où $\widehat{\mathrm{d}}_{\mathbb{S}}^{\mathbf{V}}(\rho)$ et $\widehat{\mathrm{e}}_{\mathbb{S}}^{\mathbf{V}}(\rho)$ sont les estimations empiriques de $\mathrm{d}_{\mathcal{D}}^{\mathbf{V}}(\rho)$ et de $\mathrm{e}_{\mathcal{D}}^{\mathbf{V}}(\rho)$ sur $\mathbb{S}$. Comme dans le cadre PAC-Bayes classique (avec une seule vue), le risque de Gibbs est une relaxation du risque du vote de majorité. On a $\mathrm{R}_{\mathcal{D}}(\mathrm{MV}_\rho^{\mathbf{V}}) \leq 2\mathrm{R}_{\mathcal{D}}(G_\rho^{\mathbf{V}})$. En outre, la C-borne du Théorème 2.4.3 peut être étendue au cadre multi-vues.

Théorème 3.2.1 (La C-borne multi-vues)**.** Soit $V \geq 2$ le nombre de vues. Pour toute distribution $\mathcal{D}$ on $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble de distributions $\{Q_v\}_{v=1}^V$ sur $\{\mathbb{H}_v\}_{v=1}^V$ et pour toute distribution ρ sur les vues $\mathbf{V}$, si $\mathrm{R}_{\mathcal{D}}(G_\rho^{\mathbf{V}}) < \frac{1}{2}$, alors on a

$$\begin{aligned}
\mathrm{R}_{\mathcal{D}}\left(\mathrm{MV}_\rho^{\mathbf{V}}\right) &\leq 1 - \frac{\left(1 - 2\mathrm{R}_{\mathcal{D}}\left(G_\rho^{\mathbf{V}}\right)\right)^2}{1 - 2\mathrm{d}_{\mathcal{D}}^{\mathbf{V}}(\rho)} &(3.3)\\
&\leq 1 - \frac{\left(1 - 2\mathop{\mathbb{E}}_{v\sim\rho}\mathrm{R}_{\mathcal{D}}\left(G_{Q_v}\right)\right)^2}{1 - 2\mathop{\mathbb{E}}_{v\sim\rho}\mathrm{d}_{\mathcal{D}}\left(Q_v\right)}, &(3.4)
\end{aligned}$$

où $\mathrm{R}_{\mathcal{D}}(G_{Q_v})$ et $\mathrm{d}_{\mathcal{D}}(Q_v)$ sont *resp.* le risque de Gibbs et le désaccord pour une seule vue v.

Comme dans la situation classique avec une seule vue, les Équations (3.3) et (3.4) suggèrent que pour apprendre un vote de majorité performant, il faut trouver bon compromis entre le risque de Gibbs $\mathrm{R}_{\mathcal{D}}(G_\rho^{\mathbf{V}})$ et le désaccord $\mathrm{d}_{\mathcal{D}}^{\mathbf{V}}(\rho)$.

3.3 Bornes PAC-Bayésiennes pour le multi-vues

3.3.1 Théorème PAC-Bayésien général

Nous présentons, dans le Théorème 3.3.1, notre théorème général PAC-Bayésien pour l'apprentissage multi-vues. Comme souligné dans la Section 2.4.5, une étape clé dans les preuves PAC-Bayésiennes est l'utilisation d'une inégalité de changement de mesure basée sur l'inégalité de Donsker-Varadan (Proposition 2.4.1). Le Lemme 3.3.1 étend cet outil à notre cadre multi-vues.

Lemme 3.3.1 (Inégalité de Donsker-Varadhan pour le multi-vues)**.** Pour tout ensemble de distributions *prior* $\{P_v\}_{v=1}^V \in \{\mathbb{M}^*(\mathbb{H}_v)\}_{v=1}^V$, pour toute distribution *hyper-prior* $\pi \in \mathbb{M}^*(\mathbf{V})$, pour toute fonction mesurable $\varphi : \mathbb{H}_v \times (\mathbb{X} \times \mathbb{Y})^m \to \mathbb{R}$ telle que $\mathbb{E}_{h'_v \sim P_v} e^{\varphi(h'_v, \mathbb{S})} < +\infty$, on a

$$\forall \mathbb{S} \in (\mathbb{X} \times \mathbb{Y})^m, \quad \forall \{Q_v\}_{v=1}^V \in \{\mathbb{M}(\mathbb{H}_v)\}_{v=1}^V, \quad \forall \rho \in \mathbb{M}(\mathbf{V}),$$

$$\mathop{\mathbb{E}}_{v\sim\rho} \mathop{\mathbb{E}}_{h_v\sim Q_v} \varphi(h_v, \mathbb{S}) \leq \mathop{\mathbb{E}}_{v\sim\rho} \mathrm{KL}(Q_v \| P_v) + \mathrm{KL}(\rho\|\pi) + \ln\left(\mathop{\mathbb{E}}_{v\sim\pi} \mathop{\mathbb{E}}_{h_v\sim P_v} e^{\varphi(h_v,\mathbb{S})}\right).$$

En se basant sur ce lemme, le théorème suivant est une généralisation du Théorème 2.4.5 au cadre multi-vues.

Théorème 3.3.1 (Borne PAC-Bayésienne générale multi-vues)**.** Soit $V \geq 2$ vues. Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble de distributions *prior* $\{P_v\}_{v=1}^V \in \{\mathbb{M}^*(\mathbb{H}_v)\}_{v=1}^V$, pour toute distribution *hyper-prior* $\pi \in \mathbb{M}^*(\mathbf{V})$, pour toute fonction mesurable $\varphi : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y})^m \to \mathbb{R}$, pour tout $\delta \in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\begin{array}{ll} \forall \{Q_v\}_{v=1}^V \in \{\mathbb{M}(\mathbb{H}_v)\}_{v=1}^V, & \forall \rho \in \mathbb{M}(\mathbf{V}), \\ \mathop{\mathbb{E}}_{v\sim\rho} \mathop{\mathbb{E}}_{h_v\sim Q_v} \varphi(h_v,\mathbb{S}) \leq & \mathop{\mathbb{E}}_{v\sim\rho} \mathrm{KL}(Q_v\|P_v) + \mathrm{KL}(\rho\|\pi) \\ & + \ln\left[\frac{1}{\delta} \mathop{\mathbb{E}}_{\mathbb{S}'\sim\mathcal{D}^m} \mathop{\mathbb{E}}_{v\sim\pi} \mathop{\mathbb{E}}_{h'\sim P_v} e^{\varphi(h',\mathbb{S}')}\right] \end{array}\right] \geq 1-\delta.$$

Ce résultat ressemble au Théorème 2.4.5 du cadre classique, la principale différence provient de l'introduction des distributions *prior* et *posterior* spécifiques aux vues. En effet, ces distributions induisent le terme additionnel $\mathbb{E}_{v\sim\rho}\mathrm{KL}(Q_v\|P_v)$ qui correspond à l'espérance sur les vues de la KL-divergence spécifique à une vue tirée selon l'*hyper-posterior* ρ. Cette KL-divergence capture donc la différence "en moyenne" entre les distributions *posterior* et *prior* de chaque vue.

3.3.2 Spécialisations aux approches classiques

En suivant le principe de la Section 2.4.4 (pour les bornes classiques), nous spécialisons le Théorème 3.3.1 en instanciant la fonction $\varphi()$. Dans les bornes obtenues, nous décomposons le risque empirique de Gibbs avec l'Équation (3.2) pour mettre explicitement en évidence la diversité et la complémentarité entre les vues via le désaccord et l'erreur jointe.

Corollaire 3.3.1 (Borne à la McAllester)**.** Soit $V \geq 2$ vues. Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble de distributions *prior* $\{P_v\}_{v=1}^{V} \in \{\mathbb{M}^*(\mathbb{H}_v)\}_{v=1}^{V}$, pour toute distribution *hyper-prior* $\pi \in \mathbb{M}^*(\mathbf{V})$, pour tout $\delta \in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\begin{array}{l}\forall\,\{Q_v\}_{v=1}^{V} \in \{\mathbb{M}(\mathbb{H}_v)\}_{v=1}^{V},\ \forall \rho \in \mathbb{M}(\mathbf{V}),\\ \mathsf{R}_{\mathcal{D}}\left(G_\rho^{\mathbf{V}}\right) \leq \frac{1}{2}\widehat{\mathrm{d}}_{\mathbb{S}}^{\mathbf{V}}(\rho) + \widehat{\mathrm{e}}_{\mathbb{S}}^{\mathbf{V}}(\rho) + \sqrt{\frac{1}{2m}\left(\mathop{\mathbb{E}}_{v\sim\rho}\mathrm{KL}\left(Q_v\|P_v\right) + \mathrm{KL}\left(\rho\|\pi\right) + \ln\frac{2\sqrt{m}}{\delta}\right)}\end{array}\right] \geq 1-\delta.$$

Corollaire 3.3.2 (Borne à la Catoni)**.** Soit $V \geq 2$ vues. Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble de distributions *prior* $\{P_v\}_{v=1}^{V} \in \{\mathbb{M}^*(\mathbb{H}_v)\}_{v=1}^{V}$, pour toute distribution *hyper-prior* $\pi \in \mathbb{M}^*(\mathbf{V})$, pour tout $C > 0$, pour tout $\delta \in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\begin{array}{l}\forall\,\{Q_v\}_{v=1}^{V} \in \{\mathbb{M}(\mathbb{H}_v)\}_{v=1}^{V},\ \forall \rho \in \mathbb{M}(\mathbf{V}),\quad \mathsf{R}_{\mathcal{D}}\left(G_\rho^{\mathbf{V}}\right) \leq \\ \frac{1}{1-e^{-C}}\left[1-\exp\left(-C\left(\frac{1}{2}\widehat{\mathrm{d}}_{\mathbb{S}}^{\mathbf{V}}(\rho)+\widehat{\mathrm{e}}_{\mathbb{S}}^{\mathbf{V}}(\rho)\right)-\frac{1}{m}\left[\mathop{\mathbb{E}}_{v\sim\rho}\mathrm{KL}(Q_v\|P_v)+\mathrm{KL}(\rho\|\pi)+\ln\frac{1}{\delta}\right]\right)\right]\end{array}\right] \geq 1-\delta.$$

Corollaire 3.3.3 (Borne à la Seeger)**.** Soit $V \geq 2$ vues. Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble de distributions *prior* $\{P_v\}_{v=1}^{V} \in \{\mathbb{M}^*(\mathbb{H}_v)\}_{v=1}^{V}$, pour toute distribution *hyper-prior* $\pi \in \mathbb{M}^*(\mathbf{V})$, pour tout $\delta \in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\begin{array}{l}\forall\,\{Q_v\}_{v=1}^{V} \in \{\mathbb{M}(\mathbb{H}_v)\}_{v=1}^{V},\ \forall \rho \in \mathbb{M}(\mathbf{V}),\\ \mathrm{kl}\left[\frac{1}{2}\widehat{\mathrm{d}}_{\mathbb{S}}^{\mathbf{V}}(\rho)+\widehat{\mathrm{e}}_{\mathbb{S}}^{\mathbf{V}}(\rho), \mathsf{R}_{\mathcal{D}}\left(G_\rho^{\mathbf{V}}\right)\right] \leq \frac{1}{m}\left(\mathop{\mathbb{E}}_{v\sim\rho}\mathrm{KL}(Q_v\|P_v)+\mathrm{KL}(\rho\|\pi)+\ln\frac{2\sqrt{m}}{\delta}\right)\end{array}\right] \geq 1-\delta.$$

Les bornes des trois corollaires précédents majorent le risque de Gibbs. Comme dans le cadre classique avec une seule vue, un facteur 2 doit s'appliquer pour obtenir des bornes sur le risque du vote de majorité.

3.3.3 C-Borne multi-vues PAC-Bayésienne

Pour obtenir une borne plus précise sur le risque du vote de majorité multi-vues, nous énonçons dans le théorème suivant une borne en généralisation PAC-Bayésienne pour la C-borne multi-vues du Théorème 3.2.1.

Théorème 3.3.2 (C-borne multi-vues PAC-Bayésienne)**.** Soit $V \geq 2$ vues. Pour toute distribution $\mathcal{D}$ sur $\mathbb{X} \times \mathbb{Y}$, pour tout ensemble de distributions *prior* $\{P_v\}_{v=1}^{V} \in \{\mathbb{M}^*(\mathbb{H}_v)\}_{v=1}^{V}$, pour toute distribution *hyper-prior* $\pi \in \mathbb{M}^*(\mathbf{V})$, pour tout $\delta \in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\mathsf{R}_{\mathcal{D}}\left(\mathsf{MV}_{\rho}^{\mathbf{V}}\right) \leq 1 - \frac{\left(1 - 2\mathop{\mathbb{E}}_{v\sim\rho}\sup\left(\mathbf{r}_{Q_v,\mathbb{S}}^{\delta/2}\right)\right)^2}{1 - 2\mathop{\mathbb{E}}_{v\sim\rho}\inf\mathbf{d}_{Q_v,\mathbb{S}}^{\delta/2}}\right] \geq 1-\delta,$$

où $\quad \mathbf{r}_{Q_v,\mathbb{S}}^{\delta/2} = \left\{r : \mathrm{kl}\left(\widehat{\mathsf{R}}_{\mathbb{S}}\left(G_{Q_v}\right)\|r\right) \leq \frac{1}{m}\left[\mathrm{KL}\left(Q_v\|P_v\right) + \ln\frac{4\sqrt{m}}{\delta}\right] \text{ et } r \leq \frac{1}{2}\right\},$

et $\quad \mathbf{d}_{Q_v,\mathbb{S}}^{\delta/2} = \left\{d : \mathrm{kl}\left(\widehat{\mathrm{d}}_{\mathbb{S}}(Q_v)\|d\right) \leq \frac{1}{m}\left[2\,\mathrm{KL}\left(Q_v\|P_v\right) + \ln\frac{4\sqrt{m}}{\delta}\right]\right\}.$

Nous utilisons ce théorème comme justification théorique de la minimisation de la C-borne multi-vues dans l'algorithme décrit dans la section suivante.

3.4 Algorithme multi-vues basé sur la C-borne

Il est établi que la diversité est un élément clé du succès de combinaisons de classifieur, et plus généralement des méthodes ensemblistes (*e.g.*, Kuncheva, 2014). Dans le cadre de l'apprentissage multi-vues, fusionner des votants suffisamment diverses est donc essentiel pour obtenir de bonnes performances. Bien qu'il n'y ait pas de consensus sur la définition de la diversité, il existe des métriques populaires de diversité basées sur la différence entre chaque paire de votants individuels, telles que les Q-statistiques, le coefficient de corrélation, etc. Comme nous l'avons détaillé dans Morvant et al. (2014), le désaccord entre paire de votants, intervenant tout particulièrement dans la C-borne, correspond à une de ces métriques (pondérée par le *posterior*). Ainsi les algorithmes qui découlent de la minimisation de la C-borne favorise les combinaisons de votants diverses tout en gardant de bonnes performances individuelles et apparaît ainsi comme une bonne solution pour combiner les prédictions de classifieurs appris séparément à partir de différentes vues.

Cette section présente un algorithme, appelé PB-MVBoost et basé sur le principe du *boosting*, dans le cadre de l'apprentissage multi-vues à deux niveaux. PB-MVBoost, résumé dans l'Algorithme 3.1, est une méthode ensembliste qui renvoie un vote de majorité multi-vues exprimé comme une combinaison de votants spécifiques à une vue. Pour ce faire, nous proposons un algorithme itératif qui vise à minimiser la C-borne multi-vues du Théorème 3.2.1 en contrôlant le compromis entre diversité et performance. Plus précisément, à chaque itération, nous apprenons *(i)* les poids sur les votants spécifiques aux vues, basés sur leur capacité à traiter les exemples de la vue considérée (capturant ainsi des informations spécifiques à chaque vue), et *(ii)* les poids sur les vues en minimisant la C-borne multi-vues.

Étant donné V vues et un ensemble d'apprentissage $\mathbb{S} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{m} \in (\mathbb{X}\times\{-1,+1\})^m$ de taille m, l'Algorithme 3.1 maintient une distribution sur les exemples (initialisée à la distribution uniforme, Lignes 1-2). À chaque itération t, (Ligne 4) nous apprenons V classifieurs faibles spécifiques à une vue selon la distribution courante $\mathcal{D}_t$; (Ligne 5) les erreurs associées sont estimées par le terme ϵ_v^t. Comme dans l'algorithme Adaboost (Freund et Schapire,

Algorithme 3.1 PB-MVBoost *(PAC-Bayesian Multiview Boosting)*

Entrées : $\mathbb{S} = \{(\mathbf{x}_i, y_i), \ldots, (\mathbf{x}_m, y_m)\}$, avec $\mathbf{x}_i = (x_i^1, x_i^2, \ldots, x_i^V)$ et $y_i \in \{-1, 1\}$, pour chaque vue $v \in \mathbf{V}$, un ensemble d'hypothèses $\mathbb{H}_v$, nombre d'itérations T

1: $\forall \mathbf{x}_i \in \mathbb{S},\ \mathcal{D}_1(\mathbf{x}_i) \leftarrow \frac{1}{m}$

2: $\forall v \in \mathbf{V},\ \rho_v^1 \leftarrow \frac{1}{V}$ et $H_v \leftarrow \phi$

3: **pour** $t = 1, \ldots, T$ **faire**

4: $\quad \forall v \in \mathbf{V}$, apprendre un classifieur faible spécifique à la vue $h_v^{(t)}$ avec $\mathcal{D}_{(t)}$

5: $\quad \forall v \in \mathbf{V}$, calcul de l'erreur de h_v^t : $\epsilon_v^{(t)} \leftarrow \mathbb{E}_{(\mathbf{x}_i, y_i) \sim \mathcal{D}_{(t)}} \mathrm{I}\left[h_v^t(x_i^v) \neq y_i\right]$

6: $\quad \forall v \in \mathbf{V}$, calcul du poids de h_v^t : $Q_v^{(t)} \leftarrow \frac{1}{2}\left[\ln\left(\frac{1-\epsilon_v^{(t)}}{\epsilon_v^{(t)}}\right)\right]$

7: $\quad \forall v \in \mathbf{V}, \mathbb{H}_v \leftarrow \mathbb{H}_v \cup \{h_v^{(t)}\}$

8: $\quad$ **Optimisation** de la C-Borne multi-vues pour apprendre poids sur les vues

$$\max_{\rho} \frac{\left[1 - 2\sum_{v=1}^{V} \rho_v^{(t)} r_v^{(t)}\right]^2}{1 - 2\sum_{v=1}^{V} \rho_v^{(t)} d_v^{(t)}} \quad \text{telle que } \forall v \in \mathbf{V},\ \sum_{v=1}^{V} \rho_v^{(t)} = 1, \quad \rho_v^{(t)} \geq 0$$

$$\text{où } \forall v \in \mathbf{V},\ r_v^{(t)} \leftarrow \mathop{\mathbb{E}}_{(\mathbf{x}_i, y_i) \sim \mathcal{D}_{(t)}} \mathop{\mathbb{E}}_{h_v \sim \mathbb{H}_v} \mathrm{I}\left[h_v(x_i^v) \neq y_i\right]$$

$$\forall v \in \mathbf{V},\ d_v^{(t)} \leftarrow \mathop{\mathbb{E}}_{(\mathbf{x}_i, y_i) \sim \mathcal{D}_{(t)}} \mathop{\mathbb{E}}_{h_v, h_v' \sim \mathbb{H}_v} \mathrm{I}\left[h_v(x_i^v) \neq h_v'(x_i^v)\right]$$

9: $\quad$ **pour** tout $\mathbf{x}_i \in S$ **faire**

10: $\quad\quad \mathcal{D}_{(t+1)}(\mathbf{x}_i) \leftarrow \dfrac{\mathcal{D}_{(t)}(\mathbf{x}_i) \exp\left(-y_i \sum_{v=1}^{V} \rho_v^{(t)} \left(Q_v^{(t)} h_v^{(t)}(x_i^v)\right)\right)}{\sum_{j=1}^{m} \mathcal{D}_{(t)}(\mathbf{x}_j) \exp\left(-y_j \sum_{v=1}^{V} \rho_v^{(t)} \left(Q_v^{(t)} h_v^{(t)}\left(x_j^v\right)\right)\right)}$

11: **Renvoyer** le vote de majorité multi-vues tel que pour tout exemple $\mathbf{x}$, on a

$$\mathrm{MV}_{\rho}^{\mathbf{V}}(\mathbf{x}) = \operatorname{sign}\left(\sum_{v=1}^{V} \rho_v^T \sum_{t=1}^{T} Q_v^{(t)} h_v^{(t)}(x^v)\right)$$

1997), (Ligne 6) les poids des classifieurs $\left\{Q_v^{(t)}\right\}_{v=1}^{V}$ sont définis en fonction de ces erreurs par

$$\forall v \in \mathbf{V}, Q_v^{(t)} \leftarrow \frac{1}{2}\left[\ln\left(\frac{1-\epsilon_v^{(t)}}{\epsilon_v^{(t)}}\right)\right].$$

Pour apprendre les poids $(\rho_v)_{1 \leq v \leq V}$ sur les vues (Ligne 8), nous optimisons la C-borne empirique. Notons que dans l'article original (Goyal et al., 2019), nous avons montré empiriquement que l'algorithme minimise la C-borne multi-vues tout au long des itérations. Enfin (Lignes 9-10), toujours en suivant le principe de l'algorithme Adaboost, nous mettons à jour la distribution sur les exemples d'apprentissage $\mathbf{x}_i$ en s'assurant que les poids des exemples mal classés (*resp.* bien classés) par le vote de majorité final augmentent (*resp.* diminuent) :

$$\mathcal{D}_{(t+1)}(\mathbf{x}_i) \leftarrow \frac{\mathcal{D}_{(t)}(\mathbf{x}_i) \exp\left(-y_i \sum_{v=1}^{V} \rho_v^{(t)} \left(Q_v^{(t)} h_v^{(t)}(x_i^v)\right)\right)}{\sum_{j=1}^{m} \mathcal{D}_{(t)}(\mathbf{x}_j) \exp\left(-y_j \sum_{v=1}^{V} \rho_v^{(t)} \left(Q_v^{(t)} h_v^{(t)}\left(x_j^v\right)\right)\right)}$$

Intuitivement, cela contraint les classifieurs spécifiques à chaque vue à être cohérents les uns avec les autres, ce qui est essentiel pour l'apprentissage multi-vues (Sebban, Suchier

et al., 2009 ; Koço et Capponi, 2011 ; Morvant et al., 2014). Enfin, après T itérations de l'algorithme, nous obtenons le vote de majorité multi-vues :

$$\mathrm{MV}_{\rho}^{\mathbf{V}}(\mathbf{x}) = \mathrm{sign}\left(\sum_{v=1}^{V} \rho_v^T \sum_{t=1}^{T} Q_v^{(t)} h_v^{(t)}\left(x^v\right)\right).$$

3.5 Résumé des expériences

Nous avons comparé notre algorithme PB-MVBoost avec des arbres de décisions comme votants sur MNIST et Reuters (en classification binaire). Les algorithmes auxquels nous nous sommes comparés sont les suivants. En autre, nous avons considéré des algorithmes d'apprentissage d'un vote de majorité sur uniquement la sortie des classifieurs spécifiques aux vues : avec des poids uniforme comme Amini et al. (2009), ou avec des poids appris en utilisant Adaboost. Nous nous sommes également comparés à rBoost.SH (Peng et al., 2017), une méthode de *boosting* pour l'apprentissage multi-vues durant laquelle une distribution globale est maintenue sur l'ensemble des exemples et des vues, et où, à chaque itération, une vue est sélectionnée et un classifieur spécifique à cette vue est appris. Afin de mettre en évidence l'utilité de la C-borne, nous avons également considéré une version de PB-MVBoost sans l'optimisation de la C-borne.

Les expériences ont permis de montrer que PB-MVBoost était en moyenne meilleur en termes de performance et de F1-mesure sur les jeux de données considérés. En faisant varier artificiellement la quantité d'exemples par classe, nous avons pu également observer que PB-MVBoost était capable de gérer des données déséquilibrées. La prise en considération d'une stratégie avec une hiérarchie de distributions à deux niveaux a donc permis une meilleure prise en considération de la diversité/complémentarité à la fois des vues, mais également des votants de base permettant même de contrer d'éventuels problèmes de déséquilibre dans l'ensemble d'apprentissage.

3.6 Conclusion

Dans ce chapitre, nous avons introduit la première analyse PAC-Bayésienne de l'apprentissage multi-vues avec plus de 2 vues. La principale nouveauté par rapport au cadre mono-vue réside dans l'introduction d'une hiérarchie à deux niveaux de distributions : un niveau intra-vue (où l'on apprend une distribution sur un ensemble de votants spécifiques à une vue donnée) et un niveau inter-vues (où l'on apprend une distribution sur les différentes vues). Les résultats théoriques obtenus prennent la forme de bornes PAC-Bayésiennes relativement classiques, adaptées à ce contexte hiérarchique. Nous avons ensuite utilisé ces résultats comme source d'inspiration pour concevoir un algorithme inspiré du *boosting*. Cet algorithme apprend de manière itérative les deux niveaux de distributions conjointement, en s'appuyant sur la C-borne multi-vues. Cette approche permet de prendre en compte non seulement la diversité entre les vues, mais aussi la diversité au sein des votants d'une même vue, améliorant l'apprentissage du vote de majorité final.

Adaptation de domaine PAC-Bayésienne 4

Contexte

Les travaux présentés dans ce chapitre sont en continuité directe avec les travaux effectués durant ma thèse suite à ma collaboration avec François Laviolette et Pascal Germain. Ces travaux ont donné lieu non seulement à une collaboration de long terme avec Pascal Germain, mais également à la co-rédaction d'un livre sur la théorie de l'adaptation de domaine (Redko et al., 2019, 2020). D'un point de vue scientifique, les résultats obtenus sont les premières bornes PAC-Bayésiennes pour l'adaptation de domaine. Le contenu de ce chapitre est proche de l'article publié dans le journal Neurocomputing (Germain et al., 2020) qui résume l'ensemble de nos contributions dans ce contexte (Germain et al., 2013, 2016a).

Je tiens à mentionner que François Laviolette, décédé en 2021, a joué un rôle important dans l'orientation de ma carrière, puisque c'est avec lui que je me suis familiarisée avec la théorie PAC-Bayésienne, fil rouge de ce manuscrit. Je tiens donc à lui rendre hommage au travers de ce chapitre.

4.1 Introduction

En tant qu'être humain, nous apprenons de ce que nous avons vu auparavant. Prenons l'exemple de notre processus éducatif : lorsqu'un étudiant assiste à un nouveau cours, les connaissances qu'il a acquises des cours précédents l'aident à comprendre ce nouveau cours. Cependant, les approches traditionnelles en apprentissage automatique supposent que les données d'apprentissage et les données sur lesquelles nous voulons appliquer notre modèle sont tirées selon la même distribution de probabilité. Cette hypothèse est difficile à vérifier pour de nombreuses applications réelles dès que l'on souhaite réutiliser un modèle pour une autre tâche. Par exemple, un système de filtrage de spams performant pour un utilisateur donné ne le sera pas nécessairement pour un utilisateur recevant des e-mails de natures

différentes. En d'autres termes, les données d'apprentissage associées à un ou plusieurs utilisateurs peuvent ne pas être représentatives des données d'un autre utilisateur. Cela renforce le besoin de concevoir des méthodes pour adapter un classifieur appris à partir des données d'apprentissage (la source) à des données différentes (la cible). Une solution pour s'attaquer à ce problème est de considérer le cadre de l'*adaptation de domaine*[1], qui correspond à la situation pour laquelle la distribution générant les données cibles (le *domaine cible*) diffère de celle générant les données sources (le *domaine source*). Notons que l'adaptation de domaine est une tâche connue pour être difficile même sous de fortes hypothèses (Ben-David et al., 2010b ; Ben-David et Urner, 2012, 2014).

De nombreuses approches existent pour s'attaquer à l'adaptation de domaine, souvent avec la même idée sous-jacente : si nous sommes en mesure de trouver une transformation pour "rapprocher" les distributions, alors nous pouvons apprendre un modèle avec les étiquettes observées. Cette technique peut être réalisée par une repondération de l'importance des données étiquetées (*importance reweighting,* Huang et al., 2006 ; Sugiyama et al., 2007 ; Cortes et al., 2010, 2015), qui une méthode populaire en cas de *covariate-shift* lorsque les domaines partagent la même fonction d'étiquetage (*e.g.,* Huang et al., 2006 ; Sugiyama et al., 2008). Une autre approche se base sur des procédures d'auto-étiquetage, où le but est de transférer les étiquettes sources sur les données cibles none-étiquetées (*e.g.*, Bruzzone et Marconcini, 2010 ; Habrard et al., 2013 ; Morvant, 2015). Une troisième solution consiste à construire un nouvel espace de représentation dans lequel les données sources et cibles seront les plus indiscernables possible. Puis, un algorithme standard d'apprentissage supervisé peut être utilisé pour apprendre un modèle sur les données sources étiquetées (*e.g.*, Glorot et al., 2011 ; Chen et al., 2012 ; Courty et al., 2016 ; Courty et al., 2017 ; Li et al., 2019b). La représentation et le modèle peuvent également être appris simultanément ; dû à l'essor des méthodes d'apprentissage profond, cette stratégie est devenue très populaire (*e.g.*, Ganin et al., 2016 ; Ding et Fu, 2018 ; Shu et al., 2018 ; Li et al., 2019a ; Sebag et al., 2019).

Le travail présenté dans ce chapitre appartient à une autre approche principalement explorée pour dériver des bornes en généralisation pour l'adaptation de domaine. Cette approche implique généralement une mesure de divergence entre les distributions source et cible (*e.g.* Ben-David et al., 2006 ; Li et Bilmes, 2007 ; Ben-David et al., 2010a ; Morvant et al., 2012a ; Zhang et al., 2012 ; Cortes et Mohri, 2014 ; Redko et al., 2017). Une telle mesure dépend de l'ensemble d'hypothèses $\mathbb{H}$ considéré par l'algorithme d'apprentissage. L'idée est de chercher un ensemble $\mathbb{H}$ qui permet de minimiser la divergence entre les distributions tout en conservant de bonnes performances sur les données étiquetées sources : si les distributions sont proches selon cette mesure, alors la capacité en généralisation sera "plus facile" à estimer. En fait, la définition d'une telle mesure pour quantifier à quel point les domaines sont reliés est une question majeure en adaptation de domaine. Par exemple, en classification binaire avec la fonction perte 0-1, Ben-David et al. (2006, 2010a) ont considéré la *$\mathcal{H}\Delta\mathcal{H}$-divergence* entre les distributions marginales source et cible. Cette quantité dépend du désaccord maximal entre paires d'hypothèses et permet de déduire une borne en généralisation d'adaptation de domaine basée sur la dimension VC (Définition 2.3.2). La *discrepancy distance* (Mansour et al., 2009a) généralise la *$\mathcal{H}\Delta\mathcal{H}$-divergence* à des fonctions à valeurs réelles et à des fonctions de pertes plus générales. Cette mesure permet d'obtenir des bornes en généralisation d'adaptation de domaine basées sur la complexité de Rademacher (Définition 2.3.3). D'autres mesures ont été exploitées sous différentes hypothèses, comme la divergence de Rényi pour l'*importance reweighting* (Mansour et al., 2009b) ou la distance de Wasserstein qui permet

1. L'adaptation de domaine est souvent associée à l'apprentissage par transfert (Pan et Yang, 2010).

d'utiliser une stratégie de transport optimal en adaptation de domaine (Courty et al., 2016 ; Courty et al., 2017 ; Redko et al., 2017). Dans ces situations, l'adaptation de domaine peut être vue comme un compromis entre la complexité de la classe d'hypothèses $\mathbb{H}$, l'adaptabilité de $\mathbb{H}$ selon la divergence entre les domaines et le risque empirique source. Cependant, de nombreuses des méthodes se décomposent en deux étapes : *(i)* construire un espace de représentation en minimisant la divergence entre les domaines, puis *(ii)* apprendre un modèle sur le domaine source dans ce nouvel espace de représentation.

Le point de vue PAC-Bayésien. Une particularité de la théorie PAC-Bayésienne (Section 2.4) est qu'elle se concentre sur les algorithmes qui apprennent une distribution *posterior* ρ sur $\mathbb{H}$ (*i.e.*, un moyennage selon ρ) plutôt que sur un seul prédicteur $h \in \mathbb{H}$ (comme Ben-David et al. (2006) et d'autres travaux cités ci-dessus). Plus précisément, nous étudions le cadre de l'adaptation de domaine non-supervisée pour la classification binaire, où aucune étiquette cible n'est fournie à l'apprenant. Nous proposons deux analyses d'adaptation de domaine, introduites séparément dans Germain et al. (2013, 2016a).

Notre première approche suit la philosophie des travaux fondateurs de Ben-David et al. (2006) et de Mansour et al. (2009b) : le risque sur le domaine cible est majoré conjointement par le risque sur le domaine source, une divergence entre les distributions marginales et un terme non estimable [2] lié à la capacité d'adaptation dans l'espace de représentation. Pour obtenir un tel résultat, nous définissons une pseudo-métrique compatible avec le PAC-Bayes en définissant la divergence entre les domaines comme l'espérance selon ρ des désaccords entre paires d'hypothèses sur les deux domaines. Nous avons démontré que cette divergence est plus petite que la $\mathcal{H}\Delta\mathcal{H}$-divergence et est facilement estimable à partir de données.

Notre second résultat (Germain et al., 2016a) est une borne supérieure sur le risque sur le domaine cible qui apporte une vision originale de l'adaptation de domaine. Concrètement, le risque cible est toujours majoré par trois termes, mais ils diffèrent dans l'information qu'ils capturent. Le premier terme est estimable à partir de données non étiquetées et repose uniquement sur le désaccord sur le domaine cible. Le deuxième terme dépend de la performance sur le domaine source ; il est intéressant de noter que cette performance est pondérée par une mesure de divergence entre les domaines source et cible qui permet de contrôler la relation entre les domaines. Le troisième terme estime le "volume" du domaine cible trop éloigné du domaine source (qui doit être faible pour assurer l'adaptation).

À partir de ces résultats, nous dérivons des bornes en généralisation PAC-Bayésiennes pour nos deux bornes d'adaptation de domaine. Inspirés par ces bornes, nous proposons deux algorithmes adaptés aux classifieurs linéaires, appelés pbda et dalc. Contrairement à de nombreuses méthodes d'adaptation qui effectuent une procédure en deux étapes, pbda et dalc conjointement les compromis impliqués par les bornes.

4.2 Les travaux fondateurs

Avant de présenter nos résultats, nous rappelons les résultats fondateurs de la théorie de l'adaptation de domaine basée sur une mesure de divergence entre les domaines (Ben-David et al., 2006 ; Mansour et al., 2009a ; Ben-David et al., 2010a).

2. Le terme *non-estimable* des bornes d'adaptation peut être estimé avec des étiquettes des 2 domaines.

4.2.1 Notations et contexte

Nous étudions l'adaptation de domaine pour la classification binaire où $\mathbb{X} \subseteq \mathbb{R}^d$ est l'espace d'entrée de dimension d et $\mathbb{Y} = \{-1, +1\}$ est l'espace d'étiquetage. Le domaine source $\mathcal{S}$ et le domaine cible $\mathcal{T}$ sont des distributions (inconnues et fixées) sur $\mathbb{X} \times \mathbb{Y}$; les distributions marginales respectives sur $\mathbb{X}$ sont notées $\mathcal{S}_{\mathbb{X}}$ et $\mathcal{T}_{\mathbb{X}}$. Nous nous attaquons à la tâche difficile de l'adaptation de domaine non supervisée où aucune étiquette cible n'est disponible. Nous considérons donc un échantillon source étiqueté $\mathbb{S} = \{(\mathbf{x}_i^s, y_i)\}_{i=1}^{m_s}$ composé de m_s exemples tirés *i.i.d.* selon $\mathcal{S}$, et un échantillon cible non étiqueté $\mathbb{T} = \{\mathbf{x}_j^t\}_{j=1}^{m_t}$ composé de m_t exemples tirés *i.i.d.* selon $\mathcal{T}_{\mathbb{X}}$. Nous supposons que $\mathbb{H}$ est un ensemble d'hypothèses de $\mathbb{X}$ à $\mathbb{Y}$. Le risque source réel et le risque cible réel d'une hypothèse $h \in \mathbb{H}$ sur $\mathcal{S}$, *resp.* $\mathcal{T}$, sont la probabilité que h se trompe sur l'ensemble de la distribution $\mathcal{S}$, *resp.* $\mathcal{T}$,

$$\mathrm{R}_{\mathcal{S}}(h) = \mathop{\mathbb{E}}_{(\mathbf{x}^s, y) \sim \mathcal{S}} \ell_{01}\Big(h, (\mathbf{x}^s, y)\Big), \quad \text{et} \quad \mathrm{R}_{\mathcal{T}}(h) = \mathop{\mathbb{E}}_{(\mathbf{x}^t, y) \sim \mathcal{T}} \ell_{01}\Big(h, (\mathbf{x}^t, y)\Big),$$

où $\ell_{01}(h, (\mathbf{x}, y)) = \mathrm{I}[h(\mathbf{x}) \neq y]$ est la fonction perte 0-1. Nous notons $\widehat{\mathrm{R}}_{\mathbb{S}}(h)$ le risque empirique source associé estimé sur l'échantillon source $\mathbb{S}$. L'objectif principal en adaptation de domaine est alors d'apprendre — sans étiquette cible — un modèle $h \in \mathbb{H}$ amenant à la plus petite erreur réelle possible sur le domaine cible $\mathrm{R}_{\mathcal{T}}(h)$.

Dans la suite, nous avons besoin de la notion de désaccord entre deux hypothèses $(h, h') \in \mathbb{H}^2$. Nous notons $\mathrm{d}_{\mathcal{S}_{\mathbb{X}}}(h, h')$, respectivement $\mathrm{d}_{\mathcal{T}_{\mathbb{X}}}(h, h')$, le désaccord réel source, respectivement le désaccord réel cible. Ils sont définis par

$$\mathrm{d}_{\mathcal{S}_{\mathbb{X}}}(h, h') = \mathop{\mathbb{E}}_{\mathbf{x}^s \sim \mathcal{S}_{\mathbb{X}}} \mathrm{I}\Big[h(\mathbf{x}) \neq h'(\mathbf{x}^s)\Big], \quad \text{et} \quad \mathrm{d}_{\mathcal{T}_{\mathbb{X}}}(h, h') = \mathop{\mathbb{E}}_{\mathbf{x}^t \sim \mathcal{T}_{\mathbb{X}}} \mathrm{I}\Big[h(\mathbf{x}) \neq h'(\mathbf{x}^t)\Big].$$

Ces quantités mesurent la probabilité que h et h' renvoient une étiquette différente (*i.e.*, la probabilité qu'ils soient en désaccord).

4.2.2 La nécessité d'une divergence entre les domaines

L'objectif de l'adaptation de domaine est de trouver une hypothèse cible de risque faible, même si aucune étiquette cible n'est disponible. Cette tâche peut être impossible à résoudre même sous de fortes hypothèses (Ben-David et al., 2010b ; Ben-David et Urner, 2012, 2014). Cependant, pour étudier la capacité en généralisation dans une telle situation (via ce que l'on appelle une borne d'adaptation de domaine), il est crucial d'utiliser une mesure de "distance" entre le domaine source et le domaine cible : plus les domaines sont similaires, plus l'adaptation sera "facile". Concrètement, les domaines $\mathcal{S}$ et $\mathcal{T}$ diffèrent si leurs marginales $\mathcal{S}_{\mathbb{X}}$ et $\mathcal{T}_{\mathbb{X}}$ sont différentes et/ou si la fonction d'étiquetage source diffère de celle de la cible. Cela suggère de considérer deux mesures : une entre les marginales $\mathcal{S}_{\mathbb{X}}$ et $\mathcal{T}_{\mathbb{X}}$, et l'autre entre les étiquetages. Évidemment, si nous disposons d'étiquettes cibles, alors les deux mesures peuvent être combinées (*e.g.*, Zhang et al., 2012). Dans le cas contraire, les mesures sont distinctes puisque la meilleure hypothèse cible sera impossible à estimer. L'hypothèse généralement faites en adaptation de domaine est que la fonction d'étiquetage source est d'une certaine manière liée à celle du domaine cible. Sous cette hypothèse forte, l'objectif est alors de construire une représentation dans laquelle les marginales $\mathcal{S}_{\mathbb{X}}$ et $\mathcal{T}_{\mathbb{X}}$ sont proches tout en gardant une bonne performance sur le domaine source.

4.2.3 Bornes d'adaptation de domaine en classification binaire

Ben-David et al. (2010a) ont proposé la première borne d'adaptation de domaine, rappelée ci-dessous, sous l'hypothèse qu'il existe une hypothèse dans $\mathbb{H}$ de faible risque à la fois sur le domaine source et sur le domaine cible.

Théorème 4.2.1 (Borne d'adaptation de Ben-David et al. (2006, 2010a))**.** Soit $\mathbb{H}$ un ensemble symétrique[3] d'hypothèses. Pour tout domaine $\mathcal{S}$ et $\mathcal{T}$ sur $\mathbb{X}\times\mathbb{Y}$, on a

$$\forall h \in \mathbb{H}, \quad \mathrm{R}_{\mathcal{T}}(h) \leq \mathrm{R}_{\mathcal{S}}(h) + \tfrac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}}) + \mu_{h^*}\,, \tag{4.1}$$

$$\text{où} \quad \tfrac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}}) = \sup_{(h,h')\in\mathcal{H}^2} \left| \mathrm{d}_{\mathcal{T}_{\mathbb{X}}}(h,h') - \mathrm{d}_{\mathcal{S}_{\mathbb{X}}}(h,h') \right|, \tag{4.2}$$

est la $\mathcal{H}\Delta\mathcal{H}$-divergence entre les marginales $\mathcal{S}_{\mathbb{X}}$ et $\mathcal{T}_{\mathbb{X}}$, et $\mu_{h^*} = \mathrm{R}_{\mathcal{S}}(h^*) + \mathrm{R}_{\mathcal{T}}(h^*)$ est l'erreur de la meilleure hypothèse globale $h^* = \operatorname{argmin}_{h\in\mathbb{H}} (\mathrm{R}_{\mathcal{S}}(h) + \mathrm{R}_{\mathcal{T}}(h))$.

Cette borne repose sur 3 termes. Le premier $\mathrm{R}_{\mathcal{S}}(h)$ est l'erreur réelle classique sur domaine source. Le deuxième $\frac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}})$ dépend de $\mathbb{H}$ et correspond à l'écart maximum entre le désaccord source et cible entre paire d'hypothèses. En d'autres termes, $\frac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}})$ quantifie à quel point les hypothèses de $\mathbb{H}$ peuvent "capturer" les différences entre les marginales : plus cette mesure est faible pour un ensemble $\mathbb{H}$ donné, meilleures seront les garanties en généralisation. Le dernier terme $\mu_{h^*} = \mathrm{R}_{\mathcal{S}}(h^*) + \mathrm{R}_{\mathcal{T}}(h^*)$ est, quant à lui, lié à la meilleure hypothèse $h^* \in \mathbb{H}$ sur les domaines et agit comme une mesure de qualité de $\mathbb{H}$ en termes d'étiquetage. Si h^* n'est pas capable d'obtenir de bonnes performances sur les domaines source et cible, alors l'adaptation du domaine source vers le domaine cible ne sera pas possible. Comme mentionné par les auteurs, l'Équation (4.1) exprime un compromis entre la performance d'une hypothèse h, la complexité de $\mathbb{H}$ (quantifiée par Ben-David et al. avec une borne basée sur la dimension VC), et l'incapacité des hypothèses de $\mathbb{H}$ à détecter les différences entre les domaines source et cible.

Dans un second temps, Mansour et al. (2009a) ont proposé une extension de la $\mathcal{H}\Delta\mathcal{H}$-divergence de l'Équation (4.2), appelée la *discrepancy*, pour la régression et pour des fonctions pertes symétriques vérifiant l'inégalité triangulaire. Étant donné une telle fonction $\ell : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y}) \to \mathbb{R}^+$, la *discrepancy* $\mathrm{disc}_\ell(\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}})$ entre $\mathcal{S}_{\mathbb{X}}$ et $\mathcal{T}_{\mathbb{X}}$ est

$$\mathrm{disc}_\ell(\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}}) = \sup_{(h,h')\in\mathbb{H}^2} \left| \mathop{\mathbb{E}}_{\mathbf{x}^t\sim\mathcal{T}_{\mathbb{X}}} \ell(h, (\mathbf{x}^t, h'(\mathbf{x}^t)) - \mathop{\mathbb{E}}_{\mathbf{x}^s\sim\mathcal{S}_{\mathbb{X}}} \ell(h, (\mathbf{x}^s, h'(\mathbf{x}^s)) \right|$$

Pour la classification binaire, avec la perte 0-1, on a $\frac{1}{2}\mathrm{d}_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}}) = \mathrm{disc}_{\ell_{01}}(\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}})$. Bien que les deux mesures puissent coïncider, la borne d'adaptation de domaine de Mansour et al. (2009a), rappelée dans le théorème suivant, diffère du Théorème 4.2.1.

Théorème 4.2.2 (Borne d'adaptation de Mansour et al.)**.** Soit $\mathbb{H}$ un ensemble symétrique d'hypothèses. Pour tout domaine $\mathcal{S}$ et $\mathcal{T}$ sur $\mathbb{X}\times\mathbb{Y}$, on a

$$\forall h \in \mathbb{H}, \ \mathrm{R}_{\mathcal{T}}(h) - \mathrm{R}_{\mathcal{T}}(h^*_{\mathcal{T}}) \leq \mathrm{d}_{\mathcal{S}_{\mathbb{X}}}(h, h^*_{\mathcal{S}}) + \mathrm{disc}_{\ell_{01}}(\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}}) + \nu_{(h^*_{\mathcal{S}}, h^*_{\mathcal{T}})}\,, \tag{4.3}$$

où $h^*_{\mathcal{T}} = \operatorname{argmin}_{h\in\mathbb{H}} \mathrm{R}_{\mathcal{T}}(h)$ est la meilleure hypothèse cible et $h^*_{\mathcal{S}} = \operatorname{argmin}_{h\in\mathbb{H}} \mathrm{R}_{\mathcal{S}}(h)$ la meilleure hypothèse source, et $\nu_{(h^*_{\mathcal{S}}, h^*_{\mathcal{T}})} = \mathrm{d}_{\mathcal{S}_{\mathbb{X}}}(h^*_{\mathcal{S}}, h^*_{\mathcal{T}})$ est le désaccord $h^*_{\mathcal{S}}$ et $h^*_{\mathcal{T}}$.

2. Un ensemble $\mathbb{H}$ est symétrique si pour tout $h \in \mathbb{H}$, son opposé $-h$ est également dans $\mathbb{H}$.

L'Équation (4.3) peut être plus précise[4] que l'Équation (4.1) puisqu'elle majore l'écart entre l'erreur cible d'une hypothèse h et celle de la meilleure hypothèse $h^*_\mathcal{T}$ sur le domaine cible. En se basant sur le Théorème 4.2.2 et sur une analyse avec la complexité de Rademacher, MANSOUR et al. (2009a) ont dérivé une borne en généralisation sur le risque cible. Cette borne exprime un compromis entre le désaccord entre h et la meilleure hypothèse source $h^*_\mathcal{S}$, la complexité de $\mathbb{H}$, et — encore une fois — l'incapacité des hypothèses à détecter les différences entre les domaines.

Pour résumer, les bornes d'adaptation de domaine des Théorèmes 4.2.1 et 4.2.2 suggèrent que si la divergence entre les deux domaines est faible, une hypothèse de risque faible sur le domaine source peut amener à un risque faible sur le domaine cible. Les mesures de divergence entre domaines associées à ces résultats peuvent être vues comme la valeur dans le *pire cas* du désaccord entre les paires d'hypothèses.

4.3 Deux bornes d'adaptation pour le PAC-Bayes

L'originalité de nos travaux (GERMAIN et al., 2020) est la définition de deux cadres théoriques d'adaptation de domaine adaptés au PAC-Bayes. Notre première approche (Section 4.3.1, dont la première version a été publiée dans GERMAIN et al. (2013)) s'inscrit dans la philosophie de ces travaux fondateurs en prouvant une borne sur le risque de Gibbs (*i.e.* l'espérance des erreurs) et qui exprime un compromis similaire. Notre seconde approche (Section 4.3.2, publiée dans GERMAIN et al. (2016a)) apporte un point de vue différent et novateur en proposant un compromis original basé sur la décomposition du risque Gibbs de l'Équation (2.9).

4.3.1 Dans l'esprit des travaux fondateurs

4.3.1.1 Une mesure divergence entre domaines pour le PAC-Bayes

Alors que les bornes d'adaptation de domaine présentées dans la Section 4.2 se focalisent sur le risque d'une seule hypothèse de $\mathbb{H}$, nous nous focalisons ici sur l'espérance des risques des hypothèses de $\mathbb{H}$ selon la distribution *posterior* ρ, autrement dit, nous nous intéressons au risque de Gibbs. Pour les besoins du cadre PAC-Bayésien, nous définissons une pseudo-métrique[5] pour mesurer la différence structurelle entre les marginales des domaines en espérance selon la distribution *posterior* ρ sur $\mathbb{H}$. Comme nous nous intéressons à l'apprentissage d'un vote de majorité pondéré par ρ qui amène à de bonnes garanties en généralisation, nous proposons de suivre l'idée sous-jacente de la C-borne de l'Équation (2.14) : étant donné un domaine source $\mathcal{S}$, un domaine cible $\mathcal{T}$ et une distribution *posterior* ρ, si $\mathrm{R}_\mathcal{S}(G_\rho)$ et $\mathrm{R}_\mathcal{T}(G_\rho)$ sont proches, alors $\mathrm{R}_\mathcal{S}(\mathrm{MV}_\rho)$ et $\mathrm{R}_\mathcal{T}(\mathrm{MV}_\rho)$ sont proches lorsque $\mathrm{d}_\mathcal{S}(\rho)$ et $\mathrm{d}_\mathcal{T}(\rho)$ sont également proches. Ainsi, les domaines $\mathcal{S}$ et $\mathcal{T}$ sont proches selon ρ si le désaccord attendu sur les domaines tend à être proche. Nous appelons cette pseudo-métrique le *désaccord entre domaines* et nous la définissons comme suit.

4. Dans certains cas, l'Équation (4.1) peut conduire à un terme d'erreur 3 fois plus élevé que l'Équation (4.3) (voir MANSOUR et al., 2009a pour plus de détails).

5. Une pseudo-métrique d est une métrique pour laquelle la propriété $d(x,y) = 0 \iff x = y$ de désaccord entre domaines est relâchée pour devenir $d(x,y) = 0 \impliedby x = y$.

Définition 4.3.1 (Désaccord entre domaines)**.** Soit $\mathbb{H}$ un espace d'hypothèses. Pour tout domaine $\mathcal{S}$ et $\mathcal{T}$ sur $\mathbb{X}\times\mathbb{Y}$, toute distribution ρ sur $\mathbb{H}$, le désaccord entre domaines $\mathrm{dis}_\rho(\mathcal{S}_\mathbb{X}, \mathcal{T}_\mathbb{X})$ est défini par

$$\mathrm{dis}_\rho(\mathcal{S}_\mathbb{X}, \mathcal{T}_\mathbb{X}) \;=\; \left| \mathop{\mathbb{E}}_{(h,h')\sim\rho^2} \Big[\mathrm{d}_{\mathcal{T}_\mathbb{X}}(h,h') - \mathrm{d}_{\mathcal{S}_\mathbb{X}}(h,h') \Big] \right| \;=\; \Big| \mathrm{d}_\mathcal{T}(\rho) - \mathrm{d}_\mathcal{S}(\rho) \Big| .$$

4.3.1.2 Comparaison avec la $\mathcal{H}\Delta\mathcal{H}$-divergence

Alors que la $\mathcal{H}\Delta\mathcal{H}$-divergence du Théorème 4.2.1 est difficile à optimiser conjointement avec l'erreur empirique source, la version empirique de notre désaccord entre domaines est plus facile à manipuler : il suffit de calculer le désaccord moyen selon ρ plutôt que de chercher la paire d'hypothèses qui maximise le désaccord. En effet, $\mathrm{dis}_\rho(\mathcal{S}_\mathbb{X}, \mathcal{T}_\mathbb{X})$ dépend du *posterior* appris ρ, ce qui suggère que l'on peut directement le minimiser via son estimation empirique. Cette minimisation peut être réalisée dans l'espace d'origine sans aucune modification de l'espace d'hypothèses et/ou de l'importance des exemples. Au contraire, $\frac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{S}_\mathbb{X}, \mathcal{T}_\mathbb{X})$ exprime un *supremum* sur toutes les hypothèses $h \in \mathbb{H}$ et ne dépend donc pas de l'hypothèse pour laquelle l'erreur est considérée. De plus, $\mathrm{dis}_\rho(\mathcal{S}_\mathbb{X}, \mathcal{T}_\mathbb{X})$ ("le cas moyen" selon ρ) est plus petit que $\frac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{S}_\mathbb{X}, \mathcal{T}_\mathbb{X})$ (le "pire cas"). En effet, pour tout $h \in \mathbb{H}$ et ρ sur $\mathbb{H}$, on a :

$$\begin{aligned} \tfrac{1}{2}\, d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{S}_\mathbb{X}, \mathcal{T}_\mathbb{X}) \;&=\; \sup_{(h,h')\in\mathcal{H}^2} \Big| \mathrm{d}_{\mathcal{T}_\mathbb{X}}(h,h') - \mathrm{d}_{\mathcal{S}_\mathbb{X}}(h,h') \Big| \;\geq\; \mathop{\mathbb{E}}_{(h,h')\sim\rho^2} \Big| \mathrm{d}_{\mathcal{T}_\mathbb{X}}(h,h') - \mathrm{d}_{\mathcal{S}_\mathbb{X}}(h,h') \Big| \\ &\geq\; \mathrm{dis}_\rho(\mathcal{S}_\mathbb{X}, \mathcal{T}_\mathbb{X}). \end{aligned}$$

4.3.1.3 Une borne d'adaptation pour le risque de Gibbs

Nous énonçons maintenant notre borne d'adaptation de domaine dans le cadre PAC-Bayésien, qui dépend de notre mesure de désaccord entre domaines (Définition 4.3.1).

Théorème 4.3.1 (Borne 1 d'adaptation de domaine pour le risque de Gibbs)**.** Soit $\mathbb{H}$ un espace d'hypothèses. Pour tout domaine $\mathcal{S}$ et $\mathcal{T}$ sur $\mathbb{X}\times\mathbb{Y}$, on a

$$\forall \rho \text{ sur } \mathbb{H},\; \mathop{\mathbb{E}}_{h\sim\rho} \mathsf{R}_\mathcal{T}(h) = \mathsf{R}_\mathcal{T}(G_\rho) \;\leq\; \mathsf{R}_\mathcal{S}(G_\rho) + \frac{1}{2}\mathrm{dis}_\rho(\mathcal{S}_\mathbb{X}, \mathcal{T}_\mathbb{X}) + \lambda_\rho \,,$$

où λ_ρ est l'écart entre les erreurs jointes source et cible associées à G_ρ

$$\lambda_\rho \;=\; \Big| \mathrm{e}_\mathcal{T}(\rho) - \mathrm{e}_\mathcal{S}(\rho) \Big| . \tag{4.4}$$

À l'instar des bornes des Théorèmes 4.2.1 et 4.2.2, notre borne peut être interprétée comme un compromis entre différentes quantités. Les termes $\mathsf{R}_\mathcal{S}(G_\rho)$ et $\mathrm{dis}_\rho(\mathcal{S}_\mathbb{X}, \mathcal{T}_\mathbb{X})$ sont similaires aux deux premiers termes de la borne d'adaptation de domaine du Théorème 4.2.1 : $\mathsf{R}_\mathcal{S}(G_\rho)$ est le risque source moyen sur $\mathbb{H}$ pondéré par ρ, et $\mathrm{dis}_\rho(\mathcal{T}_\mathbb{X}, \mathcal{S}_\mathbb{X})$ mesure la différence entre les marginales via l'écart moyen pondéré par ρ des désaccords (entre paire d'hypothèses) source et cible, mais est spécifique au modèle considéré qui dépend de ρ. Le terme λ_ρ mesure l'écart entre les erreurs jointes cible et source associées à ρ. D'après cette théorie, une bonne adaptation de domaine est possible si cet écart est faible. Cependant, nous supposons que nous n'avons aucune étiquette cible, nous ne pouvons donc ni le contrôler, ni l'estimer. En

pratique, nous supposons que λ_ρ est faible et nous le négligeons. En d'autres termes, nous supposons que l'étiquetage sur le domaine source et l'étiquetage sur le domaine cible sont liés et que le désaccord entre domaines et les étiquettes source sont suffisants pour trouver une bonne distribution *posterior* ρ. Enfin, comme $\mathrm{dis}_\rho(\mathcal{T}_\mathbb{X}, \mathcal{S}_\mathbb{X})$ et λ_ρ dépendent de ρ, notre borne est, en général, incomparable avec celles des Théorèmes 4.2.1 et 4.2.2. Cependant, elle repose sur la même idée : en supposant que les domaines soient suffisamment liés, il faut chercher un modèle (ici, un *posterior)* qui minimise un compromis entre son risque source et une divergence entre les marginales des domaines.

4.3.2 Une vision originale de l'adaptation de domaine

Dans cette section, nous introduisons une approche originale pour majorer le risque de Gibbs sur le domaine cible $\mathcal{T}$ par un terme dépendant de la distribution marginale cible $\mathcal{T}_\mathbb{X}$, un terme dépendant du domaine source $\mathcal{S}$ et un terme capturant le "volume" de la distribution source non informatif pour la tâche cible. Notre résultat est basé sur l'Équation (2.9) qui, pour un domaine $\mathcal{T}$, décompose le risque de Gibbs en un compromis entre le désaccord $\frac{1}{2}\mathrm{d}_\mathcal{T}(\rho)$ et l'erreur jointe $\mathrm{e}_\mathcal{T}(\rho)$:

$$\mathsf{R}_\mathcal{T}(G_\rho) = \mathrm{e}_\mathcal{T}(\rho) + \frac{1}{2}\,\mathrm{d}_\mathcal{T}(\rho). \tag{2.9}$$

Un point clé est que désaccord peut se calculer sans les étiquettes : $\mathrm{d}_\mathcal{T}(\rho)$ est calculable à partir de la distribution marginale $\mathcal{T}_\mathbb{X}$. Dans notre contexte, les étiquettes cibles sont inconnues, nous avons donc accès à l'estimation empirique de $\mathrm{d}_\mathcal{T}(\rho)$, mais nous ne pouvons pas estimer $\mathrm{e}_\mathcal{T}(\rho)$. Par contre, l'erreur jointe peut être calculée sur le domaine source étiqueté, c'est ce que nous avons gardé à l'esprit pour définir une nouvelle divergence entre domaines.

4.3.2.1 Une autre divergence entre domaines pour le PAC-bayes

Nous définissons une divergence entre domaines qui permet de relier l'erreur jointe cible $\mathrm{e}_\mathcal{T}(\rho)$ à l'erreur jointe source $\mathrm{e}_\mathcal{S}(\rho)$ grâce à une pondération particulière. Nous appelons cette nouvelle divergence la β_q-divergence paramétrée par un réel $q > 0$:

$$\beta_q(\mathcal{T}\|\mathcal{S}) \;=\; \left[\underset{(\mathbf{x}^s,y^s)\sim\mathcal{S}}{\mathbb{E}}\left(\frac{\mathcal{T}(\mathbf{x},y)}{\mathcal{S}(\mathbf{x},y)}\right)^q\right]^{\frac{1}{q}}. \tag{4.5}$$

Certaines valeurs de q permettent de retrouver des divergences connues. Par exemple, avec $q=2$ on retrouve la distance χ^2 entre les domaines : $\beta_2(\mathcal{T}\|\mathcal{S}) = \sqrt{\chi^2(\mathcal{T}\|\mathcal{S})+1}$. Nous pouvons également relier $\beta_q(\mathcal{T}\|\mathcal{S})$ à la divergence de Rényi [6] (Erven et Harremoës, 2014) utilisée parfois pour l'*importance reweighting*. Nous notons $\beta_\infty(\mathcal{T}\|\mathcal{S})$ le cas limite $q\to\infty$:

$$\beta_\infty(\mathcal{T}\|\mathcal{S}) \;=\; \sup_{(\mathbf{x},y)\in\mathrm{SUPP}(\mathcal{S})}\left(\frac{\mathcal{T}(\mathbf{x},y)}{\mathcal{S}(\mathbf{x},y)}\right), \tag{4.6}$$

où $\mathrm{SUPP}(\mathcal{S})$ est le support de $\mathcal{S}$. La β_q-divergence capture les régions de l'espace d'entrée qui appartiennent à l'intersection entre le support du domaine source et le support du domaine cible. Il semble raisonnable de supposer que lorsque l'adaptation est possible, ces régions sont grandes. Or, il est probable que $\mathrm{SUPP}(\mathcal{T})$ ne soit pas entièrement inclus dans $\mathrm{SUPP}(\mathcal{S})$.

6. Pour $q \geq 0$, on peut montrer que $\beta_q(\mathcal{T}\|\mathcal{S}) = 2^{\frac{q-1}{q}\,\mathrm{D}_q(\mathcal{T}\|\mathcal{S})}$, où $\mathrm{D}_q(\cdot\|\cdot)$ est la divergence de Rényi.

Nous notons $\mathcal{T}\backslash\mathcal{S}$ la distribution de $(\mathbf{x}, y) \sim \mathcal{T}$ conditionnée par $(\mathbf{x}, y) \in \text{SUPP}(\mathcal{T}) \backslash \text{SUPP}(\mathcal{S})$. Puisque l'estimation de l'erreur jointe $\mathrm{e}_{\mathcal{T}\backslash\mathcal{S}}(\rho)$ sans faire d'hypothèse supplémentaire est complexe, nous définissons $\eta_{\mathcal{T}\backslash\mathcal{S}}$ comme le pire risque possible dans cette région inconnue :

$$\eta_{\mathcal{T}\backslash\mathcal{S}} = \Pr_{(\mathbf{x},y)\sim\mathcal{T}}\Big((\mathbf{x}, y) \notin \text{SUPP}(\mathcal{S})\Big) \sup_{h\in\mathbb{H}} \mathrm{R}_{\mathcal{T}\backslash\mathcal{S}}(h)\,. \tag{4.7}$$

Même si nous ne pouvons pas calculer $\sup_{h\in\mathbb{H}} \mathrm{R}_{\mathcal{T}\backslash\mathcal{S}}(h)$, la valeur de $\eta_{\mathcal{T}\backslash\mathcal{S}}$ est nécessairement plus petite que $\Pr_{(\mathbf{x},y)\sim\mathcal{T}}\left((\mathbf{x}, y) \notin \text{SUPP}(\mathcal{S})\right)$.

4.3.2.2 Une nouvelle borne d'adaptation de domaine

Le théorème ci-dessous présente notre nouvelle vision de l'adaptation.

Théorème 4.3.2 (Borne 2 d'adaptation de domaine pour le risque de Gibbs)**.** Soit $\mathbb{H}$ un espace d'hypothèses. Soit $q > 0$. Pour tout domaine $\mathcal{S}$ et $\mathcal{T}$ sur $\mathbb{X}\times\mathbb{Y}$, on a

$$\forall \rho \text{ sur } \mathbb{H}, \quad \mathrm{R}_{\mathcal{T}}(G_\rho) \;\leq\; \frac{1}{2}\,\mathrm{d}_{\mathcal{T}}(\rho) + \beta_q(\mathcal{T}\|\mathcal{S}) \times \Big[\mathrm{e}_{\mathcal{S}}(\rho)\Big]^{1-\frac{1}{q}} + \eta_{\mathcal{T}\backslash\mathcal{S}}\,,$$

où $\mathrm{d}_{\mathcal{T}}(\rho)$, $\mathrm{e}_{\mathcal{S}}(\rho)$, $\beta_q(\mathcal{T}\|\mathcal{S})$ et $\eta_{\mathcal{T}\backslash\mathcal{S}}$ sont respectivement définis dans les Équations (2.8), (2.7), (4.5) et (4.7).

La borne du Théorème 4.3.2 est atteinte si les domaines sont égaux ($\mathcal{S} = \mathcal{T}$). Ainsi, lorsque l'adaptation n'est pas nécessaire, notre analyse est toujours correcte et amène à un résultat non dégénéré :

$$\mathrm{R}_{\mathcal{S}}(G_\rho) \;=\; \mathrm{R}_{\mathcal{T}}(G_\rho) \;\leq\; \tfrac{1}{2}\,\mathrm{d}_{\mathcal{T}}(\rho) + 1 \times [\mathrm{e}_{\mathcal{S}}(\rho)]^1 + 0 \;=\; \tfrac{1}{2}\,\mathrm{d}_{\mathcal{S}}(\rho) + \mathrm{e}_{\mathcal{S}}(\rho) = \;\mathrm{R}_{\mathcal{S}}(G_\rho)\,.$$

Comme pour les résultats précédents, notre borne majore le risque cible par un compromis entre trois termes. Cependant, ces termes correspondent à des quantités atypiques :

(i) Le désaccord $\mathrm{d}_{\mathcal{T}}(\rho)$ capture une information de second degré sur le domaine cible (sans étiquette).

(ii) La β_q-divergence $\beta_q(\mathcal{T}\|\mathcal{S})$ n'est pas un terme additionnel : la divergence pondère l'influence/l'importance de l'erreur jointe source $\mathrm{e}_{\mathcal{S}}(\rho)$; le paramètre q permet de considérer différentes relations entre $\beta_q(\mathcal{T}\|\mathcal{S})$ et $\mathrm{e}_{\mathcal{S}}(\rho)$.

(iii) Le terme $\eta_{\mathcal{T}\backslash\mathcal{S}}$ quantifie la pire erreur cible possible dans les régions où le domaine source n'apporte aucune information sur le domaine cible. Dans ce travail, nous supposons que cette région est petite.

4.3.2.3 Relations avec des hypothèses d'adaptation de domaine

Nous détaillons ici les connexions avec des hypothèses classiques en adaptation de domaine. Ben-David et Urner (2012) ont présenté trois hypothèses, pouvant faciliter l'adaptation, afin de caractériser si une tâche d'adaptation est apprenable. Nous discutons ci-dessous de leur interprétation pour notre Théorème 4.3.2 ; il est important de préciser que notre résultat ne fait intervenir aucune de ces hypothèses et qu'il reste valide en leur absence.

À propos du *covariate-shift*. Une tâche d'adaptation satisfait les hypothèses du *covariate-shift* (Shimodaira, 2000) si les domaines diffèrent uniquement en leur marginale selon l'espace d'entrée (*i.e.*, $\mathcal{T}_{\mathbb{Y}|\mathbf{x}}(y) = \mathcal{S}_{\mathbb{Y}|\mathbf{x}}(y)$). Dans ce scénario, il est possible d'estimer la valeur

de $\beta_q(\mathcal{T}_{\mathbb{X}}\|\mathcal{S}_{\mathbb{X}})$, et même celle de $\eta_{\mathcal{T}\setminus\mathcal{S}}$, en utilisant des méthodes non supervisées d'estimation de densité. Notons qu'avec l'hypothèse supplémentaire que les domaines partagent le même support, on a $\eta_{\mathcal{T}\setminus\mathcal{S}} = 0$. Ainsi, on obtient

$$\mathrm{R}_{\mathcal{T}}(G_\rho) = \tfrac{1}{2}\mathrm{d}_{\mathcal{T}}(\rho) + \mathop{\mathbb{E}}_{\mathbf{x}^s\sim\mathcal{S}_{\mathbb{X}}} \frac{\mathcal{T}_{\mathbb{X}}(\mathbf{x})}{\mathcal{S}_{\mathbb{X}}(\mathbf{x})} \mathop{\mathbb{E}}_{h\sim\rho} \mathop{\mathbb{E}}_{h'\sim\rho} \mathrm{I}[h(\mathbf{x}) \neq y]\, \mathrm{I}[h'(\mathbf{x}) \neq y].$$

Ce résultat suggère que pour corriger le *shift* entre les domaines, une solution est de pondérer le domaine source (étiqueté) en considérant l'information du désaccord cible.

À propos du *weight ratio*. Le *weight ratio* (Ben-David et Urner, 2012) ("le rapport de poids" entre les domaines source et cible en français), est défini par

$$W_{\mathcal{B}}(\mathcal{S},\mathcal{T}) \;=\; \inf_{b\in\mathcal{B},\, \mathcal{T}_{\mathbb{X}}(b)\neq 0} \frac{\mathcal{S}_{\mathbb{X}}(b)}{\mathcal{T}_{\mathbb{X}}(b)}\,,$$

avec $\mathcal{B} \subseteq 2^{\mathbb{X}}$ une collection de sous-ensembles de l'espace d'entrée. Lorsque $W_{\mathcal{B}}(\mathcal{S},\mathcal{T})$ est borné loin de 0, l'adaptation devrait être possible sous l'hypothèse du *covariate-shift*. Dans ce contexte, si $\textsc{supp}(\mathcal{S}) = \textsc{supp}(\mathcal{T})$, le cas limite $\beta_\infty(\mathcal{T}\|\mathcal{S})$ est égal à l'inverse du "rapport de poids ponctuel" obtenu avec $\mathcal{B} = \{\mathbf{x} : \mathbf{x} \in \mathbb{X}\}$ dans $W_{\mathcal{B}}(\mathcal{S},\mathcal{T})$. En effet, β_q et $W_{\mathcal{B}}$ comparent les densités des domaines source et cible, mais offrent différentes stratégies pour relâcher "le rapport de poids ponctuel" ; la première en diminuant la valeur de q et la seconde en considérant des sous-espaces plus grands $\mathcal{B}$.

À propos de l'hypothèse de cluster. Un domaine cible satisfait l'*hypothèse de cluster* si les exemples de même étiquette appartiennent à une région commune de l'espace d'entrée, et si les régions différemment étiquetées sont bien séparées par des régions de faible densité (formalisé par la *probabilistic Lipschitzness* par Urner et al. (2011)). Une fois spécialisé aux classifieurs linéaires, $\mathrm{d}_{\mathcal{T}}(\rho)$ se comporte bien dans ce contexte (voir Section 4.5).

À propos de l'apprentissage de représentation. L'hypothèse principale sous-jacente à notre algorithme d'adaptation de domaine (voir Section 4.5 ci-après) est que le support du domaine cible est majoritairement inclus dans le support du domaine source, *i.e.*, la valeur de $\eta_{\mathcal{T}\setminus\mathcal{S}}$ est faible. Quand $\mathcal{T}\setminus\mathcal{S}$ est suffisamment grand pour empêcher l'adaptation, une solution consiste à réduire le volume de cet ensemble tout en veillant à préserver un bon compromis entre $\mathrm{d}_{\mathcal{T}}(\rho)$ et $\mathrm{e}_{\mathcal{S}}(\rho)$. Pour cela, il est possible d'utiliser des méthodes d'apprentissage de représentation pour projeter les exemples source et cible dans un nouvel espace commun (*e.g.*, Chen et al., 2012 ; Ganin et al., 2016)

4.3.3 Comparaisons des bornes

Puisqu'elles reposent sur des approximations différentes, l'écart entre les bornes des Théorèmes 4.3.1 et 4.3.2 varie en fonction du contexte. La principale différence entre nos bornes réside dans les termes estimables qui servent de fondement à nos algorithmes d'adaptation de domaine, décrits dans la Section 4.5. Dans le Théorème 4.3.2, les termes non estimables sont la divergence entre les domaines $\beta_q(\mathcal{T}\|\mathcal{S})$ et le terme $\eta_{\mathcal{T}\setminus\mathcal{S}}$. Contrairement au terme non contrôlable λ_ρ du Théorème 4.3.1, ces termes ne dépendent pas de la distribution apprise ρ : pour chaque ρ, les valeurs de $\beta_q(\mathcal{T}\|\mathcal{S})$ et $\eta_{\mathcal{T}\setminus\mathcal{S}}$ sont constantes et mesurent la relation entre les domaines pour la tâche considérée. De plus, contrairement à $\mathrm{dis}_\rho(\mathcal{S}_{\mathbb{X}},\mathcal{T}_{\mathbb{X}})+\lambda_\rho$ du Théorème 4.3.1, le fait que la β_q-divergence soit un terme multiplicatif et non un terme additif est une contribution du Théorème 4.3.2. Un des intérêts est que

$\beta_q(\mathcal{T}\|\mathcal{S})$ peut être considéré comme un hyperparamètre de contrôle du compromis entre le désaccord cible et l'erreur jointe source.

Lorsque $\mathrm{e}_{\mathcal{T}}(\rho) \geq \mathrm{e}_{\mathcal{S}}(\rho)$, nous pouvons majorer λ_ρ comme suit :

$$\mathrm{e}_{\mathcal{T}}(\rho) \;\geq\; \mathrm{e}_{\mathcal{S}}(\rho) \quad\Longrightarrow\quad \lambda_\rho = \mathrm{e}_{\mathcal{T}}(\rho) - \mathrm{e}_{\mathcal{S}}(\rho) \;\leq\; \beta_q(\mathcal{T}\|\mathcal{S}) \times \left[\mathrm{e}_{\mathcal{S}}(\rho)\right]^{1-\frac{1}{q}} + \eta_{\mathcal{T}\setminus\mathcal{S}} - \mathrm{e}_{\mathcal{S}}(\rho)\,.$$

Dans ce cas particulier, le Théorème 4.3.1 peut donc se réécrire pour tout ρ sur $\mathbb{H}$:

$$\mathrm{R}_{\mathcal{T}}(G_\rho) \leq \mathrm{R}_{\mathcal{S}}(G_\rho) + \frac{1}{2}\,\mathrm{dis}_\rho(\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}}) + \beta_q(\mathcal{T}\|\mathcal{S}) \times \Big[\mathrm{e}_{\mathcal{S}}(\rho)\Big]^{1-\frac{1}{q}} - \mathrm{e}_{\mathcal{S}}(\rho) + \eta_{\mathcal{T}\setminus\mathcal{S}}\,.$$

Il s'avère que lorsque $\mathrm{d}_{\mathcal{T}}(\rho) \geq \mathrm{d}_{\mathcal{S}}(\rho)$ et $\mathrm{e}_{\mathcal{T}}(\rho) \geq \mathrm{e}_{\mathcal{S}}(\rho)$, la borne ci-dessus se simplifie pour correspondre à celle du Théorème 4.3.2. Cela se produit dans le cas très particulier où le désaccord cible et l'erreur jointe cible sont supérieurs à leurs homologues sources, ce qui peut être interprété comme une situation plutôt favorable. Dans tous les autres cas, le Théorème 4.3.2 est plus précis. Cela s'explique de la manière suivante. Pour prouver le Théorème 4.3.1, nous avons suivi la technique de preuve de l'analyse classique de l'adaptation de domaine qui fait intervenir l'introduction "artificielle" d'une valeur absolue. Cette approche peut, en fait, conduire à une approximation grossière. En revanche, pour prouver le Théorème 4.3.2, nous avons adopté une démarche différente en exploitant directement la définition du risque de Gibbs, nous permettant d'obtenir une analyse plus appropriée pour le PAC-Bayes. Les expériences que nous avons réalisées confirment cette différence de comportement et la supériorité analytique du Théorème 4.3.2 par rapport au Théorème 4.3.1 (Germain et al., 2020).

4.4 Bornes en généralisation PAC-Bayésiennes

Pour calculer nos bornes d'adaptation de domaine, il est nécessaire de connaître les distributions $\mathcal{S}$ et $\mathcal{T}_{\mathbb{X}}$, ce qui n'est jamais le cas pour des tâches réelles. Nous utilisons la théorie PAC-Bayésienne pour convertir les bornes des Théorèmes 4.3.1 et 4.3.2 en bornes en généralisation sur le risque de Gibbs cible. Ces bornes sont calculables à partir d'une paire d'échantillons source-cible $(\mathbb{S}, \mathbb{T}) \sim \mathcal{S}^{m_s} \times \mathcal{T}_{\mathbb{X}}^{m_t}$. Pour obtenir ces résultats, nous avons dérivé des bornes en généralisation "à la Catoni" (Théorème 2.4.7) pour les termes intervenant dans les bornes d'adaptation : $\mathrm{d}_{\mathcal{T}}(\rho)$, $\mathrm{e}_{\mathcal{S}}(\rho)$ et $\mathrm{dis}_\rho(\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}})$. Nous avons ensuite combiné ces bornes pour obtenir les bornes en généralisation suivantes. Nous commençons par énoncer la borne en généralisation associée à la borne d'adaptation de domaine du Théorème 4.3.1.

Théorème 4.4.1 (Borne PAC-Bayésienne 1 pour l'adaptation de domaine)**.** Étant donné les domaines source $\mathcal{S}$ et cible $\mathcal{T}$ sur $\mathbb{X}\times\mathbb{Y}$, un ensemble d'hypothèses $\mathbb{H}$, pour toute distribution $\pi \in \mathbb{M}^*(\mathbb{H})$, pour tout $\omega > 0$ et $a > 0$, pour tout $\delta \in]0, 1]$, on a

$$\mathop{\mathbb{P}}_{(\mathbb{S}\times\mathbb{T})\sim(\mathcal{S}\times\mathcal{T}_{\mathbb{X}})^m}\left[\begin{array}{l}\forall \rho \in \mathbb{M}(\mathbb{H}),\ \mathrm{R}_{\mathcal{T}}(G_\rho) \leq \omega' \widehat{\mathrm{R}}_{\mathbb{S}}(G_\rho) + a'\dfrac{1}{2}\widehat{\mathrm{dis}}_\rho(\mathbb{S}, \mathbb{T}) \\ \qquad\qquad + \left[\dfrac{\omega'}{\omega} + \dfrac{a'}{a}\right]\dfrac{\mathrm{KL}(\rho\|\pi) + \ln\frac{3}{\delta}}{m} + \lambda_\rho + \dfrac{1}{a'-1}\end{array}\right] \geq 1-\delta,$$

où $\omega' = \frac{\omega}{1-e^{-\omega}}$ et $a' = \frac{2a}{1-e^{-2a}}$ et $\lambda_\rho = |\mathrm{e}_{\mathcal{T}}(\rho) - \mathrm{e}_{\mathcal{S}}(\rho)|$, et $\widehat{\mathrm{R}}_{\mathbb{S}}(G_\rho)$ et $\widehat{\mathrm{dis}}_\rho(\mathbb{S}, \mathbb{T})$ sont les estimations empiriques du risque source et du désaccord entre domaines.

Nous énonçons maintenant la borne en généralisation associée au Théorème 4.3.2. Pour des raisons de simplification algorithmique, nous ne considérons que le cas où $q \to \infty$.

Théorème 4.4.2 (Borne PAC-Bayésienne 2 pour l'adaptation de domaine)**.** Étant donné les domaines source $\mathcal{S}$ et cible $\mathcal{T}$ sur $\mathbb{X}\times\mathbb{Y}$, un ensemble d'hypothèses $\mathbb{H}$, pour toute distribution $\pi \in \mathbb{M}^*(\mathbb{H})$, pour tout $b>0$ et $c>0$, pour tout $\delta\in]0,1]$, on a

$$\mathop{\mathbb{P}}_{(\mathbb{S}\times\mathbb{T})\sim\mathcal{S}^{m_s}\times\mathcal{T}_{\mathbb{X}}^{m_t}}\left[\begin{array}{l}\forall \rho \in \mathbb{M}(\mathbb{H}),\ \mathsf{R}_{\mathcal{T}}(G_\rho) \ \le c'\,\frac{1}{2}\,\widehat{\mathrm{d}}_{\mathbb{T}}(\rho) + b'\,\widehat{\mathrm{e}}_{\mathbb{S}}(\rho) \\ \qquad\qquad + \left[\frac{c'}{m_t\,c}+\frac{b'}{m_s\,b}\right]\left[2\,\mathrm{KL}(\rho\|\pi)+\ln\tfrac{2}{\delta}\right] + \eta_{\mathcal{T}\setminus\mathcal{S}}\end{array}\right] \ge 1-\delta,$$

où $b'=\frac{b}{1-e^{-b}}\beta_\infty(\mathcal{T}\|\mathcal{S})$ et $c'=\frac{c}{1-e^{-c}}$ et $\eta_{\mathcal{T}\setminus\mathcal{S}}$ est défini dans l'Équation (4.7) ; $\widehat{\mathrm{d}}_{\mathbb{T}}(\rho)$ et $\widehat{\mathrm{e}}_{\mathbb{S}}(\rho)$ sont les estimations empiriques du désaccord cible et de l'erreur jointe source.

D'un point de vue optimisation, le problème suggéré par la borne du Théorème 4.4.2 est plus simple à minimiser que celle du Théorème 4.4.1. La première est plus régulière que la seconde : la valeur absolue due au désaccord entre les domaines $\mathrm{dis}_\rho(\mathcal{S}_{\mathbb{X}},\mathcal{T}_{\mathbb{X}})$ disparaît au profit de la divergence entre les domaines $\beta_\infty(\mathcal{T}\|\mathcal{S})$ qui est une constante pouvant être considérée comme un hyperparamètre de l'algorithme. En outre, le Théorème 4.4.1 exige des tailles d'échantillons source et cible égales tandis que le Théorème 4.4.2 autorise différentes tailles, *i.e.*, $m_s \neq m_t$. De plus, pour des raisons algorithmiques, nous ignorons le terme λ_ρ (du Théorème 4.4.1) non constant et dépendant de ρ. Dans notre seconde analyse, un tel compromis n'est pas obligatoire pour appliquer le résultat théorique puisque le terme non estimable $\eta_{\mathcal{T}\setminus\mathcal{S}}$ est constant et ne dépend pas du *posterior* ρ appris. Nous ignorons donc $\eta_{\mathcal{T}\setminus\mathcal{S}}$ sans impact sur le problème d'optimisation. Par ailleurs, il est assez réaliste de supposer que $\eta_{\mathcal{T}\setminus\mathcal{S}}$ est faible dans les situations où les supports source et cible sont similaires.

4.5 Adaptation de domaine PAC-Bayésienne spécialisée aux classifieurs linéaires

Dans cette section, nous présentons deux algorithmes pour l'adaptation de domaine inspirés par l'algorithme PAC-Bayésien PBGD3 de Germain et al. (2009). L'idée est de spécialiser les bornes en généralisation PAC-Bayésiennes de la section précédente aux classifieurs linéaires. L'approche adoptée est celle privilégiée dans de nombreux travaux PAC-Bayésiens (*e.g.*, Langford et Shawe-Taylor, 2002 ; Ambroladze et al., 2006 ; Germain et al., 2009 ; McAllester et Keshet, 2011 ; Parrado-Hernández et al., 2012a ; Germain et al., 2013), car elle permet de faire coïncider le risque du classifieur linéaire et le risque du vote de majorité, tout en favorisant en même temps des classifieurs à vaste marge.

4.5.1 L'astuce pour spécialiser [7]

Ici, $\mathbb{H}$ est un ensemble de classifieurs linéaires dans un espace de dimension d. Chaque $h_{\mathbf{w}'} \in \mathbb{H}$ est défini par un vecteur de poids $\mathbf{w}' \in \mathbb{R}^{\mathrm{d}}$, tel que $h_{\mathbf{w}'}(\mathbf{x}) = \mathrm{sign}\,(\mathbf{w}' \cdot \mathbf{x})$, où $\cdot$ est le produit scalaire. Langford et Shawe-Taylor (2002) ont spécialisé la théorie PAC-Bayésienne pour n'importe quel classifieur linéaire $h_{\mathbf{w}} \in \mathbb{H}$. Étant donné une distribution *prior* $\pi_{\mathbf{0}}$ et une

7. Voir Germain et al. (2020) pour plus de détails sur la spécialisation aux classifieurs linéaires.

distribution *posterior* $\rho_{\mathbf{w}}$ définies comme des distributions gaussiennes sphériques (de matrice de covariance l'identité centrée sur les vecteurs $\mathbf{0}$ et $\mathbf{w}$), pour tout $h_{\mathbf{w}'} \in \mathbb{H}$, on a

$$\pi_{\mathbf{0}}(h_{\mathbf{w}'}) = \left(\frac{1}{\sqrt{2\pi}}\right)^{\mathrm{d}} \exp\left(-\frac{\|\mathbf{w}'\|^2}{2}\right), \quad \text{et} \quad \rho_{\mathbf{w}}(h_{\mathbf{w}'}) = \left(\frac{1}{\sqrt{2\pi}}\right)^{\mathrm{d}} \exp\left(-\frac{\|\mathbf{w}'-\mathbf{w}\|^2}{2}\right).$$

Une propriété de ces distributions — considérées comme des distributions normales multivariées $\pi_{\mathbf{0}} = \mathcal{N}(\mathbf{0}, \mathbf{I})$ et $\rho_{\mathbf{w}} = \mathcal{N}(\mathbf{w}, \mathbf{I})$ — est que la prédiction du vote de majorité $\mathsf{MV}_{\rho_{\mathbf{w}}}$ pondéré par $\rho_{\mathbf{w}}$ coïncide avec celle du classifieur linéaire $h_{\mathbf{w}}$. En effet, on a

$$\forall\, \mathbf{x} \in \mathbb{X},\ \forall\, \mathbf{w} \in \mathbb{H}, \quad h_{\mathbf{w}}(\mathbf{x}) \ = \ \mathsf{MV}_{\rho_{\mathbf{w}}}(\mathbf{x}) \ = \ \mathrm{sign}\left[\mathop{\mathbb{E}}_{h_{\mathbf{w}'}\sim\rho_{\mathbf{w}}} h_{\mathbf{w}'}(\mathbf{x})\right].$$

Le risque de Gibbs sur un domaine $\mathcal{D}$ est alors donné par [8]

$$\mathsf{R}_{\mathcal{D}}(G_{\rho_{\mathbf{w}}}) = \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} \Phi_{\mathsf{R}}\left(y\,\frac{\mathbf{w}\cdot\mathbf{x}}{\|\mathbf{x}\|}\right), \quad \text{où } \Phi_{\mathsf{R}}(x) = \ \tfrac{1}{2}\left[1 - \mathrm{Erf}\left(\tfrac{x}{\sqrt{2}}\right)\right], \tag{4.8}$$

avec $\mathrm{Erf}()$ la fonction d'erreur de Gauss définie par $\mathrm{Erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x \exp\left(-t^2\right) \mathrm{d}t$. Ici, $\Phi_{\mathsf{R}}(x)$, parfois appelée la *probit-loss* (*e.g.*, McAllester et Keshet, 2011), est vue comme une relaxation lisse de la fonction perte 0-1 dépendant de $y\frac{\mathbf{w}\cdot\mathbf{x}}{\|\mathbf{x}\|}$. Notons que $\|\mathbf{w}\|$ joue un rôle important sur la valeur de $\mathsf{R}_{\mathcal{D}}(G_{\rho_{\mathbf{w}}})$ mais pas sur celle de $\mathsf{R}_{\mathcal{D}}(h_{\mathbf{w}})$. En effet, $\mathsf{R}_{\mathcal{D}}(G_{\rho_{\mathbf{w}}})$ tend vers $\mathsf{R}_{\mathcal{D}}(h_{\mathbf{w}})$ lorsque $\|\mathbf{w}\|$ augmente, ce qui peut amener à des bornes très précises (Ambroladze et al., 2006 ; Germain et al., 2009). La KL-divergence entre $\rho_{\mathbf{w}}$ et $\pi_{\mathbf{0}}$ est

$$\mathrm{KL}\left(\rho_{\mathbf{w}} \| \pi_{\mathbf{0}}\right) \ = \ \mathrm{KL}\left(\mathcal{N}(\mathbf{w}, \mathbf{I}) \,\|\, \mathcal{N}(\mathbf{0}, \mathbf{I})\right) = \ \tfrac{1}{2}\|\mathbf{w}\|^2.$$

Le désaccord $\mathrm{d}_{\mathcal{D}}(\rho_{\mathbf{w}})$ et l'erreur jointe $\mathrm{e}_{\mathcal{D}}(\rho_{\mathbf{w}})$ sont

$$\mathrm{d}_{\mathcal{D}}(\rho_{\mathbf{w}}) \ = \ 2 \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}} \Phi_{\mathsf{R}}\left(\frac{\mathbf{w}\cdot\mathbf{x}}{\|\mathbf{x}\|}\right) \Phi_{\mathsf{R}}\left(-\frac{\mathbf{w}\cdot\mathbf{x}}{\|\mathbf{x}\|}\right) \ = \ \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}} \Phi_{\mathrm{d}}\left(\frac{\mathbf{w}\cdot\mathbf{x}}{\|\mathbf{x}\|}\right), \tag{4.9}$$

$$\text{et} \quad \mathrm{e}_{\mathcal{D}}(\rho_{\mathbf{w}}) \ = \ \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} \left[\Phi_{\mathsf{R}}\left(y\,\frac{\mathbf{w}\cdot\mathbf{x}}{\|\mathbf{x}\|}\right)\right]^2 \ = \ \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} \Phi_{\mathrm{e}}\left(y\,\frac{\mathbf{w}\cdot\mathbf{x}}{\|\mathbf{x}\|}\right), \tag{4.10}$$

avec $\Phi_{\mathrm{d}}(x) = 2\,\Phi_{\mathsf{R}}(x)\,\Phi_{\mathsf{R}}(-x)$ et $\Phi_{\mathrm{e}}(x) = [\Phi_{\mathsf{R}}(x)]^2$; ces fonctions sont ici vues comme des fonctions pertes pour les classifieurs linéaires (voir la Figure 4.1a). Ainsi, le désaccord entre domaines est

$$\mathrm{dis}_{\rho_{\mathbf{w}}}(\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}}) = \left| \mathop{\mathbb{E}}_{\mathbf{x}^s\sim\mathcal{S}_{\mathbb{X}}} \Phi_{\mathrm{d}}\left(\frac{\mathbf{w}\cdot\mathbf{x}^s}{\|\mathbf{x}^s\|}\right) - \mathop{\mathbb{E}}_{\mathbf{x}^t\sim\mathcal{T}_{\mathbb{X}}} \Phi_{\mathrm{d}}\left(\frac{\mathbf{w}\cdot\mathbf{x}^t}{\|\mathbf{x}^t\|}\right) \right|. \tag{4.11}$$

4.5.2 Spécialisation des bornes

Les Théorèmes 4.3.1 et 4.3.2 (quand $q \to \infty$) spécialisés aux classifieurs linéaires impliquent les bornes suivantes. Rappelons que $\mathsf{R}_{\mathcal{T}}(h_{\mathbf{w}}) = \mathsf{R}_{\mathcal{T}}(\mathsf{MV}_{\rho_{\mathbf{w}}}) \leq 2\,\mathsf{R}_{\mathcal{T}}(G_{\rho_{\mathbf{w}}})$.

8. Les calculs menant à l'Équation (4.8) peuvent être trouvés dans Langford (2005).

Corollaire 4.5.1 (Bornes d'adaptation de domaine pour les classifieurs linéaires)**.** Soit $\mathcal{S}$ et $\mathcal{T}$ les domaines source et cible sur $\mathbb{X}\times\mathbb{Y}$. Pour tout $\mathbf{w}\in\mathbb{R}$, on a

$$\mathrm{R}_{\mathcal{T}}(h_{\mathbf{w}}) \;\leq\; 2\,\mathrm{R}_{\mathcal{S}}(G_{\rho_{\mathbf{w}}}) + \mathrm{dis}_{\rho_{\mathbf{w}}}(\mathcal{S}_{\mathbb{X}},\mathcal{T}_{\mathbb{X}}) + 2\lambda_{\rho_{\mathbf{w}}}\,, \tag{4.12}$$

$$\text{et}\quad \mathrm{R}_{\mathcal{T}}(h_{\mathbf{w}}) \;\leq\; \mathrm{d}_{\mathcal{T}_{\mathbb{X}}}(\rho_{\mathbf{w}}) + 2\,\beta_\infty(\mathcal{T}\|\mathcal{S})\times \mathrm{e}_{\mathcal{S}}(\rho_{\mathbf{w}}) + 2\,\eta_{\mathcal{T}\setminus\mathcal{S}}\,, \tag{4.13}$$

où $\mathrm{dis}_{\rho_{\mathbf{w}}}(\mathcal{S}_{\mathbb{X}},\mathcal{T}_{\mathbb{X}})$, $\lambda_{\rho_{\mathbf{w}}}$ $\mathrm{d}_{\mathcal{T}_{\mathbb{X}}}(\rho_{\mathbf{w}})$, $\mathrm{e}_{\mathcal{S}}(\rho_{\mathbf{w}})$, $\beta_\infty(\mathcal{T}\|\mathcal{S})$ et $\eta_{\mathcal{T}\setminus\mathcal{S}}$ sont respectivement définis dans les Équations (4.11), (4.4), (4.9), (4.10), (4.6) et (4.7).

Dans l'Équation (4.13), pour des valeurs fixées de $\beta_\infty(\mathcal{T}\|\mathcal{S})$ et $\eta_{\mathcal{T}\setminus\mathcal{S}}$, le risque cible $\mathrm{R}_{\mathcal{T}}(h_{\mathbf{w}})$ est majoré par la somme pondérée par $\beta_\infty(\mathcal{T}\|\mathcal{S})$ des deux fonctions pertes. La perte $\Phi_{\mathrm{e}}()$ (l'erreur jointe) est calculée à partir du domaine source étiqueté ; elle vise à étiqueter correctement les exemples source, mais est plus permissive sur la marge requise que la perte $\Phi()$ (le risque de Gibbs). La perte $\Phi_{\mathrm{d}}()$ (le désaccord cible) est calculé à partir du domaine cible non étiqueté ; elle favorise une vaste marge cible (non signée). Ainsi, si un domaine cible satisfait l'hypothèse de cluster (Section 4.3.2.3), $\mathrm{d}_{\mathcal{T}_{\mathbb{X}}}(\rho_{\mathbf{w}})$ sera petit lorsque la frontière de décision traverse une région de faible densité entre les clusters étiquetés. L'Équation (4.13) reflète donc du fait que certaines erreurs sur le domaine source peuvent être autorisées si la séparation des données dans le domaine cible est améliorée. La Figure 4.1a donne une interprétation géométrique des bornes des Équations (4.12) et (4.13).

4.5.3 Bornes en généralisation et algorithmes

4.5.3.1 Un premier algorithme d'adaptation de domaine (PBDA).

Le Théorème 4.4.1 spécialisé aux classifieurs linéaires implique le résultat suivant.

Corollaire 4.5.2 (Borne PAC-Bayésienne 1 pour l'adaptation de domaine de classifieurs linéaires)**.** Étant donné les domaines source $\mathcal{S}$ et cible $\mathcal{T}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout $\omega>0$ et $a>0$, pour tout $\delta\in]0,1]$, on a,

$$\mathop{\mathbb{P}}_{(\mathbb{S}\times\mathbb{T})\sim(\mathcal{S}\times\mathcal{T}_{\mathbb{X}})^m}\left[\begin{array}{rl}\forall\mathbf{w}\in\mathbb{R},\ \mathrm{R}_{\mathcal{T}}(h_{\mathbf{w}}) \;\leq\; & 2\,\omega'\widehat{\mathrm{R}}_{\mathbb{S}}(G_{\rho_{\mathbf{w}}}) + a'\widehat{\mathrm{dis}}_{\rho_{\mathbf{w}}}(\mathbb{S},\mathbb{T}) + 2\,\lambda_{\rho_{\mathbf{w}}} \\ & +2\left(\dfrac{\omega'}{\omega}+\dfrac{a'}{a}\right)\dfrac{\|\mathbf{w}\|^2+\ln\frac{3}{\delta}}{m} + (a'-1)\end{array}\right] \geq 1-\delta,$$

où $\omega'=\frac{\omega}{1-e^{-\omega}}$, et $a'=\frac{2a}{1-e^{-2a}}$, et $\lambda_{\rho_{\mathbf{w}}}=|\mathrm{e}_{\mathcal{T}}(\rho_{\mathbf{w}})-\mathrm{e}_{\mathcal{S}}(\rho_{\mathbf{w}})|$, et $\widehat{\mathrm{R}}_{\mathbb{S}}(G_{\rho_{\mathbf{w}}})$ et $\widehat{\mathrm{dis}}_{\rho_{\mathbf{w}}}(\mathbb{S},\mathbb{T})$ sont les valeurs empiriques du risque source et du désaccord entre domaines.

Soit un échantillon source $\mathbb{S}=\{(\mathbf{x}_i^s,y_i^s)\}_{i=1}^m$ et un échantillon cible $\mathbb{T}=\{(\mathbf{x}_i^t)\}_{i=1}^m$, nous nous focalisons sur la minimisation de la borne du Corollaire 4.5.2. Nous rappelons que nous supposons que le terme $\lambda_{\rho_{\mathbf{w}}}$ est négligeable. Ainsi, la distribution *posterior* $\rho_{\mathbf{w}}$ qui minimise la borne sur $\mathrm{R}_{\mathcal{T}}(h_{\mathbf{w}})$ correspond à celle qui minimise

$$\begin{aligned}&\Omega\,m\,\widehat{\mathrm{R}}_{\mathbb{S}}(G_{\rho_{\mathbf{w}}}) + A\,m\,\widehat{\mathrm{dis}}_{\rho_{\mathbf{w}}}(\mathbb{S},\mathbb{T}) + \mathrm{KL}(\rho_{\mathbf{w}}\|\pi_{\mathbf{0}}) \\ &\quad= \Omega\sum_{i=1}^m \Phi_{\mathrm{R}}\left(y_i^s\frac{\mathbf{w}\cdot\mathbf{x}_i^s}{\|\mathbf{x}_i^s\|}\right) + A\left|\sum_{i=1}^m\left[\Phi_{\mathrm{d}}\left(\frac{\mathbf{w}\cdot\mathbf{x}_i^s}{\|\mathbf{x}_i^s\|}\right)-\Phi_{\mathrm{d}}\left(\frac{\mathbf{w}\cdot\mathbf{x}_i^t}{\|\mathbf{x}_i^t\|}\right)\right]\right| + \frac{1}{2}\|\mathbf{w}\|^2\,.\end{aligned} \tag{4.14}$$

Les valeurs $\Omega>0$ et $A>0$ sont les hyperparamètres de l'algorithme. Les constantes ω et a du Théorème 4.4.1 peuvent être retrouvées pour tout Ω et A. L'Équation (4.14) est

difficile à optimiser par descente de gradient puisqu'elle fait intervenir une valeur absolue et est fortement non convexe. Pour contrer cet inconvénient, nous remplaçons la fonction perte $\Phi_{\mathsf{R}}()$ par sa relaxation convexe $\widetilde{\Phi}_{\mathsf{R}}()$ définie par :

$$\widetilde{\Phi}_{\mathsf{R}}(x) \;=\; \max\left\{\Phi_{\mathsf{R}}(x),\, \frac{1}{2} - \frac{x}{\sqrt{2\pi}}\right\} \;=\; \begin{cases} \dfrac{1}{2} - \dfrac{x}{\sqrt{2\pi}} & \text{si } x \leq 0, \\ \Phi_{\mathsf{R}}(x) & \text{sinon.} \end{cases} \tag{4.15}$$

La dérivée de $\widetilde{\Phi}_{\mathsf{R}}()$ en x est $\widetilde{\Phi}'_{\mathsf{R}}(x) = \Phi'_{\mathsf{R}}\left(\max\{0, x\}\right)$, *i.e.*, $\widetilde{\Phi}'_{\mathsf{R}}(x) = -1/\sqrt{2\pi}$ si $x < 0$, et $\Phi'_{\mathsf{R}}(x)$ sinon. Notons que $\widetilde{\Phi}_{\mathsf{R}}()$ peut être interprétée comme une version lisse la fonction perte hinge $\max\{0, 1-x\}$ des SVM. Empiriquement, nous avons mis en évidence que le minimum de $\Phi_{\mathsf{R}}()$ et $\widetilde{\Phi}_{\mathsf{R}}()$ coïncident généralement (Germain et al., 2020). Bien que $\Phi_{\mathrm{d}}()$ soit quasi-concave et implique une tâche d'optimisation non convexe, notre étude empirique a montré qu'il est inutile d'effectuer plusieurs redémarrages lors de la descente de gradient pour trouver une solution adéquate. Nous appelons cet algorithme d'adaptation de domaine PBDA *(PAC-Bayesian Domain Adaptation)*.

Pour résumer, étant donné un échantillon source $\mathbb{S} = \{(\mathbf{x}_i^s, y_i^s)\}_{i=1}^m$, un échantillon cible non étiqueté $\mathbb{T} = \{(\mathbf{x}_i^t)\}_{i=1}^m$, et les hyperparamètres Ω et A, l'algorithme PBDA réalise une descente de gradient pour minimiser la fonction objectif :

$$F_{\mathsf{PBDA}}(\mathbf{w}) = \Omega \sum_{i=1}^{m} \widetilde{\Phi}_{\mathsf{R}}\left(y_i^s \frac{\mathbf{w}\cdot\mathbf{x}_i^s}{\|\mathbf{x}_i^s\|}\right) + A\left|\sum_{i=1}^{m}\left[\Phi_{\mathrm{d}}\left(\frac{\mathbf{w}\cdot\mathbf{x}_i^s}{\|\mathbf{x}_i^s\|}\right) - \Phi_{\mathrm{d}}\left(\frac{\mathbf{w}\cdot\mathbf{x}_i^t}{\|\mathbf{x}_i^t\|}\right)\right]\right| + \frac{1}{2}\|\mathbf{w}\|^2, \tag{4.16}$$

avec $\widetilde{\Phi}_{\mathsf{R}}(x) = \max\left\{\Phi_{\mathsf{R}}(x),\, \frac{1}{2} - \frac{x}{\sqrt{2\pi}}\right\}$ et $\Phi_{\mathrm{d}}(x) = 2\,\Phi_{\mathsf{R}}(x)\,\Phi_{\mathsf{R}}(-x)$. La Figure 4.1a montre le comportement des fonctions pertes. Le gradient de l'Équation (4.16) est

$$\nabla F_{\mathsf{PBDA}}(\mathbf{w}) = \Omega \sum_{i=1}^{m} \widetilde{\Phi}'_{\mathsf{R}}\left(\frac{y_i^s \mathbf{w}\cdot\mathbf{x}_i^s}{\|\mathbf{x}_i^s\|}\right)\frac{y_i^s \mathbf{x}_i^s}{\|\mathbf{x}_i^s\|} + s\times A\left(\sum_{i=1}^{m}\left[\Phi'_{\mathrm{d}}\left(\frac{\mathbf{w}\cdot\mathbf{x}_i^t}{\|\mathbf{x}_i^t\|}\right)\frac{\mathbf{x}_i^t}{\|\mathbf{x}_i^t\|} - \Phi'_{\mathrm{d}}\left(\frac{\mathbf{w}\cdot\mathbf{x}_i^s}{\|\mathbf{x}_i^s\|}\right)\frac{\mathbf{x}_i^s}{\|\mathbf{x}_i^s\|}\right]\right) + \mathbf{w},$$

avec $s = \operatorname{sign}\left(\sum_{i=1}^{m}\left[\Phi_{\mathrm{d}}\left(\frac{\mathbf{w}\cdot\mathbf{x}_i^s}{\|\mathbf{x}_i^s\|}\right) - \Phi_{\mathrm{d}}\left(\frac{\mathbf{w}\cdot\mathbf{x}_i^t}{\|\mathbf{x}_i^t\|}\right)\right]\right)$.
Dans l'article original, nous avons étendu PBDA à des fonctions noyaux.

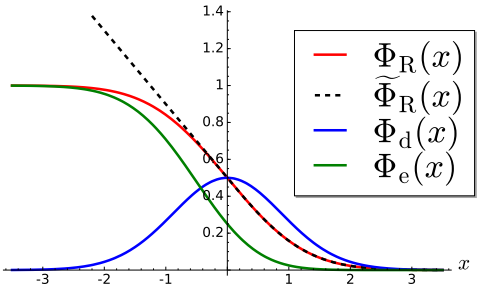
4.5.3.2 Un second algorithme d'adaptation de domaine (DALC)

Nous spécialisons maintenant le Théorème 4.4.2 aux classifieurs linéaires.

Corollaire 4.5.3 (Borne PAC-Bayésienne 2 pour l'adaptation de domaine de classifieurs linéaires)**.** Étant donné les domaines source $\mathcal{S}$ et cible $\mathcal{T}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout $b > 0$ et $c > 0$, pour tout $\delta \in]0, 1]$, on a

$$\mathop{\mathbb{P}}_{(\mathbb{S}\times\mathbb{T})\sim\mathcal{S}^{m_s}\times\mathcal{T}_{\mathbb{X}}^{m_t}}\left[\forall \mathbf{w}\in\mathbb{R},\; \mathsf{R}_{\mathcal{T}}(h_{\mathbf{w}}) \;\leq\; c'\,\widehat{\mathrm{d}}_{\mathbb{T}}(\rho_{\mathbf{w}}) + 2\,b'\,\widehat{\mathrm{e}}_{\mathbb{S}}(\rho_{\mathbf{w}}) + 2\,\eta_{\mathcal{T}\setminus\mathcal{S}} + 2\left(\frac{c'}{m_t\times c} + \frac{b'}{m_s\times b}\right)\left(\|\mathbf{w}\|^2 + \ln\tfrac{2}{\delta}\right)\right] \geq 1-\delta,$$

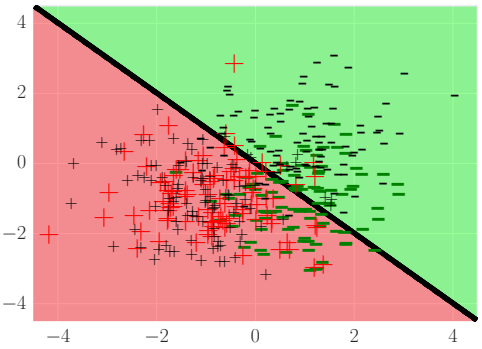
où $b' = \frac{b}{1-e^{-b}}\beta_\infty(\mathcal{T}\|\mathcal{S})$, et $c' = \frac{c}{1-e^{-c}}$, et $\eta_{\mathcal{T}\setminus\mathcal{S}}$ est défini dans l'Équation (4.7), et $\widehat{\mathrm{d}}_{\mathbb{T}}(\rho_{\mathbf{w}})$ et $\widehat{\mathrm{e}}_{\mathbb{S}}(\rho_{\mathbf{w}})$ sont les valeurs empiriques du désaccord cible et de l'erreur jointe source.



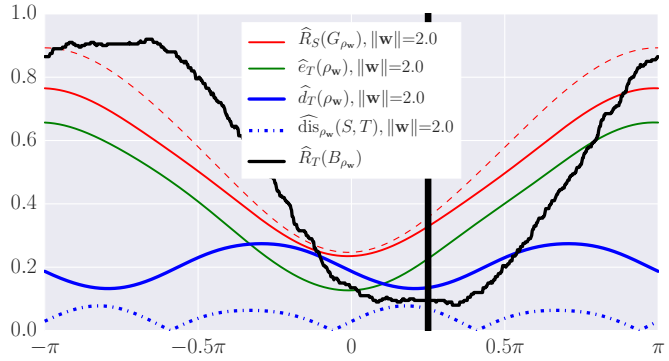
(a) Les fonctions pertes associées à la spécialisation aux classifieurs linéaires.

FONCTION	DÉRIVÉE
$\Phi_{\mathrm{R}}(x) = \frac{1}{2}\big[1 - \mathrm{Erf}\big(\frac{x}{\sqrt{2}}\big)\big]$	$\Phi'_{\mathrm{R}}(x) = -\frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}$
$\widetilde{\Phi}_{\mathrm{R}}(x) = \max\big\{\Phi_{\mathrm{R}}(x), \frac{1}{2} - \frac{x}{\sqrt{2\pi}}\big\}$	$\Phi'_{\mathrm{R}}\big(\max\{0, x\}\big)$
$\Phi_{\mathrm{d}}(x) = 2\,\Phi_{\mathrm{R}}(x)\,\Phi_{\mathrm{R}}(-x)$	$-\sqrt{\frac{2}{\pi}}\mathrm{Erf}\big(\frac{x}{\sqrt{2}}\big) e^{-\frac{1}{2}x^2}$
$\Phi_{\mathrm{e}}(x) = \big[\Phi_{\mathrm{R}}(x)\big]^2$	$2\,\Phi_{\mathrm{R}}(x)\,\Phi'_{\mathrm{R}}(x)$

(b) Les fonctions pertes et leur dérivée.



(c) Données jouets et la frontière de décision pour $\theta = \frac{\pi}{4}$ (la frontière coïncide avec la ligne verticale de la Fig.(d)). Les points sources sont en rouge et vert, les cible en noir.



(d) Valeur des pertes pour PBDA *et* DALC *en fonction de θ.* PBDA *calcule un compromis entre $\widehat{\mathrm{dis}}_{\rho_{\mathbf{w}}}(\mathbb{S},\mathbb{T})$ et $\widehat{R}_{\mathbb{S}}(G_{\rho_{\mathbf{w}}})$ (ou sa relaxation convexe représenté par la ligne rouge en pointillés);* DALC *calcule un compromis entre $\widehat{d}_{\mathbb{T}}(\rho_{\mathbf{w}})$ et $\widehat{e}_{\mathbb{S}}(\rho_{\mathbf{w}})$.*

Figure 4.1. *Illustration du comportement des fonctions pertes de* PBDA *et* DALC. *Les figures du haut (a-b) montrent les fonctions perte. Les figures du bas (c-d) montrent le comportement sur des données jouets.*

Pour des échantillons source $\mathbb{S}{=}\{(\mathbf{x}_i^s, y_i^s)\}_{i=1}^{m_s}$ et cible $\mathbb{T}{=}\{(\mathbf{x}_i^t)\}_{i=1}^{m_t}$ de taille potentiellement différente, et pour des hyperparamètres $B{>}0$ et $C{>}0$, minimiser la fonction objectif suivante par rapport à $\mathbf{w}\in\mathbb{R}$ revient à minimiser la borne du Corollaire 4.5.3 :

$$F_{\mathsf{DALC}}(\mathbf{w}) = C\widehat{\mathrm{d}}_{\mathbb{T}}(\rho_{\mathbf{w}}) + B\,\widehat{\mathrm{e}}_{\mathbb{S}}(\rho_{\mathbf{w}}) + \|\mathbf{w}\|^2 = C\sum_{i=1}^{m_t}\Phi_{\mathrm{d}}\left(\frac{\mathbf{w}\cdot\mathbf{x}_i^t}{\|\mathbf{x}_i^t\|}\right) + B\sum_{i=1}^{m_s}\Phi_{\mathrm{e}}\left(y_i^s\,\frac{\mathbf{w}\cdot\mathbf{x}_i^s}{\|\mathbf{x}_i^s\|}\right) + \|\mathbf{w}\|^2. \quad (4.17)$$

L'algorithme DALC *(Domain Adaptation of Linear Classifier)* correspond à l'optimisation par descente de gradient de l'Équation (4.17), dont le gradient est

$$\nabla F_{\mathsf{DALC}}(\mathbf{w}) = C\sum_{i=1}^{m_t}\Phi'_{\mathrm{d}}\left(\frac{\mathbf{w}\cdot\mathbf{x}_i^t}{\|\mathbf{x}_i^t\|}\right)\frac{\mathbf{x}_i^t}{\|\mathbf{x}_i^t\|} + B\sum_{i=1}^{m_s}\Phi'_{\mathrm{e}}\left(y_i^s\,\frac{\mathbf{w}\cdot\mathbf{x}_i^s}{\|\mathbf{x}_i^s\|}\right)\frac{y_i^s\mathbf{x}_i^s}{\|\mathbf{x}_i^s\|} + \frac{1}{2}\mathbf{w}.$$

Contrairement à PBDA, notre étude empirique a montré qu'il n'est pas nécessaire de rendre convexe les composantes de l'Équation (4.17) : la descente de gradient est simple à réaliser. En effet, $\Phi_{\mathrm{d}}()$ est lisse et sa dérivée est continue, contrairement à la valeur absolue de $\widehat{\mathrm{dis}}_{\rho_{\mathbf{w}}}(\mathbb{S},\mathbb{T})$ de l'Équation (4.14) (voir la Figure 4.1d). Ainsi, le problème d'optimisation de DALC est plus proche de l'analyse théorique associée que PBDA. Dans l'article original, nous avons étendu DALC à des fonctions noyaux.

4.5.4 Illustration sur des données jouets

Pour comparer et illustrer les compromis mis en jeu par PBDA et DALC, nous avons réalisé une expérience sur des données jouets. Pour générer le jeu de données 2D de la Figure 4.1c,

nous avons généré deux échantillons de 200 points chacun : un échantillon source et un cible. Les échantillons sont générés suivants les mêmes distributions : Chaque échantillon contient 100 exemples positifs générés selon une distribution gaussienne de moyenne $(-1,-1)$ et 100 exemples négatifs générés selon une gaussienne de moyenne $(+1,+1)$, les deux distributions ayant une variance unitaire. Nous avons considéré les classifieurs linéaires de la forme $h_{\mathbf{w}}$ avec $\mathbf{w}=2(\cos\theta,\sin\theta)\in\mathbb{R}^2$. La valeur de la norme est fixée à $\|\mathbf{w}\|=2$. La Figure 4.1d montre les quantités qui varient dans les algorithmes tout en faisant tourner la frontière de décision autour de l'origine avec $\theta\in[-\pi,\pi]$. PBDA minimise un compromis entre le désaccord entre domaines $\widehat{\mathrm{dis}}_{\rho_{\mathbf{w}}}(\mathbb{S},\mathbb{T})$ et la relaxation convexe du risque de Gibbs source $\widehat{\mathrm{R}}_{\mathbb{S}}(G_{\rho_{\mathbf{w}}})$. DALC minimise quant à lui un compromis entre le désaccord cible $\widehat{\mathrm{d}}_{\mathbb{T}}(\rho_{\mathbf{w}})$ et l'erreur jointe source $\widehat{\mathrm{e}}_{\mathbb{S}}(\rho_{\mathbf{w}})$. Les Figures 4.1a et 4.1d montrent que le risque de Gibbs, sa relaxation convexe et l'erreur jointe ont un comportement similaire : ils suivent la performance du classifieur linéaire sur l'échantillon source. Par contre, le désaccord entre domaines (*i.e.*, la divergence entre les domaines) et le désaccord cible diffèrent pour l'expérience de la Figure 4.1d : quand la performance cible est optimale ($\theta\approx\frac{\pi}{4}$), le désaccord cible est proche de sa valeur minimale, alors que c'est l'opposé pour le désaccord entre domaines. En supposant que les hyperparamètres, qui contrôlent le compromis $\widehat{\mathrm{d}}_{\mathbb{T}}(\rho_{\mathbf{w}})$ et $\widehat{\mathrm{e}}_{\mathbb{S}}(\rho_{\mathbf{w}})$, soient correctement choisis, la procédure de minimisation de DALC est capable de trouver une solution proche de celle minimisant le risque cible. Au contraire, pour tous les paramètres, PBDA favorise une solution qui minimise le risque source ($\theta\approx 0$), puisqu'il minimise $\widehat{\mathrm{dis}}_{\rho_{\mathbf{w}}}(\mathbb{S},\mathbb{T})$ et $\widehat{\mathrm{R}}_{\mathbb{S}}(G_{\rho_{\mathbf{w}}})$ conjointement.

4.6 Résumé des expériences

Nous avons évalué nos algorithmes PBDA et DALC sur deux jeux de données d'adaptation de domaine : le jeu de données jouet des 2 lunes et *Amazon reviews* (Blitzer et al., 2006). La minimisation de la fonction objectif de PBDA et DALC est effectuée avec la méthode de Broyden-Fletcher-Goldfarb-Shanno (BFGS) implémentée dans la librairie Python *scipy*.

Comportement des algorithmes. La Figure 4.2 illustre le comportement la frontière de décision de PBDA et DALC sur le problème jouet des lunes[9], où chaque lune correspond à une étiquette. Le domaine cible (sans étiquette lors de l'apprentissage) est une rotation du domaine source. La figure montre que PBDA et DALC arrive à adapter le modèle au domaine cible, même pour un angle de rotation de $50°$. Nous constatons que nos algorithmes ne reposent pas sur l'hypothèse du *covariate shift* puisque des points sources sont mal classés. Ce comportement met en évidence que les compromis de PBDA et DALC permettent des erreurs sur l'échantillon source pour réduire le désaccord sur l'échantillon cible.

Résumé des résultats sur *Amazon reviews*. *Amazon reviews* (Blitzer et al., 2006) est composé d'avis sur quatre types de produits (de 1 à 5 étoiles). Nous avons considéré le prétraitement proposé par Chen et al. (2011) où un avis est étiqueté $+1$ quand l'avis est supérieur à 3 étoiles, et -1 sinon. En considérant chaque type de produit comme un domaine, nous avons réalisé 12 tâches d'adaptation de domaine d'avis d'un type de produits vers un autre type. Nous avons comparé PBDA et DALC avec un noyau linéaire à des algorithmes de l'état de l'art au moment de la publication de l'article original, en particulier,

9. Chaque paire de lune est générée avec la fonction `make_moons` de `scikit-learn` (Pedregosa et al., 2011).

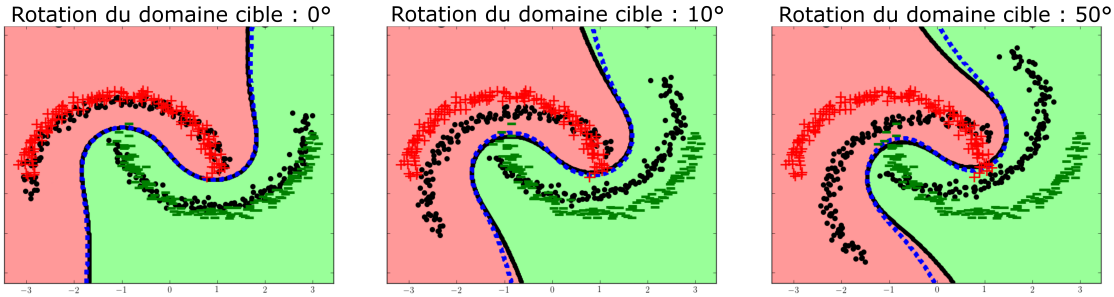


Figure 4.2. *Frontières de décision de* PBDA *(en pointillés bleu) et de* DALC *(en noir) sur le problème jouet des deux lunes pour des paramètres fixés $\alpha = A = 1$ and $B = C = 1$ et un noyau gaussien. Les points cibles sont en noirs. Les points sources positifs, resp. négatifs, sont en rouge, resp. vert.*

nous nous sommes comparés aux algorithmes d'adaptation de domaine DASVM (Bruzzone et Marconcini, 2010) et CODA (Chen et al., 2011). DASVM est un algorithme itératif qui cherche à maximiser itérativement une notion de marge sur des exemples cibles auto-étiquetés. CODA est un algorithme qui cherche itérativement les attributs cibles reliés à l'ensemble d'apprentissage. Pour la sélection des hyperparamètres des méthodes d'adaptation, puisque nous n'avons pas accès aux étiquettes cibles, il est impossible de réaliser une validation croisée classique. Pour contourner ce problème, nous avons effectué une validation inverse (Zhong et al., 2010) détaillée dans Germain et al. (2020, Sec. 7.2.2).

Tout d'abord, nous avons observé que l'algorithme le plus performant sur la tâche considérée en termes de taux d'erreurs est notre algorithme DALC. Un test de rang signé de Wilcoxon, avec un niveau de signification de 5%, a confirmé que DALC est meilleur que PBDA avec une probabilité de 89.5%. Ce résultat tend à confirmer que l'analyse de l'adaptation de domaine associée à DALC (*i.e.*, notre nouvelle vision de l'adaptation) améliore l'analyse basée sur le point de vue plus classique associée à une notion de divergence entre les domaines. Nous pouvons également noter que PBDA est en moyenne meilleur que CODA, mais moins précis que DASVM. Cependant, PBDA reste compétitif : les résultats ne sont pas significativement différents de CODA et DASVM. Il est important de noter que DALC et PBDA sont significativement plus rapides que CODA et DASVM qui reposent sur des procédures itératives coûteuses. En fait, l'avantage de l'approche PAC-Bayésienne est l'optimisation conjointe des termes de nos bornes (*i.e.*, en une seule étape).

4.7 Conclusion

Ce chapitre présente deux analyses de l'adaptation de domaine pour le contexte PAC-Bayésien : la première est basée sur un principe classique en adaptation de domaine, tandis que la seconde apporte une nouvelle perspective de l'adaptation de domaine.

Pour commencer, nous avons suivi la philosophie sous-jacente des travaux fondateurs de Ben-David et al. (2006, 2010a) et de Mansour et al. (2009a). Pour cela, nous avons dérivé une borne supérieure sur le risque de Gibbs cible grâce à une mesure de divergence entre les domaines adaptée au PAC-Bayes. Cette divergence est définie comme l'écart moyen entre le désaccord sur les domaines source et cible. Cela nous a amené à une borne qui prend la forme d'un compromis entre le risque source, la divergence entre les domaines et un terme qui capture la capacité d'adaptation. La borne en généralisation PAC-Bayésienne qui en découle est, en fait, la première borne en généralisation PAC-Bayésienne pour l'adaptation de domaine. Ensuite, nous avons proposé une borne d'adaptation de domaine en tirant parti du comportement inhérent du risque de Gibbs. De là, nous avons démontré une borne

supérieure différente qui exprime un compromis entre le désaccord uniquement sur le domaine cible, l'erreur jointe sur le domaine source et un terme reflétant l'erreur dans les régions où le domaine source est non informatif. À notre connaissance, une originalité de cette contribution est que ce compromis est contrôlé par une divergence entre les domaines : contrairement à notre première borne, la divergence n'est plus un terme additif (comme dans de nombreuses bornes d'adaptation de domaine) mais est un facteur qui pondère l'importance de l'information source.

Nos bornes d'adaptation de domaine, combinées avec des bornes en généralisation PAC-Bayésiennes, conduisent à deux nouveaux algorithmes d'adaptation de domaine pour les classifieurs linéaires : PBDA associé à la philosophie classique et DALC associé à la nouvelle perspective. Au moment de la publication de l'article (GERMAIN et al., 2020), notre étude empirique a montré que les deux algorithmes sont compétitifs avec d'autres approches et que DALC surpasse significativement PBDA.

Revisite PAC-Bayésienne des *Random Fourier Features*

5

Contexte

Ce chapitre présente les travaux de Letarte et al. (2019b, publié à AISTATS) qui ont permis de mettre en place les fondements du projet ANR PRC APRIORI financé en 2018 et dont j'ai assuré la coordination. Ces travaux ont également donné lieu à une contribution développée durant la thèse de Léo Gautheron qui a été publiée à ECML-PKDD (Gautheron et al., 2020). Le point de départ et l'originalité de ces travaux est la revisite d'une méthode d'approximation des fonctions noyaux au travers du prisme de la théorie PAC-Bayésienne.

5.1 Introduction

Les méthodes à noyaux (Shawe-Taylor et Cristianini, 2004), telles que les SVM (Boser et al., 1992 ; Cortes et Vapnik, 1995), projettent les données dans un espace à grande dimension dans lequel un prédicteur linéaire peut résoudre le problème d'apprentissage considéré. L'espace de projection n'est pas directement calculé et le prédicteur linéaire est implicitement représenté via une fonction noyau. C'est l'astuce du noyau *(kernel trick)* : la fonction noyau calcule le produit scalaire entre deux points dans un espace de grande dimension. Cependant, les méthodes à noyaux souffrent de deux inconvénients bien connus. D'une part, le calcul de tous les produits scalaires pour tous les exemples d'apprentissage est coûteux : $O(m^2)$ pour de nombreuses méthodes, où m est la taille de l'ensemble d'apprentissage. D'autre part, il est nécessaire de choisir une fonction noyau qui soit adaptée au problème d'apprentissage pour que l'algorithme fonctionne. Le premier de ces inconvénients a motivé

le développement de méthodes d'approximation pour rendre les méthodes à noyaux plus rapidement calculables. Par exemple, l'approximation de Nyström (Williams et Seeger, 2001 ; Drineas et Mahoney, 2005) construit une approximation de rang faible de la matrice de Gram [1] indépendante des données. Dans ce chapitre, nous proposons une revisite "PAC-Bayésienne" d'une autre technique : les *Random Fourier Features* (RFF, Rahimi et Recht, 2007) qui approximent le noyau à l'aide de caractéristiques aléatoires basées sur la transformation de Fourier indépendante des données (voir Yang et al., 2012, pour une comparaison des deux approches).

Le point de départ de cette revisite est l'observation du fait qu'un prédicteur basé sur les caractéristiques de Fourier du noyau peut être écrit comme une combinaison pondérée de ces caractéristiques selon une distribution indépendante des données définie par la transformée de Fourier. L'originalité de ce chapitre est que cette distribution est interprétée comme une distribution *a priori* sur un espace d'hypothèses faibles, où chaque hypothèse est une simple fonction trigonométrique obtenue par la décomposition de Fourier. Cela suggère que l'on peut améliorer l'approximation en adaptant cette distribution par rapport aux données : notre objectif est d'apprendre une distribution *a posteriori*. Ce faisant, notre étude propose des stratégies pour apprendre une telle représentation des données. Bien que cette représentation ne soit pas aussi flexible et puissante que celles pouvant être apprises par les réseaux de neurones profonds (Goodfellow et al., 2016), nous pensons qu'il est utile d'étudier cette stratégie non seulement pour résoudre le deuxième inconvénient des méthodes à noyaux (qui reposent fortement sur le choix du noyau) mais également permettre l'apprentissage avec peu de données. Dans cet esprit, alors que la majorité des travaux liés aux RFF se concentrent sur l'étude et l'amélioration de l'approximation du noyau, nous proposons également une réinterprétation à la lumière de la théorie PAC-Bayésienne. Nous dérivons des bornes en généralisation qui peuvent être directement optimisées en apprenant un *pseudo-posterior* grâce à une expression en forme close. Nous avons également développé en algorithme basé sur le principe du *gradient boosting* (Friedman, 2001) pour apprendre conjointement une représentation (parcimonieuse) et le prédicteur final.

5.2 Les RFF : *Random Fourier Features*

5.2.1 Cadre général

Nous nous plaçons dans le cadre de la classification supervisée où nous souhaitons apprendre un modèle $f : \mathbb{X} \to \mathbb{Y}$, allant d'un espace d'entrée $\mathbb{X} \subseteq \mathbb{R}^{\mathrm{d}}$ de dimension d à un espace de sortie discret. L'algorithme d'apprentissage prend en entrée un ensemble d'apprentissage $\mathbb{S} = (\mathbf{x}_i, y_i)_{i=1}^{m} \sim \mathcal{D}^m$ composé de m exemples *i.i.d.* selon $\mathcal{D}$, où $\mathcal{D}$ une distribution fixe et inconnue sur $\mathbb{X} \times \mathbb{Y}$. Nous considérons une fonction noyau semi-définie positive (PSD) $k : \mathbb{R}^{\mathrm{d}} \times \mathbb{R}^{\mathrm{d}} \to [-1, 1]$. Les méthodes à noyaux apprennent des modèles de la forme

$$f(\mathbf{x}) \;=\; \sum_{i=1}^{m} \alpha_i k(\mathbf{x}_i, \mathbf{x}) \,, \tag{5.1}$$

en optimisant les valeurs du vecteur $\boldsymbol{\alpha} \in \mathbb{R}^m$.

1. La matrice de Gram est la matrice $m \times m$ constituée de toutes les valeurs du noyau calculées sur l'ensemble d'apprentissage.

5.2.2 Les caractéristiques de Fourier

Lorsque m est grand, exécuter une méthode à noyaux (*e.g.*, SVM ou *kernel ridge regression*) est coûteux en mémoire et en temps. Pour contourner ce problème, RAHIMI et RECHT (2007) ont introduit les *random Fourier features* (caractéristiques de Fourier aléatoires) comme moyen pour approximer la valeur d'un noyau invariant par translation. Plus formellement, en se basant sur la valeur de $\boldsymbol{\kappa}=\mathbf{x}-\mathbf{x}' \in \mathbb{R}^{\mathrm{d}}$, nous écrivons la valeur d'un noyau indifféremment

$$k(\boldsymbol{\kappa}) \;=\; k(\mathbf{x}-\mathbf{x}') \;=\; k(\mathbf{x},\mathbf{x}').$$

Nous notons la transformée de Fourier d'un tel noyau $p(\boldsymbol{\omega})$ définie par

$$p(\boldsymbol{\omega}) \;=\; \frac{1}{(2\pi)^{\mathrm{d}}} \int_{\mathbb{R}^{\mathrm{d}}} k(\boldsymbol{\kappa})\, e^{-i\,\boldsymbol{\omega}\cdot\boldsymbol{\kappa}} d\,\boldsymbol{\kappa}\,. \tag{5.2}$$

En écrivant $k()$ comme l'inverse de la transformée de Fourier et en utilisant des identités trigonométriques, on a :

$$\begin{aligned} k(\mathbf{x}-\mathbf{x}') = \int_{\mathbb{R}^{\mathrm{d}}} p(\boldsymbol{\omega}) e^{i\,\boldsymbol{\omega}\cdot(\mathbf{x}-\mathbf{x}')} \mathrm{d}\,\boldsymbol{\omega} \;&=\; \mathop{\mathbb{E}}_{\boldsymbol{\omega}\sim p} e^{i\,\boldsymbol{\omega}\cdot(\mathbf{x}-\mathbf{x}')} \\ &= \mathop{\mathbb{E}}_{\boldsymbol{\omega}\sim p} \Big[\cos\Big(\boldsymbol{\omega}\cdot(\mathbf{x}-\mathbf{x}')\Big) + i \sin\Big(\boldsymbol{\omega}\cdot(\mathbf{x}-\mathbf{x}')\Big)\Big] \\ &= \mathop{\mathbb{E}}_{\boldsymbol{\omega}\sim p} \cos\Big(\boldsymbol{\omega}\cdot(\mathbf{x}-\mathbf{x}')\Big)\,. \end{aligned} \tag{5.3}$$

RAHIMI et RECHT (2007) ont proposé d'exprimer l'expression ci-dessus $\cos\Big(\boldsymbol{\omega}\cdot(\mathbf{x}-\mathbf{x}')\Big)$ comme un produit de deux caractéristiques. Une façon d'y parvenir est de transformer chaque exemple d'entrée en

$$\mathbf{z}_{\boldsymbol{\omega}}(\mathbf{x}) \;=\; \Big(\cos(\boldsymbol{\omega}\cdot\mathbf{x}),\; \sin(\boldsymbol{\omega}\cdot\mathbf{x})\Big)\,. \tag{5.4}$$

La variable aléatoire $\mathbf{z}_{\boldsymbol{\omega}}(\mathbf{x})\cdot\mathbf{z}_{\boldsymbol{\omega}}(\mathbf{x}')$, avec $\boldsymbol{\omega}$ tiré selon p, est un estimateur non biaisé de $k(\mathbf{x}-\mathbf{x}')$. En effet, à partir des Équations (5.3) et (5.4), on a :

$$\mathop{\mathbb{E}}_{\boldsymbol{\omega}\sim p} \mathbf{z}_{\boldsymbol{\omega}}(\mathbf{x})\cdot\mathbf{z}_{\boldsymbol{\omega}}(\mathbf{x}') = \mathop{\mathbb{E}}_{\boldsymbol{\omega}\sim p} \cos\Big(\boldsymbol{\omega}\cdot(\mathbf{x}-\mathbf{x}')\Big)\,.$$

Pour réduire la variance dans l'estimation de $k(\mathbf{x}-\mathbf{x}')$, une solution consiste à tirer D points $\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, \ldots, \boldsymbol{\omega}_D$ *i.i.d.* selon p. Chaque exemple d'apprentissage $\mathbf{x}_i \in \mathbb{R}^{\mathrm{d}}$ est alors plongé dans un nouveau vecteur de caractéristiques dans $\mathbb{R}^{2D}$:

$$\boldsymbol{\phi}(\mathbf{x}_i) = \frac{1}{\sqrt{D}}\Big(\cos(\boldsymbol{\omega}_1\cdot\mathbf{x}_i)\,,\; \ldots\,,\; \cos(\boldsymbol{\omega}_D\cdot\mathbf{x}_i)\,, \sin(\boldsymbol{\omega}_1\cdot\mathbf{x}_i)\,,\; \ldots\,,\; \sin(\boldsymbol{\omega}_D\cdot\mathbf{x}_i)\Big). \tag{5.5}$$

On a donc $k(\mathbf{x}-\mathbf{x}') \approx \boldsymbol{\phi}(\mathbf{x})\cdot\boldsymbol{\phi}(\mathbf{x}')$ lorsque D est "suffisamment grand". Cela fournit une décomposition du noyau PSD $k()$ qui diffère de la décomposition classique (comme discuté par BACH, 2017). En apprenant un classifieur linéaire sur l'ensemble d'apprentissage transformé $\mathbb{S} \mapsto \{(\boldsymbol{\phi}(\mathbf{x}_i), y_i)\}_{i=1}^m$ via un algorithme tel qu'un SVM linéaire, nous retrouvons un classifieur équivalent à celui appris par une méthode à noyaux. Autrement dit, nous apprenons un vecteur de poids $\mathbf{w} = (w_1, \ldots, w_{2D}) \in \mathbb{R}^{2D}$ et nous prédisons l'étiquette d'un exemple $\mathbf{x} \in \mathbb{X}$ en calculant (à la place de l'Équation (5.1)) :

$$f(\mathbf{x}) \;=\; \sum_{j=1}^{2D} w_j\, \phi_j(\mathbf{x})\,. \tag{5.6}$$

5.3 La transformée de Fourier vue tel un *prior*

Comme décrit précédemment, les RFF ont été introduits pour réduire le temps d'exécution des méthodes à noyaux. En conséquence, à quelques exceptions près, comme les algorithmes d'*apprentissage de noyaux* de YANG et al., 2015, SINHA et DUCHI (2016) et OLIVA et al. (2016), que nous discutons et mettons en relation avec notre approche dans la Section 5.5, la plupart des travaux de la littérature étudient et/ou améliorent les propriétés de l'approximation (*e.g.*, YU et al., 2016 ; BACH, 2017 ; RUDI et ROSASCO, 2017 ; CHOROMANSKI et al., 2018). Notre objectif est de réinterpréter la transformée de Fourier (*i.e.*, la distribution p de l' Équation (5.2)) comme une distribution *a priori* sur l'espace des caractéristiques. Elle peut être vue comme une représentation alternative des connaissances *a priori* qui sont encodées dans le choix d'une fonction noyau spécifique, que nous noterons $k_p()$. En accord avec l'Équation (5.3), chaque caractéristique obtenue à partir d'un vecteur $\boldsymbol{\omega} \in \mathbb{R}^{\mathrm{d}}$ peut être interprétée comme une hypothèse ou un votant

$$h_{\boldsymbol{\omega}}(\boldsymbol{\kappa}) = \cos(\boldsymbol{\omega} \cdot \boldsymbol{\kappa}) \,.$$

De ce point de vue, le noyau est alors un classifieur défini comme un vote de majorité, pondéré par p, d'hypothèses faibles. Cette interprétation de p comme un *prior* sur des hypothèses suggère que l'on peut apprendre un *posterior* sur ces hypothèses. Autrement dit, nous cherchons une distribution q qui génère un nouveau noyau

$$k_q(\boldsymbol{\kappa}) = \mathop{\mathbb{E}}_{\boldsymbol{\omega} \sim q} h_{\boldsymbol{\omega}}(\boldsymbol{\kappa}) \,.$$

Pour évaluer la qualité du noyau $k_q()$, nous considérons une fonction perte basée sur le fait que sa sortie doit être grande lorsque deux exemples ont la même étiquette, et faible dans le cas contraire. Ainsi, nous évaluons le noyau sur deux exemples $(\mathbf{x}, y)$ et $(\mathbf{x}', y')$ avec la fonction perte linéaire définie dans ce contexte par

$$\ell_{lin}\Big(k_q, (\boldsymbol{\kappa}, \lambda)\Big) \;=\; \ell\Big(k_q, (\boldsymbol{\kappa}, \lambda)\Big) \tag{5.7}$$

$$\;=\; \ell\Big(k_q(\boldsymbol{\kappa}), \lambda\Big) \;=\; \frac{1 - \lambda\, k_q(\boldsymbol{\kappa})}{2} \,, \tag{5.8}$$

où $\boldsymbol{\kappa} = \mathbf{x} - \mathbf{x}'$ est la différence entre les deux points $\mathbf{x}$ et $\mathbf{x}'$ et λ mesure la similarité entre les étiquettes

$$\lambda = \lambda(y, y') = \begin{cases} 1 & \text{si } y = y', \\ -1 & \text{sinon.} \end{cases} \tag{5.9}$$

Notons que l'utilisation de la notation $\ell\Big(k_q(\boldsymbol{\kappa}), \lambda\Big)$ dans l'Équation (5.8) au lieu $\ell\Big(k_q, (\boldsymbol{\kappa}, \lambda)\Big)$ de l'Équation (5.7) constitue un abus de notation par rapport au reste du manuscrit. Cette notation est introduite afin de simplifier la lecture de ce chapitre.

Nous définissons maintenant le risque réel d'alignement de noyau *(kernel alignment)* $\mathsf{R}_\Delta(k_q)$ sur une distribution de probabilité Δ définie sur $\mathbb{R}^{\mathrm{d}} \times [-1, 1]$ par

$$\mathsf{R}_\Delta(k_q) \;= \mathop{\mathbb{E}}_{(\boldsymbol{\kappa}, \lambda) \sim \Delta} \ell\Big(k_q(\boldsymbol{\kappa}), \lambda\Big) \,. \tag{5.10}$$

Toute distribution $\mathcal{D}$ sur les espaces d'entrée-sortie $\mathbb{X} \times \mathbb{Y}$ donne lieu à une telle distribution $\Delta_{\mathcal{D}}$. Par abus de notation, $\mathsf{R}_{\mathcal{D}}(k_q)$ désigne le risque réel correspondant et le risque empirique associée à l'alignement de noyau est

$$\widehat{\mathsf{R}}_{\mathbb{S}}(k_q) = \frac{1}{m(m-1)} \sum_{i,j=1, i \neq j}^{m} \ell\Big(k_q(\boldsymbol{\kappa}_{ij}), \lambda_{ij}\Big) \,, \tag{5.11}$$

où pour une paire $\{(\mathbf{x}_i, y_i), (\mathbf{x}_j, y_j)\} \in \mathbb{S}^2$ on a $\boldsymbol{\kappa}_{ij} = (\mathbf{x}_i - \mathbf{x}_j)$ et $\lambda_{ij} = \lambda(y_i, y_j)$.

Avec pour point de départ cette réinterprétation de la transformée de Fourier, nous proposons deux analyses PAC-Bayésiennes. La première, dans la Section 5.4, est obtenue en combinant m bornes PAC-Bayésiennes : au lieu de considérer toutes les paires possibles d'exemples, nous fixons un exemple et nous étudions la capacité en généralisation pour toutes les paires contenant l'exemple en question. La seconde analyse, dans la Section 5.5, est basée sur le fait que le risque de l'alignement de noyau peut s'exprimer comme une U-statistique de second ordre.

5.4 Analyse PAC-Bayésienne et points repères

5.4.1 Une borne du premier ordre

La fonction $\ell()$ est ici linéaire, nous pouvons donc écrire le risque de $k_q()$ comme une espérance pondérée par q des risques des hypothèses. En effet, les Équations (5.10) et (5.11) deviennent

$$\mathrm{R}_{\mathcal{D}}(k_q) \;=\; \mathop{\mathbb{E}}_{(\boldsymbol{\kappa},\lambda)\sim\Delta_{\mathcal{D}}} \ell\Big(\mathop{\mathbb{E}}_{\boldsymbol{\omega}\sim q} h_{\boldsymbol{\omega}}(\boldsymbol{\kappa}), \lambda \Big) \;=\; \mathop{\mathbb{E}}_{\boldsymbol{\omega}\sim q} \mathop{\mathbb{E}}_{(\boldsymbol{\kappa},\lambda)\sim\Delta_{\mathcal{D}}} \ell(h_{\boldsymbol{\omega}}(\boldsymbol{\kappa}), \lambda) \;=\; \mathop{\mathbb{E}}_{\boldsymbol{\omega}\sim q} \mathrm{R}_{\mathcal{D}}(h_{\boldsymbol{\omega}})\,,$$

$$\text{et}\quad \widehat{\mathrm{R}}_{\mathbb{S}}(k_q) \;=\; \frac{1}{n^2-n} \sum_{i,j=1;i\neq j}^{n} \ell\Big(\mathop{\mathbb{E}}_{\boldsymbol{\omega}\sim q} h_{\boldsymbol{\omega}}(\boldsymbol{\kappa}), \lambda_{ij} \Big) = \mathop{\mathbb{E}}_{\boldsymbol{\omega}\sim q} \widehat{\mathrm{R}}_{\mathbb{S}}(h_{\boldsymbol{\omega}})\,.$$

Cette espérance pondérée par q des risques $\mathrm{R}_{\mathcal{D}}(h_{\boldsymbol{\omega}})$ n'est autre que le risque *(stochastique)* de Gibbs (Définition 2.4.5). La différence, par rapport au cadre classique, est que les bornes PAC-Bayésiennes classiques s'appliquent sur les exemples plutôt que sur des distances. En fait, les théorèmes PAC-Bayésiens classiques ne peuvent pas s'appliquer directement pour majorer $\mathrm{R}_{\mathcal{D}}(k_q)$ car le risque empirique devrait être calculé avec des données *i.i.d.* selon $\Delta_{\mathcal{D}}$. Ce qui n'est pas le cas ici puisque le risque empirique $\widehat{\mathrm{R}}_{\mathbb{S}}(k_q)$ fait intervenir des exemples dépendants : il est calculé avec n^2-n paires composées de m exemples tirés selon $\mathcal{D}$.

Une approche "simple" pour pouvoir appliquer les résultats PAC-Bayésiens *classiques* consiste à borner séparément la perte associée à chaque exemple d'apprentissage. C'est-à-dire que, pour chaque $(\mathbf{x}_i, y_i) \in \mathbb{S}$, nous définissons

$$\mathrm{R}^i_{\mathcal{D}}(h_{\boldsymbol{\omega}}) = \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} \ell\Big(h_{\boldsymbol{\omega}}(\mathbf{x}_i - \mathbf{x}), \lambda(y_i, y) \Big)\,, \tag{5.12}$$

$$\text{et}\quad \widehat{\mathrm{R}}^i_{\mathbb{S}}(h_{\boldsymbol{\omega}}) = \frac{1}{m-1} \sum_{j=1,j\neq i}^{m} \ell\Big(h_{\boldsymbol{\omega}}(\mathbf{x}_i - \mathbf{x}_j), \lambda(y_i, y_j) \Big)\,.$$

Ainsi, le théorème suivant apporte des garanties en généralisation sur $\mathrm{R}^i_{\mathcal{D}}(k_q)$ en fonction de son estimation empirique $\widehat{\mathrm{R}}^i_{\mathbb{S}}(k_q)$. Le Théorème 5.4.1, ci-dessous, est obtenu à partir des résultats de Alquier et al. (2016, Th. 4.1 et Lem. 1), mais peut aussi être retrouvé à partir des résultats de Lever et al. (2013).

Théorème 5.4.1 (Borne PAC-Bayes du 1er ordre)**.** Pour tout $\theta > 0$, pour tout $i \in \{1, \ldots, m\}$ et pour toute une distribution *prior* p sur $\mathbb{R}^{\mathrm{d}}$, pour tout $\delta \in]0, 1]$, on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m} \left[\forall q \text{ sur } \mathbb{R}^{\mathrm{d}},\ \mathrm{R}^i_{\mathcal{D}}(k_q) \leq \widehat{\mathrm{R}}^i_{\mathbb{S}}(k_q) + \frac{1}{\theta} \left(\mathrm{KL}(q\|p) + \frac{t^2}{2(m-1)} + \ln \frac{1}{\delta} \right) \right] \geq 1 - \delta.$$

En appliquant la borne de l'union et le fait que $\mathrm{R}_{\mathcal{D}}(k_q) = \mathbb{E}_{(\mathbf{x}_i,y_i)\sim\mathcal{D}}\,\mathrm{R}^i_{\mathcal{D}}(k_q)$, nous obtenons le corollaire suivant.

Corollaire 5.4.1 (Borne PAC-Bayes du 1er ordre pour les RFF)**.** Pour tout $\theta > 0$ et pour toute une distribution *prior* p sur $\mathbb{R}^{\mathrm{d}}$, pour tout $\delta \in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\forall q \text{ sur } \mathbb{R}^{\mathrm{d}},\ \mathrm{R}_{\mathcal{D}}(k_q) \le \widehat{\mathrm{R}}_{\mathbb{S}}(k_q) + \frac{2}{\theta}\left(\mathrm{KL}(q\|p) + \frac{t^2}{2(m-1)} + \ln\frac{m+1}{\delta}\right)\right] \ge 1-\delta.$$

Puisque le Corollaire 5.4.1 est valide pour toute distribution q, nous pouvons calculer la borne pour toute distribution *posterior* apprise. Comme généralement en PAC-Bayes, la borne suggère la minimisation d'un compromis (paramétré par t) entre le risque empirique $\widehat{\mathrm{R}}_{\mathbb{S}}(k_q)$ et la KL-divergence entre le *prior* p et le *posterior* q, c'est-à-dire

$$\widehat{\mathrm{R}}_{\mathbb{S}}(k_q) + \frac{2}{\theta}\,\mathrm{KL}(q\|p)\,.$$

Il est connu en PAC-Bayes (Lem. 2.1 Germain et al., 2009), que pour θ, p et $\mathbb{S}$ fixés, la valeur minimale de la borne est obtenue avec le *posterior* "pseudo-Bayésien" q^* défini par

$$\forall \boldsymbol{\omega} \in \mathbb{R}^{\mathrm{d}},\ q^*(\boldsymbol{\omega}) \;=\; \frac{1}{Z}\,p(\boldsymbol{\omega})\,\exp\left(-\tau\,\widehat{\mathrm{R}}_{\mathbb{S}}(h_{\boldsymbol{\omega}})\right), \tag{5.13}$$

avec $\tau = \frac{1}{2}\theta$ et Z une constante de normalisation [2], la borne du Corollaire 5.4.1 converge vers $\mathrm{R}_{\mathcal{D}}(k_q)$ à un taux $O\left(\sqrt{\frac{\ln m}{m}}\right)$ pour $\theta = \sqrt{m \ln m}$. En raison de la continuité de l'espace des caractéristiques, le *pseudo-posterior* de l'Équation (5.13) est difficile à calculer. Pour l'estimer, il est possible d'utiliser des méthodes de Monte Carlo (*e.g.*, Dalalyan et Tsybakov, 2012) ou des méthodes bayésiennes variationnelles (*e.g.*, Alquier et al., 2016). Dans ce chapitre, nous explorons une méthode plus simple en travaillant à partir d'un espace de probabilité discret.

5.4.2 Apprentissage basé sur des points repère

Au lieu d'apprendre un noyau global pour chaque exemple, nous proposons d'utiliser le fait que le Théorème 5.4.1 borne la fonction noyau pour les distances à un seul exemple. Notre objectif est d'apprendre un ensemble de noyaux (pouvant aussi être interprétés comme des fonctions de similarité) pour un sous-ensemble d'exemples d'apprentissage. Nous appelons ces points, des points repère (ou *landmark*). Le but est alors d'apprendre une nouvelle représentation de l'espace d'entrée, en transformant les points en vecteurs de caractéristiques compacts, à partir desquels nous pouvons apprendre un prédicteur simple.

Concrètement, en plus de l'ensemble d'apprentissage $\mathbb{S}$ de m exemples *i.i.d.* selon $\mathcal{D}$, nous considérons un ensemble de points repère $\mathbb{L} = \{(\mathbf{x}_l, y_l)\}_{l=1}^{m_{\mathbb{L}}}$ de $m_{\mathbb{L}}$ exemples *i.i.d.* selon $\mathcal{D}$, et une distribution de transformée de Fourier *prior* p. Pour chaque point repère $(\mathbf{x}_l, y_l) \in \mathbb{L}$, nous tirons D vecteurs selon p, notés $\Omega^{\mathbb{L}} = \{\boldsymbol{\omega}_n^l\}_{n=1}^{D} \sim p^D$. Nous considérons alors une distribution uniforme P sur l'ensemble discret d'hypothèses $\Omega^{\mathbb{L}}$, telle que $P(\boldsymbol{\omega}_n^l) = \frac{1}{D}$ et $h_n^l(\boldsymbol{\kappa}) = \cos(\boldsymbol{\omega}_n^l \cdot \boldsymbol{\kappa})$. Nous cherchons à apprendre un ensemble de noyaux $\{\widehat{k}_{Q^l}\}_{l=1}^{m_{\mathbb{L}}}$ où

2. Ce compromis est le même que celui impliqué dans d'autres bornes PAC-Bayes, en particulier les bornes à la Catoni (Théorème 2.4.7). Par exemple, comme discuté par Germain et al. (2016b), une similitude existe entre la minimisation d'une telle borne PAC-Bayes et la règle de mise à jour bayésienne.

chaque $\widehat{k}_{Q^l}$ est obtenu pour un point distinct $\mathbf{x}_l \in \mathbb{L}$ avec un paramètre fixé $\beta > 0$, en calculant le *pseudo-posterior* Q^l donné par

$$Q^l_n \;=\; \frac{1}{Z_l}\,\exp\Big(-\beta\sqrt{m}\,\widehat{\mathrm{R}}^l_{\mathbb{S}}(h^l_n)\Big), \tag{5.14}$$

avec $n = \{1, \ldots, D\}$, et Z_l la constante de normalisation. Notons que l'Équation (5.14) donne le minimum du Théorème 5.4.1 avec $\theta = \beta\sqrt{m}$, *i.e.*, $\beta = 1$ correspond au régime où la borne converge. De manière similaire au Corollaire 5.4.1, la borne de l'union et le Théorème 5.4.1 permettent d'obtenir des garanties en généralisation simultanément pour les m_{L} distributions calculées. Ainsi, pour tout $\delta \in]0, 1]$, on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\forall\{Q^l\}_{l=1}^{m_{\mathrm{L}}},\; \mathrm{R}^l_{\mathcal{D}}(\widehat{k}_{Q^l}) \le \widehat{\mathrm{R}}^l_{\mathbb{S}}(\widehat{k}_{Q^l}) + \frac{1}{\theta}\left(\mathrm{KL}(Q^l\|P) + \frac{t^2}{2(m-1)} + \ln\frac{m_{\mathrm{L}}}{\delta}\right)\right] \ge 1-\delta,$$

avec $\mathrm{KL}(Q^l\|P) = \ln D + \sum_{n=1}^{D} Q^l_n \ln Q^l_n$.

Une fois tous les *pseudo-posterior* calculés avec l'Équation (5.14), notre approche basée sur les points repère permet de plonger les exemples $\mathbf{x} \in \mathbb{R}^{\mathrm{d}}$ m_{L} dans l'espace

$$\boldsymbol{\psi}(\mathbf{x}) = \Big(\widehat{k}_{Q^1}(\mathbf{x}_1 - \mathbf{x}), \ldots, \widehat{k}_{Q^{m_{\mathrm{L}}}}(\mathbf{x}_{m_{\mathrm{L}}} - \mathbf{x})\Big), \tag{5.15}$$

et d'apprendre un classifieur linéaire dans cet espace (*i.e.*, avec l'ensemble d'apprentissage transformé). Notons que cette transformation n'est pas une transformation avec un noyau, elle partage des similitudes avec la transformation proposée par Balcan et al. (2008a,b) et Zantedeschi et al. (2018) pour une fonction de similarité plus générale qu'un noyau, mais fixée pour chaque point repère.

5.4.3 Apprentissage basé sur le *gradient boosting*

La procédure de la Section 5.4.2 présente deux limitations : *(i)* les points repères doivent être fixés avant l'apprentissage de la transformation, et *(ii)* le modèle ne peut être optimisé qu'après avoir appris la transformation. Ainsi, la représentation induite par la transformation n'est pas nécessairement compacte, ni pertinente pour la méthode d'apprentissage de modèle considérée. Pour contourner ces problèmes, nous avons également proposé une stratégie pour effectuer ces deux étapes en même temps via un algorithme de *gradient boosting* (Friedman, 2001) pour apprendre conjointement l'ensemble des points repères et le modèle final.

5.4.3.1 Le *gradient boosting* en quelques mots

Le *gradient boosting* (Friedman, 2001) est une méthode ensembliste pour apprendre de manière gloutonne un vote de majorité sur un ensemble de T classifieurs faibles : à chaque itération un classifier est appris. Le vote de majorité final est :

$$\forall \mathbf{x} \in \mathbb{R}^d,\; \mathrm{sign}\left[H^0(\mathbf{x}) + \sum_{t=1}^{T} \alpha^t h_{\mathbf{a}^t}(\mathbf{x})\right],$$

où H^0 est le classifieur initial fixé avant le processus itératif (souvent défini pour renvoyer la même valeur pour chaque exemple) et α^t est le poids associé au classifieur $h_{\mathbf{a}^t}$, appris en même temps que les paramètres $\mathbf{a}^t$ de ce classifieur. Étant donné une fonction perte $\ell()$ différentiable, l'objectif du *gradient boosting* est de réaliser une descente de gradient

Algorithme 5.1 *Gradient boosting* (FRIEDMAN, 2001)

Entrées : $\mathbb{S} = \{(\mathbf{x}_i, y_i)\}_{i=1}^m$ l'ensemble d'apprentissage, fonction perte $\ell()$, nombre d'itérations T

1: $\forall i \in \{1, \ldots, m\}, \quad H^0(\mathbf{x}_i) = \operatorname{argmin}_c \sum_{i=1}^m \ell(c, y_i)$

2: **pour** $t \in \{1, \ldots, T\}$ **faire**

3: $\quad \forall i \in \{1, \ldots, m\}, \quad \tilde{y}_i = -\dfrac{\partial \ell\Big(y_i, H^{t-1}(\mathbf{x}_i)\Big)}{\partial H^{t-1}(\mathbf{x}_i)}$

4: $\quad \mathbf{a}^t = \operatorname{argmin}_{\mathbf{a}} \sum_{i=1}^m \Big(\tilde{y}_i - h_{\mathbf{a}}(\mathbf{x}_i)\Big)^2$

5: $\quad \alpha^t = \operatorname{argmin}_{\alpha} \sum_{i=1}^m \ell\Big(H^{t-1}(\mathbf{x}_i) + \alpha h_{\mathbf{a}^t}(\mathbf{x}_i), y_i\Big)$

6: $\quad \forall i \in \{1, \ldots, m\}, \quad h^t(\mathbf{x}_i) = H^{t-1}(\mathbf{x}_i) + \alpha^t h_{\mathbf{a}^t}(\mathbf{x}_i)$

retourner $\forall \mathbf{x}, \; H^T(\mathbf{x}) = \operatorname{sign}\left[H^0(\mathbf{x}) + \sum_{t=1}^T \alpha^t h_{\mathbf{a}^t}(\mathbf{x})\right]$

où la variable à optimiser est l'ensemble des classifieurs, et la fonction à minimiser est la perte empirique. L'algorithme du *gradient boosting* est résumé dans l'Algorithme 5.1. Tout d'abord, l'ensemble est constitué d'un seul classifieur : celui qui renvoie une valeur constante minimisant la perte sur l'ensemble d'apprentissage (ligne 1). Ensuite, à chaque itération, l'algorithme calcule pour chaque exemple d'apprentissage le gradient négatif de la perte (ligne 3), également appelé le résidu et noté $\tilde{y}_i$. L'étape suivante consiste à optimiser les paramètres du classifieur $h_{\mathbf{a}^t}$ pour qu'il s'ajuste au mieux aux résidus (ligne 4), avant d'apprendre la taille de pas optimale α^t qui minimise la perte en ajoutant $h_{\mathbf{a}^t}$, pondéré par α^t, au vote actuel (ligne 5). Enfin, le modèle est mis à jour en ajoutant $\alpha^t h_{\mathbf{a}^t}$ au vote (ligne 6).

5.4.4 *Gradient boosting* avec RFF

Nous proposons maintenant un algorithme d'apprentissage, appelé GBRFF1 et résumé dans l'Algorithme 5.2, qui optimise conjointement une représentation compacte des données et le modèle. GBRFF1 tire parti à la fois de l'algorithme de *gradient boosting* rappelé ci-dessus et des RFF. La fonction perte $\ell()$ utilisée par GBRFF1 est la perte exponentielle, plus adaptée à la classification binaire et définie pour tout exemple $(\mathbf{x}, y)$ par $\ell(h(\mathbf{x}), y) = \exp(-y\, h(\mathbf{x}))$. La ligne 1 de l'Algorithme 5.1 définie le classifieur initial par :

$$\forall i \in \{1, \ldots, m\}, \quad H^0(\mathbf{x}_i) = \frac{1}{2} \ln \frac{1 + \frac{1}{m}\sum_{j=1}^m y_j}{1 - \frac{1}{m}\sum_{j=1}^m y_j}. \tag{5.16}$$

Les résidus (ligne 3) sont définis par

$$\tilde{y}_i = -\frac{\partial \ell\left(y_i, H^{t-1}(\mathbf{x}_i)\right)}{\partial H^{t-1}(\mathbf{x}_i)} = y_i \, \exp\Big(-y_i H^{t-1}(\mathbf{x}_i)\Big).$$

À chaque itération t, pour tirer avantage de la décroissance exponentielle de la perte exponentielle, la ligne 4 cherche à apprendre un classifieur faible qui produit des prédictions avec une grande valeur absolue et avec le même signe que les résidus. Ainsi, nous favorisons les valeurs de paramètres minimisant la perte exponentielle entre les résidus et les prédictions des classifieurs faibles comme suit :

$$\mathbf{a}^t = \operatorname*{argmin}_{\mathbf{a}} \frac{1}{m} \sum_{i=1}^m \exp\Big(-\tilde{y}_i \, h_{\mathbf{a}}(\mathbf{x}_i)\Big). \tag{5.17}$$

En suivant le principe des RFF, nous définissons le classifieur faible par

$$h_{\mathbf{a}^t}(\mathbf{x}_i) = \sum_{j=1}^{D} q_j^t \cos\left(\boldsymbol{\omega}_j^t \cdot (\mathbf{x}_t - \mathbf{x}_i)\right), \quad \text{où } \mathbf{a}^t{=}(\{\boldsymbol{\omega}_j^t\}_{j=1}^D, \mathbf{x}_t, Q^t). \tag{5.18}$$

Au lieu d'utiliser un ensemble de points repères préfixé comme dans la Section 5.4.2, nous construisons cet ensemble itérativement $\mathbb{L} = \{\mathbf{x}_t\}_{t=1}^T$ en apprenant un point repère par itération t. Pour tirer parti de la solution en forme close de l'Équation (5.14), nous proposons la méthode gloutonne suivante pour apprendre les paramètres $\mathbf{a}^t$. À chaque itération t, nous tirons D vecteurs $\{\boldsymbol{\omega}_j^t\}_{j=1}^D \sim p^D$ où p est la transformée de Fourier de l'Équation (5.2). Nous cherchons ensuite le point repère optimal $\mathbf{x}_t$. En combinant les Équations (5.18) et (5.17) et en supposant que la distribution *prior* sur les caractéristiques aléatoires est uniforme, $\mathbf{x}_t$ est appris en minimisant

$$\mathbf{x}_t = \underset{\mathbf{x}\in\mathbb{R}^d}{\operatorname{argmin}} \underbrace{\frac{1}{m}\sum_{i=1}^{m} \exp\left(-\tilde{y}_i \frac{1}{D}\sum_{j=1}^{D} \cos(\boldsymbol{\omega}_j^t \cdot (\mathbf{x}{-}\mathbf{x}_i))\right)}_{=\ F(\mathbf{x})}. \tag{5.19}$$

Bien que ce problème ne soit pas convexe en raison de la fonction cosinus, nous pouvons calculer sa dérivée et effectuer une descente de gradient pour trouver une solution possible. La dérivée partielle de l'Équation (5.19) par rapport à $\mathbf{x}$ est

$$\frac{\partial F}{\partial \mathbf{x}}(\mathbf{x}) = \frac{1}{Dm}\sum_{i=1}^{m}\left[\frac{\tilde{y}_i}{D}\sum_{j=1}^{D}\sin(\boldsymbol{\omega}_j^t \cdot (\mathbf{x}{-}\mathbf{x}_i))\right]\exp\left[-\frac{\tilde{y}_i}{D}\sum_{j=1}^{D}\cos(\boldsymbol{\omega}_j^t \cdot (\mathbf{x}{-}\mathbf{x}_i))\right]\sum_{j=1}^{D}\boldsymbol{\omega}_j^t.$$

Comme précédemment, étant donné un point repère $\mathbf{x}_t$ trouvé par descente de gradient, nous calculons les poids des caractéristiques aléatoires Q^t comme suit

$$\forall j \in \{1,\ldots,D\},\ Q_j^t = \frac{1}{Z^t}\exp\left[\frac{-\beta\sqrt{m}}{m}\sum_{i=1}^{m}\exp\left(-\tilde{y}_i\cos\left[\boldsymbol{\omega}_j^t \cdot (\mathbf{x}_t - \mathbf{x}_i)\right]\right)\right], \tag{5.20}$$

où $\beta > 0$ et Z^t est la constante de normalisation. La dernière étape concerne le pas de gradient α^t, qui est calculé pour minimiser la combinaison du modèle courant H^{t-1} avec le classifieur faible h^t, c'est-à-dire

$$\alpha^t \ = \ \underset{\alpha}{\operatorname{argmin}}\sum_{i=1}^{m}\exp\left[-y_i\left(H^{t-1}(\mathbf{x}_i) + \alpha\, h^t(\mathbf{x}_i)\right)\right] \ = \ \underset{\alpha}{\operatorname{argmin}}\sum_{i=1}^{m} w_i \exp\left[-y_i\,\alpha\, h^t(\mathbf{x}_i)\right],$$

où $w_i{=}\exp\left(-y_i H^{t-1}(\mathbf{x}_i)\right)$. Pour obtenir une solution en forme close pour α, nous utilisons la convexité de la quantité ci-dessus et le fait que $h^t(\mathbf{x}_i) \in [-1, 1]$ pour borner la fonction perte à optimiser. En effet, on a

$$\sum_{i=1}^{m} w_i \exp\left[-y_i\alpha h^t(\mathbf{x}_i)\right] \ \leq \ \sum_{i=1}^{m}\left[\frac{1{-}y_i h^t(\mathbf{x}_i)}{2}\right] w_i \exp(\alpha){+}\sum_{i=1}^{m}\left[\frac{1{+}y_i h^t(\mathbf{x}_i)}{2}\right] w_i \exp(-\alpha).$$

Cette borne supérieure est strictement convexe. Son minimum α^t peut être trouvé en annulant la dérivée par rapport à α du côté droit de l'équation précédente. On a

$$\sum_{i=1}^{m}\left(\frac{1 - y_i h^t(\mathbf{x}_i)}{2}\right) w_i\ \exp(\alpha) = \sum_{i=1}^{m}\left(\frac{1 + y_i h^t(\mathbf{x}_i)}{2}\right) w_i\ \exp(-\alpha),$$

Algorithme 5.2 GBRFF1 : *Gradient Boosting* avec RFF

Entrées : Ensemble $\mathbb{S} = \{(\mathbf{x}_i, y_i)\}_{i=1}^m$, nombre d'itérations T, nombre de caractéristiques aléatoires D, paramètres γ, β

$\forall i \in \{1,\dots,m\}, \quad H^0(\mathbf{x}_i) = \frac{1}{2}\ln\frac{1+\frac{1}{m}\sum_{j=1}^m y_j}{1-\frac{1}{m}\sum_{j=1}^m y_j}$

pour $t \in \{1,\dots,T\}$ **faire**

$\forall i \in \{1,\dots,m\}, \quad w_i = \exp(-y_i H^{t-1}(\mathbf{x}_i))$

$\forall i \in \{1,\dots,m\}, \quad \tilde{y}_i = y_i w_i$

$\forall j \in \{1,\dots,D\}, \quad$ tirage de $\boldsymbol{\omega}_j^t \sim \mathcal{N}(0, 2\gamma)^d$

$$\mathbf{x}_t = \underset{\mathbf{x}\in\mathbb{R}^d}{\operatorname{argmin}} \frac{1}{m}\sum_{i=1}^m \exp\left(-\tilde{y}_i \frac{1}{D}\sum_{j=1}^D \cos(\boldsymbol{\omega}_j^t \cdot (\mathbf{x} - \mathbf{x}_i))\right)$$

$$\forall j \in \{1,\dots,D\},\ Q_j^t = \frac{1}{Z^t}\exp\left[\frac{-\beta\sqrt{m}}{m}\sum_{i=1}^m \exp\left(-\tilde{y}_i \cos\left(\boldsymbol{\omega}_j^t \cdot (\mathbf{x}_t - \mathbf{x}_i)\right)\right)\right]$$

$$\alpha^t = \frac{1}{2}\ln\frac{\sum_{i=1}^m \left(1 + y_i \sum_{j=1}^D Q_j^t \cos\left(\boldsymbol{\omega}_j^t \cdot (\mathbf{x}_t - \mathbf{x}_i)\right)\right) w_i}{\sum_{i=1}^m \left(1 - y_i \sum_{j=1}^D Q_j^t \cos\left(\boldsymbol{\omega}_j^t \cdot (\mathbf{x}_t - \mathbf{x}_i)\right)\right) w_i}$$

$$\forall i \in \{1,\dots,m\},\ h^t(\mathbf{x}_i) = H^{t-1}(\mathbf{x}_i) + \alpha^t \sum_{j=1}^D Q_j^t \cos\left(\boldsymbol{\omega}_j^t \cdot (\mathbf{x}_t - \mathbf{x}_i)\right)$$

retourner $\forall \mathbf{x},\ H^t(\mathbf{x}) = \operatorname{sign}\left[H^0(\mathbf{x}) + \sum_{t=1}^T \alpha^t \sum_{j=1}^D Q_j^t \cos\left(\boldsymbol{\omega}_j^t \cdot (\mathbf{x}_t - \mathbf{x})\right)\right]$

dont la solution est $\alpha^t = \frac{1}{2}\ln\left(\frac{\sum_{i=1}^m (1 - y_i h^t(\mathbf{x}_i)) w_i}{\sum_{i=1}^m (1 + y_i h^t(\mathbf{x}_i)) w_i}\right)$.

Cette démarche peut être utilisée pour trouver H^0. Comme souvent pour les RFF (Rahimi et Recht, 2007 ; Sinha et Duchi, 2016 ; Agrawal et al., 2019), nous utilisons un noyau RBF défini par $k(\mathbf{x}, \mathbf{x}') = \exp(-\gamma\|\mathbf{x} - \mathbf{x}'\|^2)$ où les vecteurs de la transformée de Fourier sont composés de d éléments, chacun tiré d'une loi normale $\mathcal{N}(0, 2\gamma)^{\mathrm{d}}$.

5.4.5 Raffinement de GBRFF1

Dans GBRFF1, le nombre de caractéristiques aléatoires D à chaque itération a un impact direct sur le temps de calcul. De plus, $\boldsymbol{\omega}^t$ est tiré selon la transformée de Fourier du noyau RBF et n'est donc pas appris. Nous proposons dans cette section deux améliorations qui donnent lieu à une variante de GBRFF1, appelée GBRFF2 et résumée dans l'Algorithme 5.3. Tout d'abord, nous mettons en évidence le fait qu'il est possible de réduire radicalement la complexité de GBRFF1 en apprenant une approximation grossière du noyau, mais beaucoup plus simple et toujours efficace, avec $D{=}1$. Dans ce scénario, nous montrons que l'apprentissage des points repères revient à trouver un seul nombre réel dans l'intervalle $[-\boldsymbol{\pi}, \boldsymbol{\pi}]$. Ensuite, pour accélérer la convergence de l'algorithme, nous proposons d'optimiser $\boldsymbol{\omega}^t$ après une initialisation aléatoire issue de la transformée de Fourier. Nous montrons qu'une simple descente de gradient par rapport à ce paramètre permet une convergence plus rapide avec de meilleures performances.

Algorithme 5.3 GBRFF2 : *Gradient Boosting* avec RFF

Entrées : Ensemble $\mathbb{S}=\{(\mathbf{x}_i, y_i)\}_{i=1}^m$, nombre d'itérations T, paramètres γ, λ

1: $\forall i \in \{1,\dots,m\}, \quad H^0(\mathbf{x}_i) = \dfrac{1}{2}\ln\dfrac{1+\frac{1}{m}\sum_{j=1}^m y_j}{1-\frac{1}{m}\sum_{j=1}^m y_j}$

2: **pour** $t \in \{1,\dots,T\}$ **faire**

3: $\quad \forall i \in \{1,\dots,m\}, \quad w_i = \exp\left(-y_i\, H^{t-1}(\mathbf{x}_i)\right)$

4: $\quad \forall i \in \{1,\dots,m\}, \quad \tilde{y}_i = y_i w_i$

5: $\quad$ Tirage de $\boldsymbol{\omega} \sim \mathcal{N}(0, 2\gamma)^d$

6: $\quad b_t = \underset{b\in[-\boldsymbol{\pi},\boldsymbol{\pi}]}{\operatorname{argmin}} \dfrac{1}{m}\sum_{i=1}^m \exp\left[-\tilde{y}_i \cos\left(\boldsymbol{\omega}\cdot\mathbf{x}_i - b\right)\right]$

7: $\quad \boldsymbol{\omega}^t = \underset{\boldsymbol{\omega}\in\mathbb{R}^d}{\operatorname{argmin}}\ \lambda\|\boldsymbol{\omega}\|_2^2 + \dfrac{1}{m}\sum_{i=1}^m \exp\left[-\tilde{y}_i\, \cos\left(\boldsymbol{\omega}\cdot\mathbf{x}_i - b_t\right)\right]$

8: $\quad \alpha^t = \dfrac{1}{2}\ln\dfrac{\sum_{i=1}^m \left(1+y_i\cos\left(\boldsymbol{\omega}^t\cdot\mathbf{x}_i - b_t\right)\right) w_i}{\sum_{i=1}^m \left(1-y_i\cos\left(\boldsymbol{\omega}^t\cdot\mathbf{x}_i - b_t\right)\right) w_i}$

9: $\quad \forall i \in \{1,\dots,m\},\ H^t(\mathbf{x}_i) = H^{t-1}(\mathbf{x}_i) + \alpha^t \cos\left(\boldsymbol{\omega}^t\cdot\mathbf{x}_i - b_t\right)$

retourner $\forall \mathbf{x},\ H^T(\mathbf{x}) = \operatorname{sign}\left[H^0(\mathbf{x}) + \sum_{t=1}^T \alpha^t \cos\left(\boldsymbol{\omega}^t\cdot\mathbf{x} - b_t\right)\right]$

Apprentissage des points repères à moindre coût grâce à la périodicité de $\cos()$. En fixant $D=1$, la classifieur faible $h_{\mathbf{a}^t}(\mathbf{x})$ devient simplement

$$h_{\mathbf{a}^t}(\mathbf{x}) = \cos\left(\boldsymbol{\omega}^t\cdot(\mathbf{x}_t - \mathbf{x})\right), \quad \text{avec } \mathbf{a}^t = (\boldsymbol{\omega}^t, \mathbf{x}_t).$$

Cette formulation permet de s'affranchir de la dépendance au paramètre D. De plus, il est possible de se passer de β car l'apprentissage des poids Q^t (ligne 7, Algo. 5.2) n'est plus nécessaire. En effet, puisque $D{=}1$, à chaque itération, α^t peut être vue comme un substitut de ces poids. Comme le classifieur faible repose maintenant sur une seule caractéristique aléatoire, la fonction objectif (ligne 6, Algo. 5.2) pour apprendre le point repère à l'itération t devient

$$\mathbf{x}_t = \underset{\mathbf{x}\in\mathbb{R}^d}{\operatorname{argmin}} \underbrace{\frac{1}{m}\sum_{i=1}^m \exp\left[-\tilde{y}_i \cos(\boldsymbol{\omega}^t\cdot(\mathbf{x}-\mathbf{x}_i))\right]}_{=\ F_{\boldsymbol{\omega}^t}(\mathbf{x})}.$$

Soit $c \in \{1,\dots,\mathrm{d}\}$ l'index de la c-ième coordonnée du point repère $\mathbf{x}_t$. Nous réécrivons la fonction objectif comme

$$\begin{aligned} F_{\boldsymbol{\omega}^t}(\mathbf{x}_t) &= \frac{1}{m}\sum_{i=1}^m \exp\left[-\tilde{y}_i \cos\left(\boldsymbol{\omega}^t\cdot\mathbf{x}_t - \boldsymbol{\omega}^t\cdot\mathbf{x}_i\right)\right] \\ &= \frac{1}{m}\sum_{i=1}^m \exp\left[-\tilde{y}_i \cos\left(\omega_c^t\, x_{tc} + \sum_{j\neq c}\omega_j^t\, x_{tj} - \boldsymbol{\omega}^t\cdot\mathbf{x}_i\right)\right]. \end{aligned}$$

Nous exploitons la périodicité de la fonction cosinus dans chaque direction pour trouver la coordonnée optimale $x_{tc} \in \left[\frac{-\boldsymbol{\pi}}{\omega_c^t}, \frac{\boldsymbol{\pi}}{\omega_c^t}\right]$ du point repère qui minimise $F_{\boldsymbol{\omega}^t}(\mathbf{x}_t)$ en fixant toutes les autres coordonnées. La Figure 5.1 illustre ce phénomène lors de l'application de GBRFF1 avec $D{=}1$ sur un ensemble de donnée jouet. Les graphiques de la première rangée montrent la périodicité de la fonction perte représentée par des bandes vertes/jaunes diagonales répétées (le jaune clair étant associé à la plus petite perte). Il existe un nombre infini de points repères

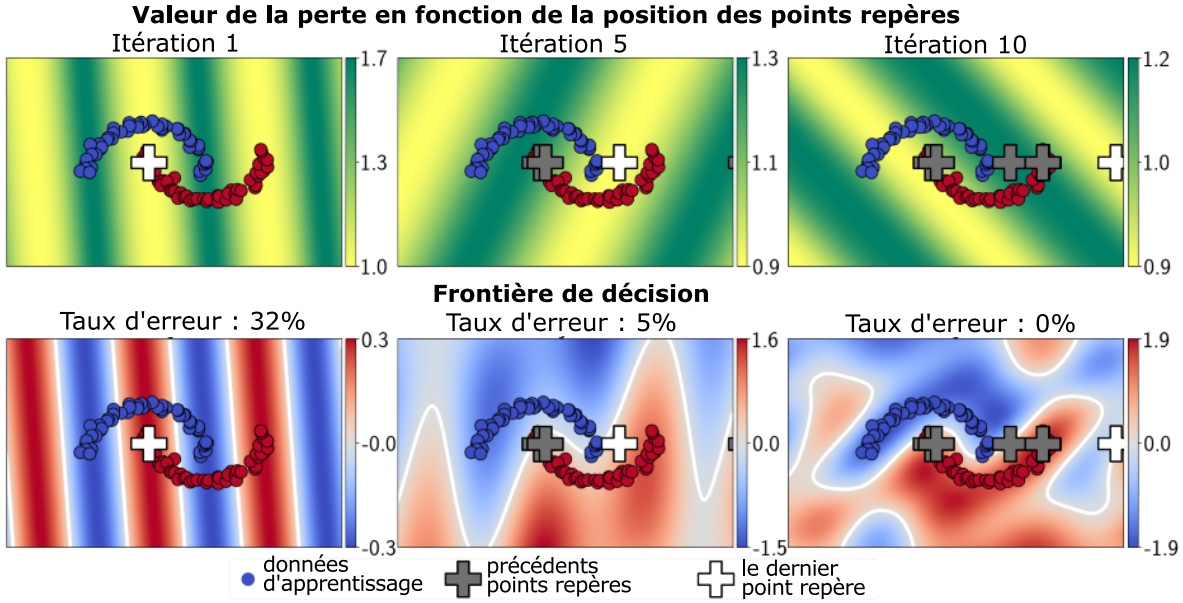


Figure 5.1. ***GBRFF1*** *avec $D=1$ sur le jeu de données des deux lunes à différentes itérations. La rangée du haut montre la périodicité de la perte (le jaune clair indique la perte minimale). La rangée du bas montre les frontières de décision résultantes entre les classes (bleu et rouge), obtenues en fixant arbitrairement une coordonnée du point repère et en minimisant la perte par rapport à l'autre coordonnée.*

produisant une perte minimale au centre des bandes jaunes. Ainsi, en fixant une coordonnée du point repère à une valeur arbitraire, l'algorithme est toujours capable à chaque itération de trouver une valeur le long de la deuxième coordonnée qui minimise la perte (le point repère associé l'itération en cours est représenté par une croix blanche). La deuxième rangée montre qu'une telle stratégie permet d'obtenir une précision de 100% sur cet ensemble de données jouet après 10 itérations. En généralisant cela, au lieu d'apprendre un point repère $\mathbf{x}_t \in \mathbb{R}^d$, nous fixons toutes les coordonnées du point à 0 sauf une, puis nous apprenons un seul scalaire $b_t \in [-\boldsymbol{\pi}, \boldsymbol{\pi}]$ qui minimise

$$F_{\boldsymbol{\omega}^t}(b_t) = \frac{1}{m}\sum_{i=1}^{m} \exp\Big(-\tilde{y}_i \cos(\boldsymbol{\omega}^t \cdot \mathbf{x}_i - b_t)\Big].$$

Apprentissage de $\boldsymbol{\omega}^t$ pour une convergence plus rapide. La seconde amélioration concerne l'aléa lié aux RFF due au vecteur $\boldsymbol{\omega}^t$ tiré selon p puis utilisé pour apprendre b_t. Nous proposons d'affiner $\boldsymbol{\omega}^t$ en effectuant une descente de gradient avec pour initialisation le vecteur tiré de p. Appuyé par les expériences réalisées dans l'article original (Gautheron et al., 2020), nous pensons qu'une telle stratégie permet à la fois d'accélérer la convergence de l'algorithme et d'améliorer la précision. Cette mise à jour nécessite l'ajout d'une ligne, juste après la ligne 6 de l'Algorithme 5.2, exprimée comme un problème d'optimisation régularisé :

$$\boldsymbol{\omega}^t = \underset{\boldsymbol{\omega}\in\mathbb{R}^d}{\operatorname{argmin}} \underbrace{\lambda\|\boldsymbol{\omega}\|_2^2 + \frac{1}{m}\sum_{i=1}^{m} \exp\left[-\tilde{y}_i \cos(\boldsymbol{\omega}\cdot\mathbf{x}_i - b_t)\right]}_{= F_\omega(\boldsymbol{\omega})},$$

dont la dérivée est

$$\frac{\partial F_\omega}{\partial \boldsymbol{\omega}}(\boldsymbol{\omega}) = 2\lambda\boldsymbol{\omega} + \frac{1}{m}\sum_{i=1}^{m} \mathbf{x}_i\tilde{y}_i \sin(\boldsymbol{\omega}\cdot\mathbf{x}_i - b_t)\, \exp\left[-\tilde{y}_i\cos(\boldsymbol{\omega}\cdot\mathbf{x}_i - b_t)\right].$$

5.4.6 Résumé des expériences

Les objectifs des expériences que nous avons menées dans les articles (LETARTE et al., 2019b ; GAUTHERON et al., 2020) sont : *(i)* justifier l'intérêt de l'approche PAC-Bayésienne pour affiner l'espace de représentation, *(ii)* justifier l'intérêt d'apprendre les points repères plutôt que de les fixer, *(iii)* étudier l'impact du nombre de caractéristiques aléatoires D, *(iv)* étudier les performances de GBRFF1 et GBRFF2. Nous avons utilisé un noyau gaussien sur des données jouets et réelles (du répertoire UCI). Nous appelons PBRFF l'approche en deux étapes suggérée par la Section 5.4.2. Nous avons comparé PBRFF, GBRFF1 et GBRFF2 à différents algorithmes de la littérature : LGBM (KE et al., 2017) qui est un algorithme de *gradient boosting* utilisant des arbres comme prédicteurs, BMKR (WU et al., 2017) qui est une méthode d'apprentissage par noyaux multiples basée sur le *gradient boosting*, GFC (OGLIC et GÄRTNER, 2016) qui est une méthode gloutonne de construction de caractéristiques basée sur la descente de gradient fonctionnelle.

Intérêt de l'approche PAC-Bayésienne. Dans un premier temps, nous avons étudié le comportement de l'approche en deux étapes PBRFF en fonction du nombre de points repères préfixés (sélectionnés via une méthode de *clustering* pour couvrir tout l'espace d'origine). Dans ce cas, dans la transformation obtenue, nous avons appris un classifieur SVM linéaire que nous avons comparé un classifieur SVM appris avec un noyau RBF dans l'espace d'origine. Nous avons observé que l'apprentissage d'un *posterior* amène à de meilleurs résultats que la considération du *prior*, parfois même meilleur que le SVM RBF appris dans l'espace d'origine (qui lui est moins compact et plus coûteux à apprendre). Ce comportement confirme de l'intérêt de l'apport du PAC-Bayes pour construire une meilleure représentation.

Intérêt de l'apprentissage des points repères dans GBRFF1. Rappelons que GBRFF1 apprend conjointement la représentation et les points repères. Nous avons comparé GBRFF1 à une version (notée GBRFF0.5) où les points repères sont choisis aléatoirement au lieu d'être appris. Tout d'abord, nous avons observé que GBRFF0.5 est plus rapide que PBRFF mais amène à des performances plus faibles, probablement dû au fait que le classifieur appris par *boosting* est moins efficace qu'un SVM dans ce cadre. Par contre, GBRFF1 améliore les résultats à la fois de GBRFF0.5 et de PBRFF (mais est moins rapide). Le résultat sans doute le plus frappant provient des performances de notre variante GBRFF2, qui surpasse PBRFF et GBRFF1 ; ces derniers ayant besoin de plus d'itérations sans parvenir à atteindre le même niveau de performance. Cela démontre clairement les avantages d'apprendre les caractéristiques aléatoires plutôt que de les générer de manière aléatoire.

Influence du nombre de caractéristiques aléatoires D. Un élément clé, à la fois pour PBRFF et GBRFF1 est le nombre de caractéristiques aléatoires D. Comme prévu, la performance s'améliore avec un plus grand nombre de caractéristiques aléatoires par point repère, mais au prix d'un temps de calcul de plus en plus élevé. Nous avons également observé que plus le nombre de caractéristiques aléatoires est élevé, plus le gain en performance devient faible, tandis que l'augmentation du temps de calcul devient plus importante. Plus particulièrement, GBRFF1 avec $D=1$ offre le meilleur compromis entre précision et temps de calcul. Cela signifie que, même si pour un nombre fixé T de points repères, de meilleures performances peuvent être obtenues avec une grande valeur de D, il est plus avantageux de fixer $D=1$ et d'utiliser un grand nombre de points repères T pour obtenir des performances similaires en moins de temps. Pour comprendre pourquoi il est préférable d'utiliser un faible

nombre D de caractéristiques aléatoires par point repère, mais un grand nombre de points repère T, nous rappelons la formule du prédicteur final de GBRFF1 pour un exemple donné $\mathbf{x}$:

$$\operatorname{sign}\left(H^0(\mathbf{x}) + \sum_{t=1}^{T} \alpha^t \sum_{j=1}^{K} q_j^t \cos\left(\boldsymbol{\omega}_j^t \cdot (\mathbf{x}_t - \mathbf{x})\right)\right),$$

qui se simplifie quand $D=1$ en

$$\operatorname{sign}\left(H^0(\mathbf{x}) + \sum_{t=1}^{T} \alpha^t \cos\left(\boldsymbol{\omega}^t \cdot (\mathbf{x}_t - \mathbf{x})\right)\right).$$

À une itération donnée t, l'objectif est d'apprendre le point repère $\mathbf{x}_t$, le poids de *boosting* α^t et les poids des caractéristiques aléatoires Q^t, de manière à bien ajuster les résidus définis par la perte exponentielle. Par conséquent, lorsque D augmente, le nombre de contraintes imposées à une itération donnée pour apprendre $\mathbf{x}_t$ et α^t augmente également. Une explication possible est que, lorsque $D=1$, il est plus simple de trouver un point repère $\mathbf{x}_t$ et un poids α^t qui ajustent correctement les résidus, car ils sont moins contraints. En revanche, avec un grand nombre de caractéristiques aléatoires, il devient impossible de trouver une solution qui ajuste correctement les résidus en respectant les contraintes imposées par ces caractéristiques.

Comparaison des performances de GBRFF1 et GBRFF2. L'algorithme GBRFF2 se distingue de GBRFF1 par *(i)* l'utilisation d'une seule caractéristique aléatoire par point repère et *(ii)* l'apprentissage de la partie aléatoire de la caractéristique aléatoire $\boldsymbol{\omega}$ (au lieu de la fixer aléatoirement). Pour les expériences, nous avons introduit une variante appelée GBRFF1.5, identique à GBRFF2, sauf que $\boldsymbol{\omega}$ n'est pas appris, mais est fixé aléatoirement. Cette variante diffère de GBRFF1 car l'utilisation d'une seule caractéristique aléatoire permet d'apprendre un scalaire unique au lieu d'un vecteur, tout en obtenant le même modèle que GBRFF1 avec $D=1$, mais de manière plus efficace. La comparaison entre GBRFF1 avec $D=1$ et GBRFF1.5 a montré, comme attendu, que les deux méthodes mènent exactement aux mêmes performances, mais avec un temps de calcul bien moindre pour GBRFF1.5. Cela confirme que lorsqu'on utilise une seule caractéristique aléatoire, il est équivalent d'apprendre un scalaire dans $[-\boldsymbol{\pi}, \boldsymbol{\pi}]$ ou un vecteur de points repères dans $\mathbb{R}^{\mathrm{d}}$, avec une exécution bien plus rapide.

GBRFF2 offre, quant à lui, de meilleures performances que GBRFF1.5, en particulier avec un très faible nombre de points repères, mais au prix d'un temps de calcul plus élevé. GBRFF2 est plus rapide que GBRFF1, pour $D>1$ ou tout aussi rapide pour $D=1$. De plus, GBRFF2 atteint des performances supérieures, même en comparaison avec GBRFF1 utilisant $D=20$ caractéristiques aléatoires.

Enfin, GBRFF2 présente d'excellents résultats comparés à l'état de l'art, obtenant le meilleur rang moyen parmi les 6 méthodes et, en moyenne, le meilleur taux d'erreurs. GBRFF2 a même montré sa capacité à apprendre des frontières de décisions complexes sur de petits ensembles de données.

5.5 Apprentissage de noyau (revisité)

Dans cette section, nous nous intéressons à nouveau au risque réel $\mathrm{R}_{\mathcal{D}}(k_q)$ qui correspond au risque réel de l'alignement de noyau sur une distribution de probabilité $\Delta_{\mathcal{D}}$, comme définie dans l'Équation (5.10). Les bornes présentées ci-après suggèrent une stratégie pour

l'alignement de noyau (ou l'apprentissage de noyau) similaire à celle proposée par Sinha et Duchi (2016). Nous soulignons que nos garanties ne s'appliquent que pour le risque de l'alignement de noyau, mais pas pour le classifieur appris avec le noyau. Ainsi, l'algorithme que nous proposons apprend un noyau indépendamment de la méthode de prédiction utilisée en aval. Cela contraste avec les cadres en une étape de Yang et al. (2015) et Oliva et al. (2016), qui apprennent un mélange de caractéristiques de noyaux aléatoires de manière *entièrement* bayésienne : ils s'appuient sur un modèle de génération de données, tandis que notre approche suppose uniquement que les exemples sont *i.i.d.*

5.5.1 Borne du second ordre

Le résultat suivant repose le fait que

$$\widehat{\mathrm{R}}_{\mathbb{S}}(h_{\boldsymbol{\omega}}) = \frac{1}{m^2-m}\sum_{i\neq j}^{m}\ell(h_{\boldsymbol{\omega}}(\boldsymbol{\kappa}_{ij}),\lambda_{ij})$$

est un estimateur de second ordre *non biaisé* de $\mathbb{E}_{(\boldsymbol{\kappa},\lambda)\sim\Delta_{\mathcal{D}}}\ell(h_{\boldsymbol{\omega}}(\boldsymbol{\kappa}),\lambda)$. Cela permet d'utiliser l'analyse PAC-Bayes pour les U-statistiques de Lever et al. (2013, Th. 7). Le Théorème 5.5.1 fournit une borne en généralisation pour le risque de l'alignement de noyau $\mathrm{R}_{\mathcal{D}}(k_q)$.

Théorème 5.5.1 (Lever et al. 2013)**.** Pour tout $\theta>0$ et pour toute une distribution *prior* p sur $\mathbb{R}^{\mathrm{d}}$, pour tout $\delta\in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\forall q \text{ sur } \mathbb{R}^{\mathrm{d}},\ \mathrm{R}_{\mathcal{D}}(k_q)\leq\widehat{\mathrm{R}}_{\mathbb{S}}(k_q)+\frac{1}{\theta}\left(\mathrm{KL}(q\|p)+\frac{t^2}{2m}+\ln\frac{1}{\epsilon}\right)\right]\geq 1-\delta.$$

À quelques constantes près, le Théorème 5.5.1 est similaire au Corollaire 5.4.1. En effet, les deux bornes sont minimisées par le même *pseudo-posterior* q^* de l'Équation (5.13) (avec $\tau=\theta$ pour le Théorème 5.5.1). Un fait intéressant est que nous nous affranchissons du terme $\ln(m+1)$ du Corollaire 5.4.1, ce qui fait que la borne du Théorème 5.5.1 converge à un taux de $O(\frac{1}{\sqrt{m}})$ lorsque $\theta=\sqrt{m}$.

5.5.2 Bornes du second ordre pour les f-divergences

Nous nous basons sur un résultat de Alquier et Guedj (2018) pour établir des bornes pour des exemples dépendants, où la KL-divergence est remplacée par d'autres f-divergences. Soit une fonction convexe $f()$ telle que $f(1)=0$ et $f(0)=\lim_{x\to 0^+}f(x)$, une f-divergence est définie par $D_f(q\|p)=\mathbb{E}_{\boldsymbol{\omega}\sim p}\,f\left(\frac{q(\boldsymbol{\omega})}{p(\boldsymbol{\omega})}\right)$. Le théorème suivant s'applique aux f-divergences définies par $f(x)=x^{\mu}-1$.

Théorème 5.5.2 (Borne PAC-Bayésienne du 2nd ordre)**.** Pour $\mu>1$ et pour toute une distribution *prior* p sur $\mathbb{R}^{\mathrm{d}}$, pour tout $\delta\in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\begin{array}{l}\forall q \text{ sur } \mathbb{R}^{\mathrm{d}},\\ \\ \mathrm{R}_{\mathcal{D}}(k_q)\leq\widehat{\mathrm{R}}_{\mathbb{S}}(k_q)+\left\{\begin{array}{ll}\left[\frac{1}{2\sqrt{m}}\right]^{\mu-1}\left[D_\mu(q\|p)+1\right]^{\frac{1}{\mu}}\left[\frac{1}{\delta}\right]^{1-\frac{1}{\mu}}, & \text{si } 1<\mu\leq 2\\ \left[\frac{1}{4m}\right]^{1-\frac{1}{\mu}}\left[D_\mu(q\|p)+1\right]^{\frac{1}{\mu}}\left[\frac{1}{\delta}\right]^{1-\frac{1}{\mu}}, & \text{si } \mu>2\end{array}\right.\end{array}\right]\geq 1-\delta.$$

Quand $\mu=2$, le Théorème 5.5.2 dépend alors de la distance χ^2 :

Corollaire 5.5.1 (Borne PAC-Bayésienne du 2nd ordre avec la distance χ^2)**.** Pour toute une distribution *prior* p sur $\mathbb{R}^{\mathrm{d}}$, pour tout $\delta \in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\forall q \text{ sur } \mathbb{R}^{\mathrm{d}}, \quad \mathsf{R}_{\mathcal{D}}(k_q) \leq \widehat{\mathsf{R}}_{\mathbb{S}}(k_q) + \sqrt{\frac{\chi^2(q\|p)+1}{4\,n\,\delta}}\right] \geq 1-\delta,$$

où $\chi^2(q\|p) = \mathbb{E}_{\boldsymbol{\omega}\sim p}\left[\frac{q(\boldsymbol{\omega})}{p(\boldsymbol{\omega})}\right]^2 - 1.$

Ce résultat ressemble à d'autres bornes PAC-Bayésiennes basées sur la distance χ^2 dans le cas de données *i.i.d.*, comme Honorio et Jaakkola (2014, Lem. 7), Bégin et al. (2016, Cor. 10) ou Alquier et Guedj (2018, Cor. 1).

5.5.3 Interprétation PAC-Bayésienne de l'alignement de noyau

Sinha et Duchi (2016) ont proposé un algorithme d'apprentissage de noyau qui pondèrent des caractéristiques aléatoires pour résoudre un un problème d'alignement de noyau.

Algorithme d'alignement de noyau. Soit p une distribution de transformée de Fourier à partir de laquelle N points sont tirés, notés $\Omega = \{\boldsymbol{\omega}_n\}_{n=1}^N \sim p^N$. Considérons également la distribution uniforme P sur l'ensemble d'hypothèses discret Ω, telle que $P(\boldsymbol{\omega}_n) = \frac{1}{N}$ et $h_n(\boldsymbol{\kappa}) = \cos(\boldsymbol{\omega}_m \cdot \boldsymbol{\kappa})$. Étant donné un ensemble $\mathbb{S}$ et les paramètres constants $\mu > 1$ et $\upsilon > 0$, l'algorithme d'optimisation proposé par Sinha et Duchi résout le problème :

$$\underset{Q\in\mathbb{R}_+^N}{\text{maximize}} \sum_{i=1}^m \sum_{j=1}^m \lambda_{ij} \sum_{n=1}^N Q_n h_n(\boldsymbol{\kappa}_{ij})\,, \tag{5.21}$$

$$\text{tel que } \sum_{n=1}^N Q_n = 1 \text{ et } D_\mu(Q\|P) \leq \upsilon\,. \tag{5.22}$$

Cette procédure itérative trouve une solution ϵ-sous-optimale en $O(N\log(\frac{1}{\epsilon}))$ étapes. La solution est un noyau appris $\widehat{k}_Q(\boldsymbol{\kappa}) = \frac{1}{N}\sum_{n=1}^N Q_n h_n(\boldsymbol{\kappa})$. Sinha et Duchi ont proposé d'utiliser cette méthode d'alignement pour réduire le nombre de caractéristiques en comparaison de la procédure classique des RFF (Section 5.2). Bien que cette méthode soit un apprentissage de noyau, les expériences empiriques montrent qu'avec un grand nombre de caractéristiques aléatoires, la procédure classique des RFF atteint une précision de prédiction tout aussi bonne. Cependant, on peut tirer (avec remise) $D < N$ caractéristiques de Ω selon Q. Pour une valeur relativement faible de D, apprendre un classifieur linéaire à partir du vecteur de caractéristiques aléatoires obtenu avec Q amène à de meilleurs résultats qu'avec la méthode classique des RFF avec le même nombre D de caractéristiques aléatoires.

Interprétation PAC-Bayésienne. Le problème d'optimisation des Équations (5.21-5.22) met en jeu le même compromis que le Théorème 5.5.2. En effet, maximiser l'Équation (5.21) revient à minimiser $\widehat{\mathsf{R}}_{\mathbb{S}}(k_q)$ et la contrainte de Équation (5.22) contrôle la f-divergence $D_\mu(Q\|P)$, qui est la même mesure de complexité que celle du Théorème 5.5.2. En outre, les expériences menées par Sinha et Duchi (2016) se focalisent sur la χ^2-divergence (quand $\mu=2$), ce qui correspond au compromis du Corollaire 5.5.1.

5.5.4 Apprentissage glouton de noyau

La méthode de Sinha et Duchi (2016) peut être adaptée pour minimiser la borne du Théorème 5.5.1, au lieu de celle du Théorème 5.5.2. Formellement, soit une distribution de transformée de Fourier p de laquelle nous tirons N points $\Omega = \{\boldsymbol{\omega}_n\}_{n=1}^N \sim p^N$, soit $P(\boldsymbol{\omega}_n) = \frac{1}{N}$ et $h_n(\boldsymbol{\kappa}) = \cos(\boldsymbol{\omega}_n \cdot \boldsymbol{\kappa})$. Étant donné l'ensemble d'apprentissage $\mathbb{S}$ et le paramètre $\beta > 0$, nous calculons le *pseudo-posterior* suivant pour $n = \{1, \dots, N\}$:

$$Q_n \;=\; \frac{1}{Z} \exp\left(-\beta \sqrt{m}\, \widehat{\mathsf{R}}_{\mathbb{S}}(h_n)\right). \tag{5.23}$$

Puis, nous effectuons un tirage avec remise de $D < N$ caractéristiques de Ω selon le *pseudo-posterior* Q. Ces caractéristiques sont utilisées pour transformer les points de l'ensemble d'apprentissage $\mathbf{x} \in \mathbb{R}^{\mathrm{d}}$ en un nouveau vecteur $\boldsymbol{\phi}(\mathbf{x}) \in \mathbb{R}^{2D}$ selon l'Équation (5.5). L'ensemble "transformé" est alors passé en entrée d'une méthode d'apprentissage de modèle linéaire.

En résumé, cette méthode est fortement inspirée par celle décrite dans la Section 5.5.3, mais la phase de calcul du *posterior* est plus rapide puisque nous utilisons une expression en forme close (Équation (5.23)). Une fois $\widehat{\mathsf{R}}_{\mathbb{S}}(h_n)$ calculé pour tout[3] h_n, nous pouvons faire varier le paramètr β et obtenir un nouveau *posterior* en $O(N)$ étapes.

5.5.5 Résumé des expériences

Nous avons étudié la méthode inspirée de Sinha et Duchi (2016) détaillée ci-dessus. Nous avons généré $N = 20\,000$ caractéristiques aléatoires selon p_σ comme donné par l'Équation (5.4), puis nous avons appris le *posterior* en utilisant deux stratégies : (okrff) le noyau optimisé de Sinha et Duchi donné par les Équations (5.21-5.22), et (pbrff) le *pseudo-posterior* donné par l'Équation (5.23). Pour les *posterior* obtenus, nous avons sous-échantillonné un nombre croissant de caractéristiques $D \in [1, 5\,000]$ pour créer la projection donnée par l'Équation (5.5), dans laquelle nous avons appris un svm linéaire. Nous nous sommes aussi comparés à (rff) qui correspond à la procédure classique des RFF (Section 5.2) avec D caractéristiques sélectionnées aléatoirement selon le prior p_σ. Nous avons observé que notre approche pbrff se comporte de manière similaire à okrff, avec un léger avantage pour cette dernière. Cependant, nous rappelons que le calcul du *posterior* de la première méthode est plus rapide. Les deux méthodes d'apprentissage de noyau ont une meilleure précision que l'algorithme rff classique pour un petit nombre de caractéristiques aléatoires, et des performances similaires pour un grand nombre de caractéristiques aléatoires.

5.6 Conclusion

Dans ce chapitre, nous avons proposé un point de vue original sur la méthode des *random Fourier features* (Rahimi et Recht, 2007) qui permet d'approximer un noyau. En considérant la transformée de Fourier comme une distribution *a priori* sur des fonctions trigonométriques, nous présentons deux types de bornes en généralisation PAC-Bayésiens qui bornent une perte sur l'alignement de noyau. En nous basant sur les résultats PAC-Bayes classiques du premier ordre, nous avons dérivé une stratégie basée sur des points repères qui apprend une représentation compacte des données. Nous avons également proposé deux algorithmes basés sur le *gradient boosting* pour apprendre conjointement de manière gloutonne la représentation compacte et le classifieur final. Ensuite, nous avons

3. Nous avons montré que chaque $\widehat{\mathsf{R}}_{\mathbb{S}}(h_n)$ peut être calculé en $O(m)$.

dérivé deux bornes en généralisation du second ordre. La première repose sur le théorème pour les U-statistiques de Lever et al. (2013). La seconde est un nouveau théorème PAC-Bayésien pour les f-divergences (remplaçant le terme de la KL-divergence). Nous montrons que cette dernière borne fournit une justification théorique à la méthode d'alignement de noyau de Sinha et Duchi (2016), et nous évaluons également empiriquement un algorithme similaire, mais plus simple, où la distribution d'alignement est obtenue par l'expression fermée du pseudo-postérieur PAC-Bayésien. Il est important de préciser que les bornes de ce chapitre ne concernent que la perte d'alignement de noyau, et non le prédicteur formé avec ce noyau.

Troisième partie

Algorithmes PAC-Bayésiens auto-certifiés

Algorithmes auto-certifiés pour le vote de majorité

6

Contexte

Dans les chapitres précédents, nous avons introduit des bornes de généralisation PAC-Bayésiennes utilisées comme sources d'inspiration et de justification pour le développement d'algorithmes. Cependant, ces approches ne fournissent pas de garanties valides accompagnant les modèles appris. En revanche, ce chapitre et le suivant présentent des algorithmes auto-certifiés pour le vote de majorité PAC-Bayésien : les algorithmes proposés minimisent directement des bornes de généralisation, garantissant ainsi que les modèles sont fournis avec des bornes valides.

Ce chapitre unifie deux travaux réalisés durant la thèse de Paul Viallard autour d'algorithmes auto-certifiés pour la minimisation de l'erreur du vote de majorité. Le premier a été publié à la conférence ECML-PKDD (Viallard et al., 2021a) et fait suite à l'un de mes travaux sur l'extension de la C-borne au cadre multiclasses publié dans le journal Neurocomputing (Laviolette et al., 2017). À notre connaissance, il s'agit du premier travail qui propose des algorithmes efficaces pour l'optimisation de la C-borne avec des valeurs de bornes en généralisation non triviales. Le second, publié à la conférence NeurIPS (Zantedeschi et al., 2021), s'attaque à des inconvénients des méthodes de la littérature en introduisant la notion de vote de majorité stochastique qui permet d'obtenir des bornes plus précises lors de l'apprentissage.

6.1 Introduction

Dans ce chapitre, nous introduisons de nouveaux algorithmes pour l'apprentissage de vote de majorité avec pour objectif de minimiser le risque réel du vote. Une première solution est de considérer la C-borne du Théorème 2.4.3, qui est une relaxation du risque du vote. Dans la littérature, une pratique consiste en la minimisation de son estimation empirique évaluée sur un ensemble d'apprentissage (*e.g.*, Roy et al., 2011 ; Bellet et al., 2014 ; Morvant et al., 2014 ; Morvant, 2015 ; Roy et al., 2016 ; Bauvin et al., 2020). Malgré le fait que les

algorithmes associés à ces publications soient empiriquement efficaces et justifiés par des analyses théoriques basées sur la C-borne, toutes ces méthodes minimisent une estimation empirique de la C-borne et non directement une borne en généralisation sur la valeur de la C-borne réelle. Ces procédures peuvent amener à bornes en généralisation non informatives et, par conséquent, à de faibles garanties sur le risque. Lorsqu'il s'agit de dériver un algorithme d'apprentissage qui minimise directement une borne PAC-Bayésienne, il est mentionné dans la littérature que l'optimisation d'une borne PAC-Bayes sur la C-borne n'est pas triviale (Lorenzen et al., 2019 ; Masegosa et al., 2020).

Dans la Section 6.3, nous couvrons tout d'abord trois visions de bornes en généralisation pour la C-borne (Seeger, 2002 ; McAllester, 2003 ; Lacasse et al., 2006) pour lesquelles nous dérivons des algorithmes pour les optimiser (via descente de gradient). Les algorithmes obtenus sont donc "auto-certifiés" (*self-bounding algorithms*, Freund, 1998) : le modèle appris est accompagné d'une borne supérieure valide sur le risque.

Ces algorithmes présentent néanmoins un défaut : ils ne minimisent pas directement le risque du vote de majorité puisqu'ils minimisent une de ses relaxations (la C-borne). Comme nous l'avons confirmé dans les expériences menées dans nos articles (Viallard et al., 2021a ; Zantedeschi et al., 2021), bien que la C-borne empirique soit une relaxation du risque du vote capable de contrôler suffisamment la diversité des votants tout en obtenant un risque faible, minimiser une C-borne PAC-Bayésienne n'amène pas nécessairement aux valeurs de bornes sur le risque les plus faibles. Pour s'attaquer à ce problème, nous introduisons dans la Section 6.4 le *vote de majorité stochastique* (pour chaque entrée, un vote de majorité est obtenu en tirant les poids selon une autre distribution). L'utilisation de ce vote stochastique permet d'obtenir des bornes en généralisation précises pour le vote de majorité classique.

6.2 Notations et contexte

Ce chapitre se place dans le cadre de la classification supervisée pour le vote de majorité PAC-Bayésien de la Section 2.4.1 avec $\mathbb{X} \subseteq \mathbb{R}^{\mathrm{d}}$ l'espace d'entrée et $\mathbb{Y}$ l'espace de sortie $\mathbb{Y} = \{-1, +1\}$ ou $\mathbb{Y} = \{1, 2, \ldots, l\}$. Nous supposons que $\mathcal{D}$ est une distribution fixe et inconnue sur $\mathbb{X} \times \mathbb{Y}$. Étant donné $\mathbb{H}$ un ensemble de votants $h : \mathbb{X} \to \mathbb{Y}$, une distribution *prior* $\pi \in \mathbb{M}^*(\mathbb{H})$ sur $\mathbb{H}$ et un ensemble d'apprentissage $\mathbb{S} = \{(\mathbf{x}_i, y_i)\}_{i=1}^m \sim \mathcal{D}^m$, l'objectif est d'apprendre un vote de majorité PAC-Bayésien (Définition 2.4.1). En d'autres termes, le but est d'apprendre une distribution *posterior* $\rho \in \mathbb{M}(\mathbb{H})$ sur $\mathbb{H}$ telle que le risque réel $\mathrm{R}_{\mathcal{D}}(\mathrm{MV}_\rho)$ du vote de majorité MV_ρ pondéré par ρ soit le plus faible possible, où

$$\mathrm{R}_{\mathcal{D}}(\mathrm{MV}_\rho) = \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} \mathrm{I}\left[\mathrm{MV}_\rho(\mathbf{x}) \neq y\right], \text{ avec } \mathrm{MV}_\rho(\mathbf{x}) = \mathop{\mathrm{argmax}}_{y' \in \mathbb{Y}} \mathop{\mathbb{E}}_{h\sim\rho} \mathrm{I}\left[h(\mathbf{x}) = y'\right].$$

Dans cette section, pour minimiser $\mathrm{R}_{\mathcal{D}}(\mathrm{MV}_\rho)$, nous utilisons une de ses relaxations, la C-borne (Théorème 2.4.3) définie par

$$\mathrm{R}_{\mathcal{D}}(\mathrm{MV}_\rho) \leq 1 - \frac{\left(1 - 2\mathrm{R}_{\mathcal{D}}(G_\rho)\right)^2}{1 - 2\mathrm{d}_{\mathcal{D}}(\rho)} = 1 - \frac{\Big(1 - \left[2\mathrm{e}_{\mathcal{D}}(\rho) + \mathrm{d}_{\mathcal{D}}(\rho)\right]\Big)^2}{1 - 2\mathrm{d}_{\mathcal{D}}(\rho)},$$

$$\text{avec } \mathrm{R}_{\mathcal{D}}(G_\rho) = \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} \mathop{\mathbb{E}}_{h\sim\rho} \mathrm{I}\left[h(\mathbf{x}) \neq y\right] \qquad \textit{(risque de Gibbs)},$$

$$\text{et } \mathrm{d}_{\mathcal{D}}(\rho) = 2 \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}'} \mathop{\mathbb{E}}_{h\sim\rho} \mathop{\mathbb{E}}_{h'\sim\rho} \mathrm{I}\left[h(\mathbf{x}) \neq y\right] \mathrm{I}\left[h'(\mathbf{x}) = y\right] \qquad \textit{(désaccord)},$$

$$\text{et } \mathrm{e}_{\mathcal{D}}(\rho) = \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}'} \mathop{\mathbb{E}}_{h\sim\rho} \mathop{\mathbb{E}}_{h'\sim\rho} \mathrm{I}\Big[h(\mathbf{x}) \neq y\Big] \mathrm{I}\Big[h'(\mathbf{x}) \neq y\Big] \qquad \textit{(erreur jointe)}.$$

Telle quelle cette C-borne n'est pas calculable, car elle dépend de la distribution $\mathcal{D}$, qui est inconnue. Nous avons donc besoin de bornes en généralisation pour majorer le risque réel du vote de majorité par une C-borne empirique estimée sur un échantillon. La combinaison de la C-borne avec la théorie PAC-Bayésienne offre une solution naturelle à l'analyse du risque réel du vote de majorité. Les bornes PAC-Bayésiennes de la littérature basées sur la C-borne sont rappelées dans la section suivante.

6.3 Bornes PAC-Bayésiennes et algorithmes pour le vote de majorité

Tout d'abord, nous rappelons, dans la Section 6.3.1, trois bornes PAC-Bayésiennes de l'état de l'art pour la C-borne, que nous désignerons par la suite sous le terme de "C-borne PAC-Bayésienne". Ensuite, nous présentons, dans la Section 6.3.2, nos trois algorithmes auto-certifiés pour minimiser directement ces trois C-bornes PAC-Bayésiennes. Il est important de rappeler que puisque la C-borne est une majoration du risque réel du vote de majorité, les C-bornes PAC-Bayésiennes permettent d'obtenir des bornes en généralisation calculables pour le risque réel du vote de majorité. Notons que ces C-bornes PAC-Bayésiennes ont été dérivées dans le cadre de la classification binaire, mais leur extension à la classification multiclasses est direct via l'utilisation de la $\frac{1}{2}$-marge que nous avons étudié dans Laviolette et al. (2017) et dont la définition est rappelée dans la Définition 2.4.4.

6.3.1 Les C-bornes PAC-Bayésiennes de la littérature

6.3.1.1 C-borne PAC-Bayésienne de Roy et al.

Roy et al. (2016) et Laviolette et al. (2017) ont démontré la C-borne PAC-Bayésienne la plus intuitive et interprétable. La preuve de cette borne a été développée par Roy et al. (2016) dans le cadre de la classification binaire, puis nous l'avons étendue à la classification multiclasse (Laviolette et al., 2017, Th. 3). Cette preuve consiste à majorer séparément avec une borne en généralisation le risque de Gibbs $\mathrm{R}_{\mathcal{D}}(G_\rho)$ et le désaccord $\mathrm{d}_{\mathcal{D}}(\rho)$ en utilisant une borne en généralisation PAC-Bayésienne à la McAllester (Théorème 2.4.6).

Théorème 6.3.1 (C-borne PAC-Bayésienne de Roy et al. (2016))**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble de votants $\mathbb{H}$, pour toute distribution *prior* $\pi\in\mathbb{M}^*(\mathbb{H})$, pour tout $\delta\in]0,1]$, on a :

$$\mathop{\mathbb{P}}_{\mathrm{S}\sim\mathcal{D}^m}\left[\begin{array}{l}\forall\rho\in\mathbb{M}(\mathbb{H}),\\ \mathrm{R}_{\mathcal{D}}(\mathrm{MV}_\rho)\leq \underbrace{1-\dfrac{\left[1-2\,\min\left\{\frac{1}{2},\,\widehat{\mathrm{R}}_{\mathrm{S}}(G_\rho)+\sqrt{\frac{1}{2m}\left[\mathrm{KL}(\rho\|\pi)+\ln\frac{4\sqrt{m}}{\delta}\right]}\right\}\right]^2}{1-2\,\max\left\{0,\,\widehat{\mathrm{d}}_{\mathrm{S}}(\rho)-\sqrt{\frac{1}{2m}\left[2\,\mathrm{KL}(\rho\|\pi)+\ln\frac{4\sqrt{m}}{\delta}\right]}\right\}}}_{=\;C^{\mathtt{M}}_{\mathrm{S}}(\rho)}\end{array}\right]\geq 1-\delta. \tag{6.1}$$

Bien qu'aucun algorithme ne minimise directement l'Équation (6.1), ce type de borne peut servir de justification théorique à l'optimisation de $\widehat{\mathrm{R}}_{\mathrm{S}}(G_\rho)$ et $\widehat{\mathrm{d}}_{\mathrm{S}}(\rho)$ comme cela est

fait, par exemple, dans les algorithmes `MinCq` (Roy et al., 2011), `CB-Boost` (Bauvin et al., 2020), ou $\texttt{MinCq}_{PW}$ (Morvant et al., 2014). Dans la Section 6.3.2.1, nous proposons un algorithme de minimisation directe de cette borne.

Cependant, si le nombre d'exemples m est faible et si le risque de Gibbs est proche de $\frac{1}{2}$, la valeur de cette C-borne PAC-Bayésienne sera proche de 1. Pour s'affranchir de cet inconvénient, une solution est d'utiliser une borne PAC-Bayésienne à la Seeger (2002) (Théorème 2.4.8) qui amène à des bornes plus précises. Dans la suite, nous rappelons deux bornes basées sur cette approche : la première dans le Théorème 6.3.2 fait intervenir le risque de Gibbs $\widehat{\mathsf{R}}_{\mathbb{S}}(G_\rho)$ et le désaccord $\widehat{\mathrm{d}}_{\mathbb{S}}(\rho)$, et la seconde dans le Théorème 6.3.3 fait intervenir l'erreur jointe $\widehat{\mathrm{e}}_{\mathbb{S}}(\rho)$ et le désaccord $\widehat{\mathrm{d}}_{\mathbb{S}}(\rho)$.

6.3.1.2 C-borne PAC-Bayésienne de Germain et al.

Comme pour le Théorème 6.3.1, la borne suivante majore indépendamment le risque de Gibbs et le désaccord (voir PAC-Bound 1 de Germain et al. (2015)).

Théorème 6.3.2 (C-borne PAC-Bayésienne de Germain et al. (2015))**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble de votants $\mathbb{H}$, pour toute distribution *prior* $\pi\in\mathbb{M}^*(\mathbb{H})$, pour tout $\delta\in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\begin{array}{l}\forall\rho\in\mathbb{M}(\mathbb{H}),\\ \mathsf{R}_{\mathcal{D}}(\mathsf{MV}_\rho)\le \underbrace{1-\frac{\left[1-2\,\min\left\{\frac{1}{2},\overline{\mathrm{kl}}\left(\widehat{\mathsf{R}}_{\mathbb{S}}(G_\rho)\;\middle|\;\frac{1}{m}\left[\mathrm{KL}(\rho\|\pi)+\ln\frac{4\sqrt{m}}{\delta}\right]\right)\right\}\right]^2}{1-2\,\max\left\{0,\underline{\mathrm{kl}}\left(\widehat{\mathrm{d}}_{\mathbb{S}}(\rho)\;\middle|\;\frac{1}{m}\left[2\,\mathrm{KL}(\rho\|\pi)+\ln\frac{4\sqrt{m}}{\delta}\right]\right)\right\}}}_{=\;C^{\mathsf{S}}_{\mathbb{S}}(\rho)}\end{array}\right]\ge 1-\delta. \tag{6.2}$$

L'Équation (6.2) "à la Seeger" est plus précise que celle "à la McAllester" de l'Équation (6.1) pour les mêmes raisons qu'évoquées dans la Section 2.4.3. Cependant, un inconvénient de cette borne est le fait que le risque de Gibbs et le désaccord soient majorés indépendamment.

6.3.1.3 C-borne PAC-Bayésienne de Lacasse et al.

Lacasse et al. (2006) ont proposé de majorer simultanément l'erreur jointe $\mathrm{e}_{\mathcal{D}}(\rho)$ et le désaccord $\mathrm{d}_{\mathcal{D}}(\rho)$. Pour ce faire, il faut trouver la pire valeur de la C-borne qui peut être atteinte avec un couple erreur jointe et désaccord, noté (e,d), appartenant à l'ensemble $\mathbb{A}_{\mathbb{S}}(\rho)$ défini par

$$\mathbb{A}_{\mathbb{S}}(\rho)=\left\{(e,d)\;\middle|\;\mathrm{kl}\left(\widehat{\mathrm{e}}_{\mathbb{S}}(\rho),\widehat{\mathrm{d}}_{\mathbb{S}}(\rho)\|e,d\right)\le\frac{1}{m}\left[2\,\mathrm{KL}(\rho\|\pi)+\ln\frac{2\sqrt{m}+m}{\delta}\right],\right.$$
$$\left.\text{et } d\le 2\sqrt{e}-2e\,,\ \text{et } 2e+d<1\right\},$$

où $\quad \mathrm{kl}(q_1,q_2\|p_1,p_2)=q_1\ln\frac{q_1}{p_1}+q_2\ln\frac{q_2}{p_2}+(1-q_1-q_2)\ln\frac{1-q_1-q_2}{1-p_1-p_2}.$

En se basant sur $\mathbb{A}_{\mathbb{S}}(\rho)$, Lacasse et al. (2006) ont obtenu le résultat suivant.

Théorème 6.3.3 (C-borne PAC-Bayésienne de Lacasse et al. (2006))**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble de votants $\mathbb{H}$, pour toute distribution *prior* $\pi\in\mathbb{M}^*(\mathbb{H})$, pour tout $\delta\in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\mathsf{R}_{\mathcal{D}}(\mathsf{MV}_\rho)\leq\sup_{(e,d)\in\mathbb{A}_{\mathbb{S}}(\rho)}\left(1-\frac{\left[1-(2e+d)\right]^2}{1-2d}\right)\right]\geq 1-\delta. \tag{6.3}$$

Cette C-borne PAC-Bayésienne peut être plus difficile à calculer : elle requiert la résolution d'un problème d'optimisation (convexe) pour trouver une valeur de la borne.

6.3.2 Algorithmes de minimisation des C-bornes PAC-Bayésiennes

Cette section présente nos trois algorithmes auto-certifiés qui minimisent les C-bornes PAC-Bayésiennes.

6.3.2.1 Algorithme basé sur l'Équation (6.1)

L'Algorithme 6.1, ci-dessous, permet de minimiser directement la C-borne PAC-Bayésienne du Théorème 6.3.1 par descente de gradient stochastique. Un aspect important de l'optimisation est que si $\widehat{\mathsf{R}}_{\mathbb{S}}(G_\rho)+\sqrt{\frac{1}{2m}[\mathrm{KL}(\rho\|\pi)+\ln\frac{4\sqrt{m}}{\delta}]}\geq\frac{1}{2}$, alors le gradient du numérateur de $C^{\mathtt{M}}_{\mathbb{S}}(\rho)$ par rapport à ρ est 0, rendant alors l'optimisation impossible. Nous proposons donc de minimiser le problème d'optimisation contraint suivant :

$$\min_{\rho\in\mathbb{M}(\mathbb{H})}\underbrace{\left\{1-\frac{\left(1-2\min\left\{\frac{1}{2},\widehat{\mathsf{R}}_{\mathbb{S}}(G_\rho)+\sqrt{\frac{1}{2m}\left[\mathrm{KL}(\rho\|\pi)+\ln\frac{4\sqrt{m}}{\delta}\right]}\right\}\right)^2}{1-2\max\left\{0,\widehat{\mathrm{d}}_{\mathbb{S}}(\rho)-\sqrt{\frac{1}{2m}\left[2\,\mathrm{KL}(\rho\|\pi)+\ln\frac{4\sqrt{m}}{\delta}\right]}\right\}}\right\}}_{=\;C^{\mathtt{M}}_{\mathbb{S}}(\rho)}$$

$$\text{tel que}\quad\widehat{\mathsf{R}}_{\mathbb{S}}(G_\rho)+\sqrt{\frac{1}{2m}\left[\mathrm{KL}(\rho\|\pi)+\ln\frac{4\sqrt{m}}{\delta}\right]}\leq\frac{1}{2}.$$

De cette formulation, nous obtenons le problème non contraint :

$$\min_{\rho\in\mathbb{M}(\mathbb{H})}\left\{C^{\mathtt{M}}_{\mathbb{S}}(\rho)+\mathrm{B}\left(\widehat{\mathsf{R}}_{\mathbb{S}}(G_\rho)+\sqrt{\frac{1}{2m}\left[\mathrm{KL}(\rho\|\pi)+\ln\frac{4\sqrt{m}}{\delta}\right]}-\frac{1}{2}\right)\right\},$$

où $\mathrm{B}()$ est la fonction barrière définie par $\mathrm{B}(a)=0$ si $a\leq 0$ ou $\mathrm{B}(a)=+\infty$ sinon. La nature de $\mathrm{B}()$ fait que la fonction objectif sera infinie lorsque $a>0$. Pour contourner ce problème, nous remplaçons $\mathrm{B}()$ par son approximation, proposée par Kervadec et al. (2022), appelée "extension log-barrière" *(log-barrier extension)* et définie par

$$\mathrm{B}_\lambda(a)=\begin{cases}-\frac{1}{\lambda}\ln(-a), & \text{si } a\leq-\frac{1}{\lambda^2},\\ \lambda a-\frac{1}{\lambda}\ln(\frac{1}{\lambda^2})+\frac{1}{\lambda}, & \text{sinon.}\end{cases}$$

L'extension log-barrière est paramétrée par λ. La fonction $\mathrm{B}_\lambda()$ tend vers $\mathrm{B}()$ lorsque λ tend vers $+\infty$ (voir la Figure 6.1). Contrairement à la log-barrière standard [1], la fonction $\mathrm{B}_\lambda()$

1. Voir Boyd et Vandenberghe (2004) pour une introduction à la log-barrière et aux méthodes de points intérieurs.

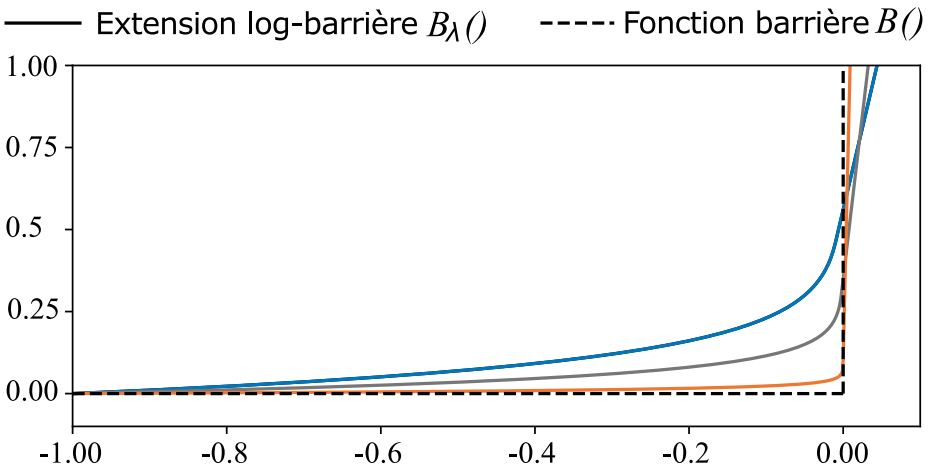


Figure 6.1. *La fonction barrière* $\mathrm{B}()$ *(en pointillés) et l'extension log-barrière* $\mathrm{B}_\lambda()$ *avec trois valeurs de* λ *différentes :* 10 *(trait plein bleu),* 20 *(trait plein gris),* 100 *(trait plein orange). Plus* λ *est grand, plus les fonctions* $\mathrm{B}_\lambda()$ *et* $\mathrm{B}()$ *sont proches.*

est différentiable même lorsque les contraintes ne sont pas satisfaites, *i.e.*, quand $a > 0$. La fonction $\mathrm{B}_\lambda()$ permet donc de considérer la contrainte $\widehat{\mathrm{R}}_{\mathbb{S}}(G_\rho) + \sqrt{\frac{1}{2m}[\mathrm{KL}(\rho\|\pi) + \ln\frac{4\sqrt{m}}{\delta}]} \leq \frac{1}{2}$. De plus, lorsque le nombre d'exemples m est grand, nous pouvons estimer la C-borne PAC-Bayésienne $C^{\mathtt{M}}_{\mathbb{S}}(\rho)$ et le risque de Gibbs $\widehat{\mathrm{R}}_{\mathbb{S}}(G_\rho)$ par mini-lot $\mathbb{U} \subseteq \mathbb{S}$. Concrètement, la fonction objectif, que nous minimisons par descente de gradient stochastique via l'Algorithme 6.1, est

$$F^{\mathtt{M}}_{\mathbb{U}}(\rho) \;=\; C^{\mathtt{M}}_{\mathbb{U}}(\rho) + \mathrm{B}_\lambda\left(r_{\mathbb{U}}(\rho) + \sqrt{\frac{1}{2m}\left[\mathrm{KL}(\rho\|\pi) + \ln\frac{4\sqrt{m}}{\delta}\right]} - \frac{1}{2}\right).$$

Étant donné λ, la procédure d'optimisation trouvera une solution avec un bon compromis entre minimiser $C^{\mathtt{M}}_{\mathbb{U}}(\rho)$ et l'extension log-barrière $\mathrm{B}_\lambda()$. Comme attendu et comme l'ont montré les expériences que nous avons menées, la minimisation de cette borne à la McAllester n'amène pas à la borne la plus précise.

Algorithme 6.1 Minimisation de l'Équation (6.1) par descente de gradient stochastique

Entrées : ensemble $\mathbb{S}$, *prior* $\pi \in \mathbb{M}^*(\mathbb{H})$, fonction objectif $F^{\mathtt{M}}_{\mathbb{S}}(\rho)$, nombre d'itérations T

$\rho \leftarrow \pi$
pour $t \leftarrow 1$ à T **faire**
  pour tout mini-lot $\mathbb{U} \subseteq \mathbb{S}$ **faire**
    $\rho \leftarrow$ Mise à jour de ρ avec $F^{\mathtt{M}}_{\mathbb{U}}(\rho)$ par descente de gradient [2]
retourner ρ

6.3.2.2 Algorithme basé sur l'Équation (6.2)

Pour obtenir de meilleures garanties en généralisation, nous proposons de minimiser la C-borne PAC-Bayésienne à la Seeger du Théorème 6.3.2. L'objectif est alors de minimiser

2. La mise à jour de ρ peut être réalisée par descente de gradient ou avec la mise à jour d'un algorithme comme Adam (Kingma et Ba, 2015) ou COCOB (Orabona et Tommasi, 2017).

le problème suivant :

$$\min_{\rho\in\mathbb{M}(\mathbb{H})} \underbrace{\left\{1-\frac{\left(1-2\min\left\{\frac{1}{2},\overline{\mathrm{kl}}\left(\widehat{\mathrm{R}}_{\mathbb{S}}(G_\rho)\;\middle|\;\frac{1}{m}\left[\mathrm{KL}(\rho\|\pi)+\ln\frac{4\sqrt{m}}{\delta}\right]\right)\right\}\right)^2}{1-2\max\left\{0,\underline{\mathrm{kl}}\left(\widehat{\mathrm{d}}_{\mathbb{S}}(\rho)\;\middle|\;\frac{1}{m}\left[2\,\mathrm{KL}(\rho\|\pi)+\ln\frac{4\sqrt{m}}{\delta}\right]\right)\right\}}\right\}}_{=\;C^{\mathbb{S}}_{\mathbb{S}}(\rho)},$$

$$\text{tel que}\quad \overline{\mathrm{kl}}\left(\widehat{\mathrm{R}}_{\mathbb{S}}(G_\rho)\;\middle|\;\frac{1}{m}\left[\mathrm{KL}(\rho\|\pi)+\ln\frac{4\sqrt{m}}{\delta}\right]\right)\leq\frac{1}{2}.$$

Pour les mêmes raisons que pour l'Algorithme 6.1, nous proposons de résoudre le problème par descente de gradient stochastique par mini-lot $\mathbb{U}\subseteq\mathbb{S}$ et de minimiser la fonction objectif :

$$F^{\mathbb{S}}_{\mathbb{U}}(\rho)=C^{\mathbb{S}}_{\mathbb{U}}(\rho)+\mathrm{B}_\lambda\left(\overline{\mathrm{kl}}\left(r_{\mathbb{U}}(\rho)\;\middle|\;\frac{1}{m}\left[\mathrm{KL}(\rho\|\pi)+\ln\frac{4\sqrt{m}}{\delta}\right]\right)-\frac{1}{2}\right).$$

Pour l'évaluation de $\overline{\mathrm{kl}}()$ et $\underline{\mathrm{kl}}()$ nous utilisons l'Algorithme 2.1 (proposé par Reeb et al., 2018) qui affine itérativement un intervalle $[p_{\min},p_{\max}]$ avec $p\in[p_{\min},p_{\max}]$ tel que $\mathrm{kl}(q\|p)=\psi$. Pour le calcul des dérivées par rapport à ρ, nous utilisons la règle de dérivation en chaîne avec une méthode d'apprentissage profond (*e.g.*, `PyTorch`, Paszke et al., 2019) et les dérivées rappelées dans l'Équation (2.23). L'algorithme général est résumé dans l'Algorithme 6.2.

Algorithme 6.2 Minimisation de l'Équation (6.2) par descente de gradient stochastique

Entrées : ensemble $\mathbb{S}$, *prior* $\pi\in\mathbb{M}^*(\mathbb{H})$, fonction objectif $F^{\mathbb{S}}_{\mathbb{S}}(\rho)$, nombre d'itérations T

1: $\rho\leftarrow\pi$
2: **pour** $t\leftarrow 1$ à T **faire**
3: **pour tout** mini-lot $\mathbb{U}\subseteq\mathbb{S}$ **faire**
4: Compute $F^{\mathbb{S}}_{\mathbb{U}}(\rho)$ avec l'Algorithme 2.1
5: $\rho\leftarrow$ Mise à jour de ρ avec $F^{\mathbb{S}}_{\mathbb{U}}(\rho)$ par descente de gradient
6: **retourner** ρ

6.3.2.3 Algorithme basé sur l'Équation (6.3)

L'Équation (6.3) a l'intérêt de majorer simultanément l'erreur jointe $\mathrm{e}_{\mathcal{D}}(\rho)$ et le désaccord $\mathrm{d}_{\mathcal{D}}(\rho)$. Cependant, comme mentionné dans la Section 6.3.1.3, son optimisation peut être difficile. Pour faciliter la manipulation de la borne, nous reformulons les contraintes impliquées dans l'ensemble $\mathbb{A}_{\mathbb{S}}(\rho)$ pour obtenir la C-borne suivante.

Théorème 6.3.4 (Reformulation de la C-borne PAC-Bayésienne de Lacasse et al.)**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble de votants $\mathbb{H}$, pour toute distribution *prior* $\pi\in\mathbb{M}^*(\mathbb{H})$, pour tout $\delta\in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\begin{array}{c}\mathrm{R}_{\mathcal{D}}(\mathrm{MV}_\rho)\leq\sup\limits_{(e,d)\in\mathbb{A}'_{\mathbb{S}}(\rho)}\left[1-\dfrac{\left[1-(2e+d)\right]^2}{1-2d}\right],\\ \text{où }\mathbb{A}'_{\mathbb{S}}(\rho)=\left\{(e,d)\;\middle|\;\mathrm{kl}\Big(\widehat{\mathrm{e}}_{\mathbb{S}}(\rho),\widehat{\mathrm{d}}_{\mathbb{S}}(\rho)\|e,d\Big)\leq\dfrac{1}{m}\left[2\,\mathrm{KL}(\rho\|\pi)+\ln\dfrac{2\sqrt{m}+m}{\delta}\right],\right.\\ \left.\text{et }d\leq 2\sqrt{\min\left\{e,\tfrac{1}{4}\right\}}-2e,\ \text{et }d<\dfrac{1}{2}\right\}\end{array}\right]\geq 1-\delta. \tag{6.4}$$

L'Équation (6.4) suggère le problème contraint suivant :

$$\min_{\rho\in\mathbb{M}(\mathbb{H})}\left\{\sup_{(e,d)\in\left[0,\frac{1}{2}\right]^2}\left(1-\frac{\left[1-(2e+d)\right]^2}{1-2d}\right)\ \text{avec}\ (e,d)\in\mathbb{A}'_{\mathbb{S}}(\rho)\right\},$$
$$\text{tel que}\ \ 2\,\widehat{\mathrm{e}}_{\mathbb{S}}(\rho)+\widehat{\mathrm{d}}_{\mathbb{S}}(\rho)\leq 1.$$

En fait, nous pouvons formuler ce problème comme un problème non contraint en utilisant la fonction barrière. On a

$$\min_{\rho\in\mathbb{M}(\mathbb{H})}\left\{\max_{(e,\,d)\in\left[0,\frac{1}{2}\right]^2}\left\{\ C^{\mathrm{L}}(e,d)-\mathrm{B}\Big(d-2\sqrt{\min\left(e,\tfrac{1}{4}\right)}-2e\Big)-\mathrm{B}\Big(d-\tfrac{1}{2}\Big)\right.\right.$$
$$\left.\left.-\,\mathrm{B}\left(\mathrm{kl}\left(\widehat{\mathrm{e}}_{\mathbb{S}}(\rho),\widehat{\mathrm{d}}_{\mathbb{S}}(\rho)\|e,d\right)-\frac{1}{m}\Big[2\,\mathrm{KL}(\rho\|\pi)+\ln\tfrac{2\sqrt{m}+m}{\delta}\Big]\right)\right\}\right.$$
$$\left.+\,\mathrm{B}\left(2\widehat{\mathrm{e}}_{\mathbb{S}}(\rho)+\widehat{\mathrm{d}}_{\mathbb{S}}(\rho)-1\right)\right\},\tag{6.5}$$

avec $C^{\mathrm{L}}(e,d)=\begin{cases}1-\frac{[1-(2e+d)]^2}{1-2d} & \text{si } d<\frac{1}{2},\\ 1 & \text{sinon.}\end{cases}$

Ce problème ne peut pas directement être optimisé par descente de gradient stochastique. Le problème d'optimisation est de type min-max, *i.e.*, pour chaque étape de descente, avant de pouvoir mettre à jour le *posterior* ρ, il faut trouver le couple (e,d) qui maximise $C^{\mathrm{L}}(e,d)$ sous les contraintes définies par $\mathbb{A}'_{\mathbb{S}}(\rho)$.

Tout d'abord, pour trouver le couple optimal (e,d), nous nous concentrons sur le problème de maximisation quand $\widehat{\mathrm{e}}_{\mathbb{S}}(\rho)$ et $\widehat{\mathrm{d}}_{\mathbb{S}}(\rho)$ sont fixés. Cependant, la fonction $C^{\mathrm{L}}(e,d)$ n'est pas concave pour tout $(e,d)\in\mathbb{R}^2$, ce qui implique que la mise en œuvre de cette maximisation peut être difficile[3]. Heureusement, $C^{\mathrm{L}}(e,d)$ est quasi-concave (Germain et al., 2015) pour $(e,d)\in[0,1]\times[0,\frac{1}{2}]$. Par définition de la quasi-concavité, pour tout $\alpha\in[0,1]$, on a

$$\left\{(e,d)\,\middle|\,1-\frac{\left[1-(2e+d)\right]^2}{1-2d}\geq 1-\alpha\right\}\iff\left\{(e,d)\ \middle|\ \alpha(1-2d)-\left[1-(2e+d)\right]^2\geq 0\right\}.$$

Ainsi, pour $\alpha\in[0,1]$ fixé, nous cherchons (e,d) qui maximise $C^{\mathrm{L}}(e,d)$ tout en respectant les contraintes de $\mathbb{A}'_{\mathbb{S}}(\rho)$. Cela revient à résoudre le problème suivant pour $\alpha\in[0,1]$

$$\max_{(e,d)\in[0,\frac{1}{2}]^2}\quad\alpha(1-2d)-\left[1-(2e+d)\right]^2\tag{6.6}$$
$$\text{tel que}\ \ d\leq 2\sqrt{\min\left(e,\tfrac{1}{4}\right)}-2e$$
$$\text{et}\qquad\mathrm{kl}\left(\widehat{\mathrm{e}}_{\mathbb{S}}(\rho),\widehat{\mathrm{d}}_{\mathbb{S}}(\rho)\|e,d\right)\leq\frac{1}{m}\left[2\,\mathrm{KL}(\rho\|\pi)+\ln\frac{2\sqrt{m}+m}{\delta}\right].$$

En fait, le but est de trouver $\alpha\in[0,1]$ tel que la maximisation de l'Équation (6.6) conduise à la valeur de $1-\alpha$ qui correspond à la plus grande valeur de $C^{\mathrm{L}}(e,d)$ respectant les

3. Par exemple, en utilisant `CVXPY` (Diamond et Boyd, 2016), qui utilise la *Disciplined Convex Programming* (DCP, Grant et al., 2006), la maximisation d'une fonction non concave n'est pas possible.

Algorithme 6.3 Minimisation de l'Équation (6.4) par descente de gradient stochastique

Entrées : Ensemble $\mathbb{S}$, *prior* $\pi \in \mathbb{M}^*(\mathbb{H})$, fonction objectif $F_{\mathbb{S}}^{e^*,d^*}(\rho)$, nombre d'itération T

$\rho \leftarrow \pi$
pour $t \leftarrow 1$ à T **faire**
  pour tout mini-lot $\mathbb{U} \subseteq \mathbb{S}$ **faire**
    $(e^*, d^*) \leftarrow$ MAX-e-$d\big(\hat{e}_{\mathbb{U}}(\rho), \hat{d}_{\mathbb{U}}(\rho)\big)$
    $\rho \leftarrow$ Mise à jour de ρ avec $F_{\mathbb{U}}^{e^*,d^*}(\rho)$ par descente de gradient
retourner ρ

Entrées : Ensemble $\mathbb{S}$, erreur jointe $\widehat{\mathrm{e}}_{\mathbb{S}}(\rho)$, désaccord $\widehat{\mathrm{d}}_{\mathbb{S}}(\rho)$, tolérance ϵ

function MAX-e-$d\big(\widehat{\mathrm{e}}_{\mathbb{S}}(\rho), \widehat{\mathrm{d}}_{\mathbb{S}}(\rho)\big)$
  $\alpha_{\min} = 0$ et $\alpha_{\max} = 1$
  tant que $\alpha_{\max} - \alpha_{\min} > \epsilon$ **faire**
    $\alpha = \frac{1}{2}(\alpha_{\min} + \alpha_{\max})$
    $(e, d) \leftarrow$ Résoudre l'Équation (6.6)
    si $C^{\mathrm{L}}(e, d) \geq 1-\alpha$ **alors** $\alpha_{\max} \leftarrow \alpha$ **sinon** $\alpha_{\min} \leftarrow \alpha$
  retourner (e, d)

contraintes. Pour cela, nous utilisons la méthode de la dichotomie pour l'optimisation quasi-convexe (Boyd et Vandenberghe, 2004) qui est résumée dans la fonction MAX-e-$d()$ de l'Algorithme 6.3. Notons (e^*, d^*) la solution de l'Équation (6.6), il reste à résoudre le problème

$$\min_{\rho \in \mathbb{M}(\mathbb{H})} \left\{ \mathrm{B}\Big(2\,\widehat{\mathrm{e}}_{\mathbb{S}}(\rho) + \widehat{\mathrm{d}}_{\mathbb{S}}(\rho) - 1\Big) - \mathrm{B}\left(\mathrm{kl}\left(\widehat{\mathrm{e}}_{\mathbb{S}}(\rho), \widehat{\mathrm{d}}_{\mathbb{S}}(\rho) \| e^*, d^*\right) - \frac{1}{m}\left[2\,\mathrm{KL}(\rho\|\pi) + \ln \frac{2\sqrt{m}+m}{\delta} \right] \right) \right\}.$$

Pour obtenir une fonction objectif compatible avec la descente de gradient stochastique, nous apportons deux modifications à ce problème de minimisation : *(i)* nous remplaçons $\mathrm{B}()$ par l'extension log-barrière $\mathrm{B}_\lambda()$ et *(ii)* nous approximons le désaccord et l'erreur jointe par mini-lots $\mathbb{U} \subseteq \mathbb{S}$. Nous obtenons la fonction objectif suivante :

$$F_{\mathbb{U}}^{e^*,d^*}(\rho) = \mathrm{B}_\lambda\Big(2e_{\mathbb{U}}(\rho) + d_{\mathbb{U}}(\rho) - 1\Big) - \mathrm{B}_\lambda\left(\mathrm{kl}\left(e_{\mathbb{U}}(\rho), d_{\mathbb{U}}(\rho) \| e^*, d^*\right) - \frac{1}{m}\left[2\,\mathrm{KL}(\rho\|\pi) + \ln \frac{2\sqrt{m}+m}{\delta} \right] \right).$$

La méthode est résumée dans l'Algorithme 6.3.

6.4 Bornes PAC-Bayésiennes et algorithmes pour le vote de majorité stochastique

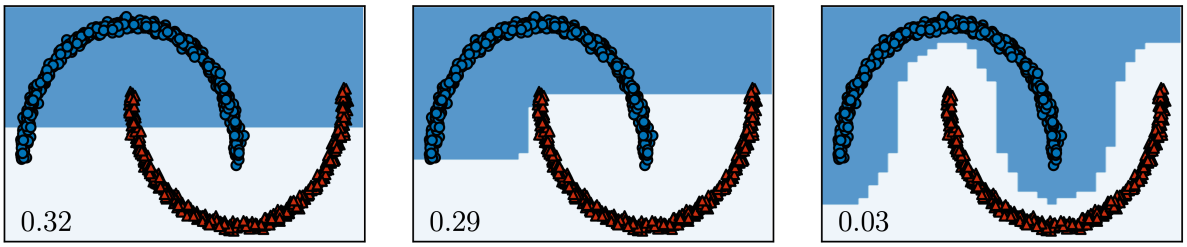
Dans la section précédente, nous avons dérivé des algorithmes auto-certifiés pour minimiser des bornes PAC-Bayésiennes sur la C-borne pour minimiser le risque réel du vote de majorité. Comme confirmé par les expériences que nous avons menées (Viallard et al., 2021a ; Zantedeschi et al., 2021), bien que la C-borne empirique soit une relaxation du risque du vote capable de contrôler suffisamment la diversité des votants tout en maintenant un risque faible, minimiser une C-borne PAC-Bayésienne ne conduit pas nécessairement aux valeurs de bornes les plus faibles. Cela s'explique par le fait que ces algorithmes ne minimisent pas directement le risque du vote de majorité, ce qui est également vrai pour toutes les relaxations mentionnées dans la Section 2.4.2. Un peu plus précisément, étant donné une distribution ρ sur un ensemble de votants $\mathbb{H}$, nous avons vu les trois relaxations suivantes :

(i) 2 fois le risque Gibbs $\mathrm{R}_{\mathcal{D}}(G_\rho)$ (Théorème 2.4.1),

(ii) 4 fois l'erreur jointe $e_{\mathcal{D}}(\rho)$ (Théorème 2.4.2),

(iii) La C-borne $C_{\mathcal{D}}(\rho)$ (Théorème 2.4.3).

La Figure 6.2 illustre la frontière de décision des modèles appris à partir de l'estimation empirique de ces approximations : seule la C-borne empirique contrôle suffisamment la diversité des votants pour obtenir $\widehat{\mathrm{R}}_{\mathbb{S}}(\mathrm{MV}_\rho)=0$. En effet, sur la figure, nous observons que les deux premières relaxations ne sont pas efficaces pour minimiser le risque empirique du vote de majorité, alors que la C-borne permet d'obtenir, ici, un risque empirique nul avec une valeur de borne proche de 0.



Minimisation de $2\widehat{\mathrm{R}}_{\mathbb{S}}(G_\rho)$ | Minimisation de $4\widehat{\mathrm{e}}_{\mathbb{S}}(\rho)$ | Minimisation de $\widehat{C}_{\mathbb{S}}(\rho)$

Figure 6.2. *Frontières de décision des votes de majorité obtenus par minimisation des bornes supérieures sur le risque empirique du vote de majorité avec les relaxations rappelées dans les Théorèmes 2.4.1, 2.4.2 et 2.4.3 sur le jeu de données jouet des lunes avec un ensemble d'apprentissage de taille $m = 1\,000$. Sur chaque figure, la valeur de la relaxation est reportée en bas à gauche.*

Lacasse et al. (2010) ont introduit une autre relaxation précise du risque empirique du vote de majorité en se basant sur le "vote de majorité randomisé" *(randomized majority vote)* défini par

$$\mathrm{MV}_\sigma(\mathbf{x}) = \underset{y'\in\mathbb{Y}}{\operatorname{argmax}} \underset{h\sim\sigma}{\mathbb{E}} \mathrm{I}\left[h(\mathbf{x})=y'\right],$$

où σ est construit comme suit : N votants $\mathbb{H}'=\{h_1,\ldots,h_N\}$ sont tirés selon la distribution ρ, puis on considère la distribution *posterior* uniforme σ définie par $\forall h\in\mathbb{H}',\ \sigma(h)=\frac{1}{N}$. Lacasse et al. (2010) définissent le risque empirique du vote de majorité randomisé par

$$\underset{\substack{(\mathbf{x},y)\sim\mathbb{S}\\ \mathrm{MV}_\sigma\sim\rho^N}}{\mathbb{P}}\left(\mathrm{MV}_\sigma(\mathbf{x})\neq y\right) \leq \underbrace{\underset{(\mathbf{x},y)\sim\mathbb{S}}{\mathbb{E}}\left[\sum_{j=\lceil\frac{N}{2}\rceil}^{N}\binom{N}{j}\left[\frac{1-\widehat{m_\rho}(\mathbf{x},y)}{2}\right]^j\left[1-\frac{1-\widehat{m_\rho}(\mathbf{x},y)}{2}\right]^{(N-j)}\right]}_{=\,b^N_{\mathbb{S}}(\rho)},$$

où $\widehat{m_\rho}(\mathbf{x},y)$ est la $\frac{1}{2}$-marge (Définition 2.4.4) et où l'abus de notation $(\mathbf{x},y)\sim\mathbb{S}$ indique qu'un exemple est tiré dans l'échantillon d'apprentissage $\mathbb{S}$ selon la distribution uniforme sur $\mathbb{S}$. Soit un exemple $(\mathbf{x},y)\sim\mathbb{S}$, la somme correspond alors à la probabilité qu'au moins $\frac{N}{2}$ des N votants tirés selon ρ commettent une erreur. Il s'agit de la fonction de répartition complémentaire de la distribution binomiale avec le paramètre $\frac{1}{2}(1-\widehat{m_\rho}(\mathbf{x},y))$ et $\lceil\frac{N}{2}\rceil$ tirages. Ainsi, $b^N_{\mathbb{S}}(\rho)$ est l'espérance de la fonction de répartition complémentaire sur $\mathbb{S}$. De plus, le vote de majorité *randomisé* est lié au vote de majorité *classique* MV_ρ grâce au terme $b^N_{\mathbb{S}}(\rho)$ qui est une relaxation de son risque empirique (Lacasse et al., 2010). On a

$$\widehat{\mathrm{R}}_{\mathbb{S}}(\mathrm{MV}_\rho) \ \leq\ 2\, b^N_{\mathbb{S}}(\rho). \tag{6.7}$$

Plus N est élevé, plus l'approximation du risque empirique du vote de majorité par $b_{\mathbb{S}}^{N}(\rho)$ est précise. Les résultats de Lacasse (2010) ont montré que le vote de majorité randomisé permet d'obtenir des garanties précises avec un risque empirique $\widehat{\mathsf{R}}_{\mathbb{S}}(\mathsf{MV}_\rho)$ faible. Cependant, ce n'est qu'une forme restreinte du vote de majorité MV_ρ originel. Pour obtenir des bornes encore plus précises pour MV_ρ, nous introduisons le *vote de majorité stochastique* : pour chaque entrée $\mathbf{x}$, un vote de majorité MV_ρ est obtenu en tirant les poids selon une autre distribution. Ce nouveau vote de majorité est présenté en détails dans la Section 6.4.1. Nous montrons également comment calculer l'approximation du risque, ainsi que sa valeur exacte. À partir du calcul exact, nous énonçons dans la Section 6.4.2 deux bornes PAC-Bayésiennes essentielles pour dériver des algorithmes auto-certifiés dans la Section 6.4.3.

6.4.1 Le vote de majorité stochastique

6.4.1.1 Définitions

Pour le vote de majorité stochastique, nous considérons que ses poids $\rho \in \mathbb{M}(\mathbb{H})$ sont tirés selon une distribution P appelée "*hyper-posterior*[4]" : on dit que P est un *hyper-posterior* sur $\mathbb{H}$. Grâce à cette notion d'*hyper-posterior*, le vote de majorité devient "stochastique", *i.e.*, pour chaque entrée $\mathbf{x} \in \mathbb{X}$ les poids ρ sont tirés selon l'*hyper-posterior* P puis la classe prédite est celle renvoyée par $\mathsf{MV}_\rho(\mathbf{x})$. Le principal avantage de considérer un vote de majorité stochastique est qu'il permet de dériver et d'optimiser directement des bornes en généralisation PAC-Bayésiennes. Les risques réel et empirique du vote de majorité stochastique, définis ci-dessous, prennent en considération les risques de MV_ρ avec ρ tiré selon P.

Définition 6.4.1 (Risques du vote de majorité stochastique)**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble de votants $\mathbb{H}$, pour tout *hyper-posterior* P sur $\mathbb{H}$, les risques réel et empirique du vote de majorité stochastique sont respectivement

$$\mathop{\mathbb{E}}_{\rho\sim\mathrm{P}} \mathsf{R}_{\mathcal{D}}(\mathsf{MV}_\rho) = \mathop{\mathbb{E}}_{\rho\sim\mathrm{P}} \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} \mathrm{I}\left[m_\rho(\mathbf{x},y) \leq 0\right],$$

$$\text{et} \quad \mathop{\mathbb{E}}_{\rho\sim\mathrm{P}} \widehat{\mathsf{R}}_{\mathbb{S}}(\mathsf{MV}_\rho) = \mathop{\mathbb{E}}_{\rho\sim\mathrm{P}} \frac{1}{m}\sum_{i=1}^{m} \mathrm{I}\left[m_\rho(\mathbf{x}_i,y_i) \leq 0\right].$$

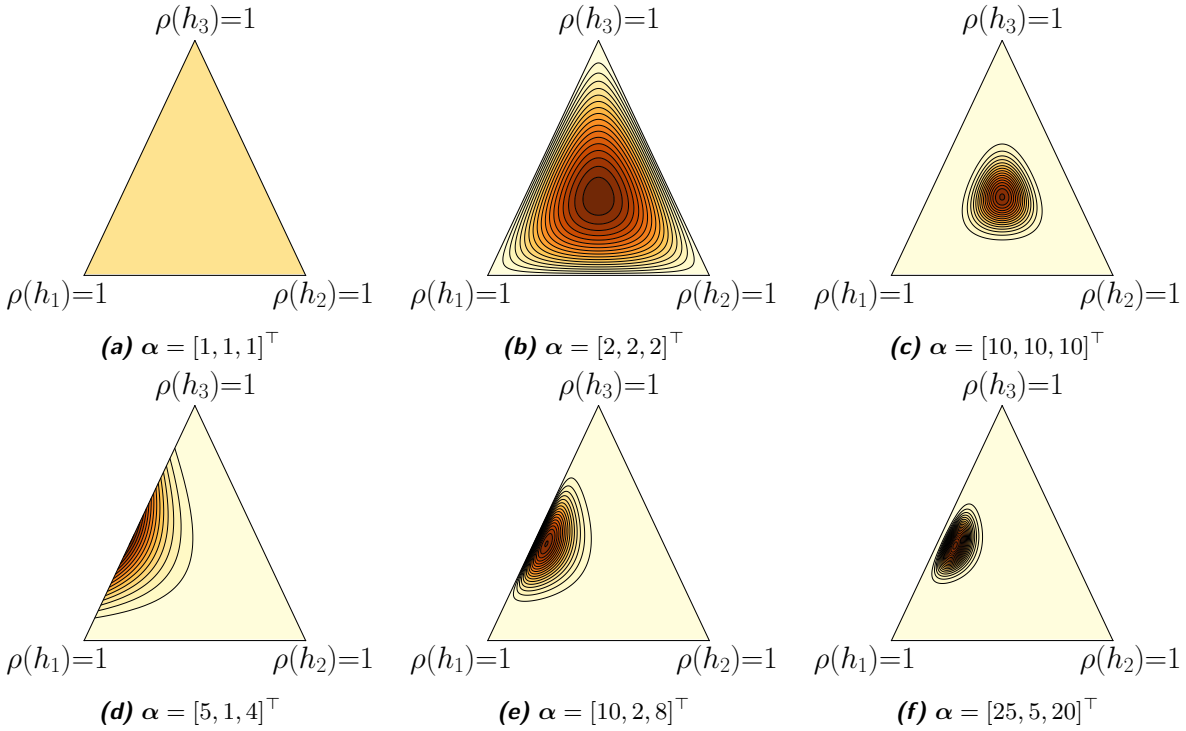
Nous utilisons la $\frac{1}{2}$-marge de Laviolette et al. (2017) pour majorer le risque du vote de majorité stochastique, on a

$$\mathop{\mathbb{E}}_{\rho\sim\mathrm{P}} \mathsf{R}_{\mathcal{D}}(\mathsf{MV}_\rho) \;\leq\; \mathop{\mathbb{E}}_{\rho\sim\mathrm{P}} \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} \mathrm{I}\left[\widehat{m_\rho}(\mathbf{x},y) \leq 0\right] \;=\; \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} s_{\mathrm{P}}(\mathbf{x},y), \tag{6.8}$$

$$\text{et} \quad \mathop{\mathbb{E}}_{\rho\sim\mathrm{P}} \widehat{\mathsf{R}}_{\mathbb{S}}(\mathsf{MV}_\rho) \;\leq\; \mathop{\mathbb{E}}_{\rho\sim\mathrm{P}} \frac{1}{m}\sum_{i=1}^{m} \mathrm{I}\left[\widehat{m_\rho}(\mathbf{x}_i,y_i) \leq 0\right] \;=\; \frac{1}{m}\sum_{i=1}^{m} s_{\mathrm{P}}(\mathbf{x}_i,y_i), \tag{6.9}$$

où $s_{\mathrm{P}}(\mathbf{x},y) = \mathbb{E}_{\rho\sim\mathrm{P}} \mathrm{I}\left[\widehat{m_\rho}(\mathbf{x},y) \leq 0\right]$ est le risque stochastique. Tout d'abord, rappelons qu'en classification binaire, l'inégalité devient une égalité (Section 2.4.2). Ensuite, comme nous le verrons dans la Section 6.4.1.3, $s_{\mathrm{P}}(\mathbf{x},y)$ permet d'obtenir une solution en forme close. Plus précisément, nous introduisons deux méthodes pour calculer ce risque : *(i)* soit en passant par une approximation (*e.g.*, avec une méthode de Monte Carlo), *(ii)* soit en calculant la forme close. Dans les deux cas, des hypothèses doivent être faites sur la distribution P. Lorsque la

4. La notion d'*hyper-posterior* a été introduite en PAC-Bayes par Pentina et Lampert (2014) pour le *life-long learning* pour permettre la considération de différents votes adaptés à différentes tâches.



(a) $\boldsymbol{\alpha} = [1, 1, 1]^\top$ **(b)** $\boldsymbol{\alpha} = [2, 2, 2]^\top$ **(c)** $\boldsymbol{\alpha} = [10, 10, 10]^\top$

(d) $\boldsymbol{\alpha} = [5, 1, 4]^\top$ **(e)** $\boldsymbol{\alpha} = [10, 2, 8]^\top$ **(f)** $\boldsymbol{\alpha} = [25, 5, 20]^\top$

Figure 6.3. *La fonction de densité de la distribution de Dirichlet pour plusieurs valeurs de $\boldsymbol{\alpha}$. Le "triangle" correspond au simplexe $\mathbb{\Delta}^2$ de dimension 2 des probabilités dont les angles correspondent aux distributions extrêmes. Un point dans le simplexe est la combinaison linéaire des distributions extrêmes.*

distribution ρ est discrète, elle se trouve dans le simplexe des probabilités noté $\mathbb{\Delta}^{(\mathrm{card}(\mathbb{H})-1)}$ (de dimension $\mathrm{card}(\mathbb{H})-1$). Un choix naturel pour l'*hyper-posterior* est alors la distribution de Dirichlet dont la fonction de densité est la suivante.

Définition 6.4.2 (Distribution de Dirichlet)**.** Soit $n = \mathrm{card}(\mathbb{H})$ le cardinal d'un ensemble fini d'hypothèses $\mathbb{H}$. Étant donné les paramètres de concentration $\boldsymbol{\alpha} \in (\mathbb{R}_*^+)^n$, la distribution de Dirichlet $\mathrm{Dir}(\boldsymbol{\alpha})$ est définie par

$$\rho \sim \mathrm{P} \iff (\rho(h_1), \ldots, \rho(h_n)) \sim \mathrm{Dir}(\boldsymbol{\alpha}),$$
$$\text{où} \quad \mathrm{P}(\rho) = \frac{1}{Z(\boldsymbol{\alpha})} \prod_{j=1}^{n} \Big[\rho(h_j)\Big]^{\alpha_j - 1} \propto \prod_{j=1}^{n} \Big[\rho(h_j)\Big]^{\alpha_j - 1}.$$

Quelques exemples de distribution de Dirichlet sont donnés sur la Figure 6.3. Notons que si $\boldsymbol{\alpha}$ est le vecteur dont tous les éléments sont à 1, alors la distribution est la distribution uniforme sur le simplexe $\mathbb{\Delta}^{(\mathrm{card}(\mathbb{H})-1)}$.

Sous l'hypothèse que l'*hyper-posterior* est exprimé comme une distribution de Dirichlet, nous proposons, dans la section suivante, un algorithme pour approximer le risque stochastique $s_{\mathrm{P}}(\mathbf{x}, y)$ qui fait partie intégrante de notre algorithme auto-certifié présenté dans la Section 6.4.3.

6.4.1.2 Approximation du risque stochastique

Nous proposons un algorithme de Monte Carlo (MC) pour calculer $s_{\mathrm{P}}(\mathbf{x}, y)$ fait pour accélérer l'optimisation. Pour cet algorithme, nous introduisons une relaxation du risque réel

pour pouvoir mettre à jour $\boldsymbol{\alpha}$ par descente de gradient. Pour ce faire, nous utilisons la fonction de perte sigmoïde tempérée *(tempered sigmoid loss)* $\mathrm{sig}_c(x) = \frac{1}{1+\exp(-cx)}$ avec $c \in \mathbb{R}^+$. Puisque cette fonction est une relaxation, l'Algorithme 6.4 suivant résout une relaxation du problème d'origine et n'est donc pas une forme exacte (Nesterov, 2005).

Algorithme 6.4 Approximation du risque stochastique

Entrées : Ensemble $\mathbb{S}$, distribution de Dirichlet $\mathrm{P} = \mathrm{Dir}(\boldsymbol{\alpha})$, nombre de tirages K

1: Tirage de $\{\rho_k\}_{k=1}^K \sim \mathrm{P}^K = \mathrm{Dir}(\boldsymbol{\alpha})^K$

2: **pour tout** exemple $(\mathbf{x}_i, y_i) \in \mathbb{S}$ **faire**

3: $$s_{\mathrm{P}}(\mathbf{x}_i, y_i) \approx \frac{1}{K} \sum_{k=1}^{K} \mathrm{sig}_c\left[-\widehat{m_{\rho_k}}(\mathbf{x}_i, y_i)\right]$$

4: **retourner** $\frac{1}{m}\sum_{i=1}^m s_{\mathrm{P}}(\mathbf{x}_i, y_i)$

Cet algorithme commence par tirer K votes de majorité et calcule une approximation du risque stochastique $s_{\mathrm{P}}(\mathbf{x}_i, y_i)$ pour chaque exemple $(\mathbf{x}_i, y_i) \in \mathbb{S}$. Un inconvénient de l'Algorithme 6.4 est qu'il requiert le tirage de K votes et qu'il calcule la prédiction pour tous les exemples de $\mathbb{S}$. Pour contourner ce problème, nous proposons une solution en forme clause dans la section suivante.

6.4.1.3 Calcul exact du risque stochastique

Toujours sous l'hypothèse que P est une distribution de Dirichlet, nous pouvons dériver la solution en forme clause du risque réel suivante.

Lemme 6.4.1 (Calcul du risque stochastique)**.** Étant donné $(\mathbf{x}, y) \in \mathbb{X}\times\mathbb{Y}$, soit

$$\mathbb{F}(\mathbf{x}, y) = \{j \ : \ h_j(\mathbf{x}) \neq y\}, \quad \text{et} \quad \mathbb{T}(\mathbf{x}, y) = \{j \ : \ h_j(\mathbf{x}) = y\},$$

respectivement l'ensemble des indices des votants qui classent incorrectement $(\mathbf{x}, y)$ et l'ensemble des indices des votants qui classent correctement $(\mathbf{x}, y)$. Alors, le risque stochastique $s_{\mathrm{P}}(\mathbf{x}, y)$ se réécrit

$$s_{\mathrm{P}}(\mathbf{x}, y) = \mathop{\mathbb{E}}_{\rho\sim\mathrm{P}} \mathrm{I}\left[\widehat{m_\rho}(\mathbf{x}, y) \leq 0\right] = I_{0.5}\left(\sum_{j\in\mathbb{T}(\mathbf{x},y)} \alpha_j, \sum_{j\in\mathbb{F}(\mathbf{x},y)} \alpha_j\right),$$

avec $I_{0.5}()$ la fonction beta incomplète régularisée évaluée en 0.5, définie par

$$I_{0.5}(a, b) = \frac{B_{0.5}(a, b)}{B_1(a, b)},$$

où $B_t(a, b) = \int_0^t x^{a-1}(1-x)^{b-1}dx$ est la fonction beta incomplète.

Le Lemme 6.4.1 nous dit que le risque stochastique pour un exemple $(\mathbf{x}, y)$ peut être calculé par une solution en forme close. En conséquence, nous pouvons calculer une borne supérieure sur le risque du vote de majorité stochastique basée sur le risque stochastique.

Corollaire 6.4.1 (Solution en forme close pour les risques stochastiques)**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble d'apprentissage $\mathbb{S}\sim\mathcal{D}^m$, pour tout ensemble fini d'hypothèses $\mathbb{H}$, pour toute distribution $\mathrm{P}=\mathrm{Dir}(\boldsymbol{\alpha})$ avec $\boldsymbol{\alpha}\in(\mathbb{R}_*^+)^{\mathrm{card}(\mathbb{H})}$, on a

$$\mathop{\mathbb{E}}_{\rho\sim\mathrm{P}} \mathsf{R}_{\mathcal{D}}(\mathsf{MV}_\rho) \;\leq\; \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} s_{\mathrm{P}}(\mathbf{x},y) \;=\; \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} I_{0.5}\left(\sum_{j\in\mathbb{T}(\mathbf{x},y)}\alpha_j\;,\;\sum_{j\in\mathbb{F}(\mathbf{x},y)}\alpha_j\right),$$

$$\text{et}\quad \mathop{\mathbb{E}}_{\rho\sim\mathrm{P}} \widehat{\mathsf{R}}_{\mathbb{S}}(\mathsf{MV}_\rho) \;\leq\; \frac{1}{m}\sum_{i=1}^m s_{\mathrm{P}}(\mathbf{x}_i,y_i) \;=\; \frac{1}{m}\sum_{i=1}^m I_{0.5}\left(\sum_{j\in\mathbb{T}(\mathbf{x}_i,y_i)}\alpha_j\;,\;\sum_{j\in\mathbb{F}(\mathbf{x}_i,y_i)}\alpha_j\right).$$

À partir de ce Corollaire, nous proposons de calculer directement le risque empirique stochastique avec l'Algorithme 6.5.

Algorithme 6.5 Calcul exact du risque du vote de majorité stochastique

Entrées : Ensemble $\mathbb{S}$, distribution de Dirichlet $\mathrm{P}=\mathrm{Dir}(\boldsymbol{\alpha})$

1: **pour tout** exemple $(\mathbf{x}_i,y_i)\in\mathbb{S}$ **faire**

2: $\quad s_{\mathrm{P}}(\mathbf{x}_i,y_i)=I_{0.5}\left(\sum_{j\in\mathbb{T}(\mathbf{x}_i,y_i)}\alpha_j,\sum_{j\in\mathbb{F}(\mathbf{x}_i,y_i)}\alpha_j\right)$

3: **retourner** $\frac{1}{m}\sum_{i=1}^m s_{\mathrm{P}}(\mathbf{x}_i,y_i)$

Grâce aux Algorithmes 6.4 et 6.5, nous pouvons calculer le risque empirique stochastique. Nous avons mené une étude empirique pour déterminer les conditions dans lesquelles chaque algorithme se révèle le plus efficace. En résumé, l'Algorithme 6.5 peut être plus rapide que l'Algorithme 6.4, en particulier lorsque le nombre de tirages K est élevé par rapport à m. Cependant, cet avantage est à nuancer : si m est suffisamment grand, l'Algorithme 6.4 atteint des performances similaires à celles de l'Algorithme 6.5, même avec $K=1$. Par ailleurs, l'augmentation de K permet d'atteindre les performances de l'Algorithme 6.5 tout en maintenant un coût inférieur pour des valeurs raisonnables de m et K.

Cette étape de calcul du risque stochastique est l'étape clé pour la dérivation, dans la Section 6.4.3, de nos algorithmes auto-certifiés de minimisation du risque du vote de majorité stochastique. En effet, les bornes en généralisation PAC-Bayésiennes, que nous dérivons dans la prochaine section, requièrent le calcul du risque stochastique afin d'être minimisée.

6.4.2 Bornes PAC-Bayésienne pour le vote stochastique

Pour obtenir une borne supérieure sur le risque réel du vote de majorité stochastique, nous suivons l'approche à la Seeger. En fait, nous énonçons deux bornes : la première pour laquelle les votants sont indépendants des données, la seconde où les votants dépendent des données.

6.4.2.1 Une borne PAC-Bayes classique

Dans cette section, nous supposons que nous avons une connaissance *a priori* sur les poids $\rho\in\mathbb{M}(\mathbb{H})$, *i.e.*, nous supposons une distribution *hyper-prior* Π sur l'ensemble $\mathbb{H}$. Nous pouvons

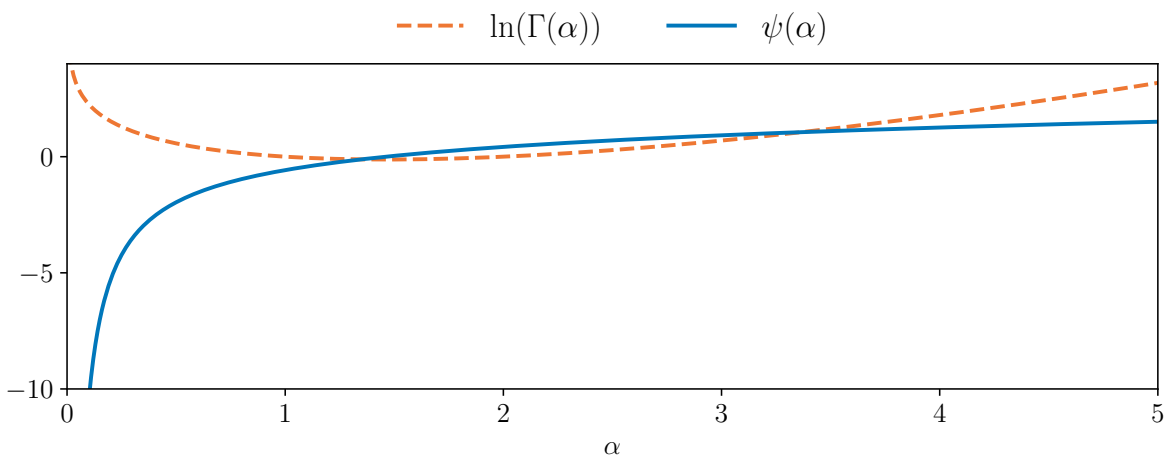


Figure 6.4. *Graphique de la fonction Digamma $\psi(\cdot)$ représentée par la courbe bleue continue, et de sa dérivée $\ln(\Gamma(\cdot))$ (i.e., le logarithme de la fonction Gamma $\Gamma(\cdot)$) représentée par la courbe orange en pointillés.*

montrer la borne suivante qui dépend de la KL-divergence $\mathrm{KL}(\mathrm{P}\|\Pi)$ entre l'*hyper-prior* Π et l'*hyper-posterior* P.

Théorème 6.4.1 (Borne PAC-Bayésienne pour le vote de majorité stochastique)**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble fini d'hypothèses $\mathbb{H}$, pour toute distribution *hyper-prior* $\Pi = \mathrm{Dir}(\boldsymbol{\beta})$ avec $\boldsymbol{\beta}\in(\mathbb{R}_*^+)^{\mathrm{card}(\mathbb{H})}$, pour tout $\delta\in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\begin{array}{l}\forall\,\mathrm{P}\text{ sur }\mathbb{H},\\ \mathop{\mathbb{E}}_{\rho\sim\mathrm{P}}\mathsf{R}_{\mathcal{D}}(\mathsf{MV}_\rho) \;\le\; \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} s_{\mathrm{P}}(\mathbf{x},y)\\ \qquad\qquad \le \overline{\mathrm{kl}}\left(\dfrac{1}{m}\displaystyle\sum_{i=1}^{m} s_{\mathrm{P}}(\mathbf{x}_i,y_i)\;\middle|\;\dfrac{\mathrm{KL}(\mathrm{P}\|\Pi)+\ln\frac{2\sqrt{m}}{\delta}}{m}\right)\end{array}\right]\ge 1-\delta, \tag{6.10}$$

$$\text{avec}\quad \mathrm{KL}(\mathrm{P}\|\Pi) = \sum_{j=1}^{\mathrm{card}(\mathbb{H})}\ln[\Gamma(\beta_j)] - \ln\left[\Gamma\left(\sum_{j=1}^{\mathrm{card}(\mathbb{H})}\beta_j\right)\right] - \sum_{j=1}^{\mathrm{card}(\mathbb{H})}\ln[\Gamma(\alpha_j)]$$
$$+\ln\left[\Gamma\left(\sum_{j=1}^{\mathrm{card}(\mathbb{H})}\alpha_j\right)\right] + \sum_{j=1}^{\mathrm{card}(\mathbb{H})}(\alpha_j-\beta_j)\left[\psi(\alpha_j)-\psi\left(\sum_{j=1}^{\mathrm{card}(\mathbb{H})}\alpha_j\right)\right],$$

où $\Gamma(\alpha)=\int_0^\infty x^{\alpha-1}e^{-x}dx$ est la fonction Gamma et la fonction Digamma $\Psi(\alpha)$ est définie comme la dérivée de $\ln[\Gamma(\alpha)]$ (voir la Figure 6.4).

Nous utilisons cette borne pour dériver dans la Section 6.4.3, un algorithme pour le vote de majorité stochastique. Bien que cette borne soit exprimée comme une borne à la Seeger, notre contribution n'est pas restreinte au choix de la forme des bornes. En effet, il est possible de démontrer des bornes à la McAllester ou à la Catoni.

Comme nous l'avons par exemple dans la Section 2.3.3, plus la KL-divergence $\mathrm{KL}(\mathrm{P}\|\Pi)$ est grande, plus les distributions P et Π sont différentes. Dans ce contexte, une augmentation des paramètres de Dirichlet $\boldsymbol{\alpha}$ peut impliquer une augmentation de la KL-divergence et une concentration de la distribution de Dirichlet (voir la Figure 6.3). En effet, si les paramètres $\boldsymbol{\alpha}$ augmentent, les risques du vote de majorité stochastique tendent vers le risque du vote de majorité où le poids pour un votant i est $\frac{\alpha_i}{\|\boldsymbol{\alpha}\|_1}$. Cependant, une augmentation des paramètres

favorise une KL-divergence grande : la borne est un compromis entre la concentration des risques et la KL-divergence.

6.4.2.2 Une borne PAC-Bayes avec des votants dépendants des données

Un inconvénient du Théorème 6.4.1 est qu'il suppose que les votants sont indépendants de l'ensemble d'apprentissage $\mathbb{S}$. Pour contourner ce problème, nous proposons une borne qui permet la prise en considération de votants dépendants des données.

Nous rappelons que le Théorème 6.4.1 est valide pour un *hyper-prior* Π et un ensemble de votants $\mathbb{H}$ définis *a priori*, *i.e.*, ils sont indépendants des données $\mathbb{S} \sim \mathcal{D}^m$. En PAC-Bayes, il est admis que considérer un *prior* dépendant des données peut amener à bornes plus précises (*e.g.*, Parrado-Hernández et al., 2012b ; Dziugaite et al., 2021). D'après les travaux de Thiemann et al. (2017) et Mhammedi et al. (2019) sur les bornes PAC-Bayésiennes permettant l'utilisation de *priors* dépendants des données, nous dérivons une borne en généralisation qui permet d'apprendre les votants à partir d'un ensemble de données additionnel. Plus précisément, nous considérons deux ensemble d'apprentissage indépendants $\mathbb{S}_1$ et $\mathbb{S}_2$ et nous apprenons un ensemble de votants avec chacun des ensembles notés $\mathbb{H}_1$ et $\mathbb{H}_2$. L'*hyper-prior* sur $\mathbb{H}_1$, *resp.* sur $\mathbb{H}_2$, est Π_1, *resp.* Π_2. De la même manière, nous définissons un *hyper-posterior* par ensemble de votants : P_1 et P_2. Le théorème suivant montre que nous pouvons borner le risque de deux votes de majorité stochastiques combinés, tant que leur risque empirique est évalué sur l'ensemble de données qui n'a pas été utilisé pour apprendre leurs votants respectifs. Nous présentons ici uniquement la borne dans sa forme simplifiée [5] qui correspond à la situation dans laquelle $\mathbb{S}_1$ et $\mathbb{S}_2$ ont la même taille (comme par exemple Mhammedi et al., 2019). En pratique, cela revient à séparer les données d'apprentissage en $50\%/50\%$ et est le découpage qui est en général associé aux meilleurs résultats empiriques.

Théorème 6.4.2 (Borne PAC-Bayésienne pour des votants dépendants des données)**.** Soit Π_1 et P_1 un *hyper-prior* et un *hyper-posterior* sur $\mathbb{H}_1$ définis à partir de $\mathbb{S}_1$, et Π_2 et P_2 un *prior* et un *posterior* sur $\mathbb{H}_2$ définis à partir de $\mathbb{S}_2$. Pour tout $\delta \in]0, 1]$, on a on a

$$\mathop{\mathbb{P}}_{\substack{\mathbb{S}_1\sim\mathcal{D}^{m_1}\\ \mathbb{S}_2\sim\mathcal{D}^{m_2}}}\left[\begin{array}{l} \dfrac{1}{2}\left(\mathop{\mathbb{E}}_{\rho\sim\mathrm{P}_1}\mathsf{R}_{\mathcal{D}}(\mathsf{MV}_\rho)+\mathop{\mathbb{E}}_{\rho'\sim\mathrm{P}_2}\mathsf{R}_{\mathcal{D}}(\mathsf{MV}_{\rho'})\right) \;\le\; \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}}\dfrac{1}{2}\Big(s_{\mathrm{P}_1}(\mathbf{x},y)+s_{\mathrm{P}_2}(\mathbf{x},y)\Big) \\ \le \overline{\mathrm{kl}}\left[\dfrac{1}{2}\left(\mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathbb{S}_1}s_{\mathrm{P}_1}(\mathbf{x},y)+\mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathbb{S}_2}s_{\mathrm{P}_2}(\mathbf{x},y)\right)\middle|\dfrac{\mathrm{KL}(\mathrm{P}_1\|\Pi_1)+\mathrm{KL}(\mathrm{P}_2\|\Pi_2)+2\ln\frac{4\sqrt{m}}{\delta}}{m}\right]\end{array}\right]\ge 1-\delta. \tag{6.11}$$

où $m = 2\lfloor\frac{m_1+m_2}{2}\rfloor$ et $\lfloor\cdot\rfloor$ est la fonction partie entière.

Comme pour le Théorème 6.4.1, le risque réel des deux votes de majorité stochastiques combinés est majoré par une borne PAC-Bayes qui dépend de deux termes : les deux risques empiriques des votes stochastiques et les deux KL-divergences entre les *hyper-priors* et les *hyper-posteriors*.

5. Voir Zantedeschi et al. (Th. 2 2021) pour la borne avec différentes tailles pour $\mathbb{S}_1$ et $\mathbb{S}_2$.

6.4.3 Algorithmes d'apprentissage pour le vote de majorité stochastique

Comme dans la Section 6.3, nous proposons deux algorithmes auto-certifiés, un premier pour minimiser l'Équation (6.10) et un second pour minimiser l'Équation (6.11). Nous suivons le principe de la descente de gradient stochastique avec des mini-lots $\mathbb{U} \subseteq \mathbb{S}$ pour optimiser les bornes. Plus précisément, la fonction objectif associée à l'Équation (6.10) est

$$F_{\mathbb{U}}(\mathrm{P}) = \overline{\mathrm{kl}}\left(\mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathbb{U}} s_{\mathrm{P}}(\mathbf{x},y) \;\middle|\; \frac{\mathrm{KL}(\mathrm{P}\|\Pi) + \ln\frac{2\sqrt{m}}{\delta}}{m} \right), \tag{6.12}$$

qui correspond à la borne supérieure appliquée sur le mini-lot $\mathbb{U}$. Pour optimiser cette fonction, nous appliquons l'Algorithme 6.6 (décrit ci-dessous) soit avec l'Algorithme 6.4 pour approximer les risques empiriques, soit avec l'Algorithme 6.5 pour calculer exactement les risques. À chaque itération, nous calculons le risque empirique sur le mini-lot $\mathbb{U}$. Puis, nous calculons la fonction objectif et nous mettons à jour l'*hyper-posterior* P avec un algorithme de descente de gradient.

Algorithme 6.6 Minimisation de l'Équation (6.10)

Entrées : Ensemble $\mathbb{S}$, distribution *hyper-prior* Π sur $\mathbb{H}$, nombre d'itérations T

1: $\mathrm{P} \leftarrow \Pi$
2: **pour** $t \leftarrow 1$ à T **faire**
3: **pour tout** mini-lot $\mathbb{U} \subseteq \mathbb{S}$ **faire**
4: Calcul du risque stochastique $\mathbb{E}_{(\mathbf{x},y)\sim\mathbb{U}} s_{\mathrm{P}}(\mathbf{x},y)$ avec l'Algorithme 6.4 ou 6.5
5: $\mathrm{P} \leftarrow$ Mise à jour de P avec $F_{\mathbb{U}}(\mathrm{P})$ par descente de gradient
6: **retourner** P

Dans le cas où les votants dépendent des données, le Théorème 6.4.2 utilise deux ensembles d'apprentissage disjoints $\mathbb{S}_1$ et $\mathbb{S}_2$; la fonction objectif associée à l'Équation (6.11) est alors

$$F_{\mathbb{U}_1,\mathbb{U}_2}(\mathrm{P}) = \overline{\mathrm{kl}}\left[\frac{1}{2}\left(\mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathbb{U}_1} s_{\mathrm{P}_1}(\mathbf{x},y) + \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathbb{U}_2} s_{\mathrm{P}_2}(\mathbf{x},y) \right) \;\middle|\; \frac{\mathrm{KL}(\mathrm{P}_1\|\Pi_1) + \mathrm{KL}(\mathrm{P}_2\|\Pi_2) + 2\ln\frac{4\sqrt{m}}{\delta}}{m} \right]. \tag{6.13}$$

Cette fonction est estimée avec les mini-lots $\mathbb{U}_1 \subseteq \mathbb{S}_1$ et $\mathbb{U}_2 \subseteq \mathbb{S}_2$. L'Algorithme 6.6 résume la procédure. À chaque itération, les deux risques stochastiques sont calculés avec l'Algorithme 6.4 ou 6.5. Ensuite, nous calculons la fonction objectif de l'Équation (6.13). Enfin, les *hyper-posteriors* P_1 et P_2 sont mis à jour par descente de gradient.

Algorithme 6.7 Minimisation de l'Équation (6.11)

Entrées : Ensembles $\mathbb{S}_1$ et $\mathbb{S}_2$, *hyper-priors* Π_1 sur $\mathbb{H}_1$ et Π_2 sur $\mathbb{H}_2$, nombre d'itérations T

1: $(\mathrm{P}_1, \mathrm{P}_2) \leftarrow (\Pi_1, \Pi_2)$
2: **pour** $t \leftarrow 1$ à T **faire**
3: **pour tout** mini-lot $\mathbb{U}_1 \subseteq \mathbb{S}_1$ et $\mathbb{U}_2 \subseteq \mathbb{S}_2$ **faire**
4: Calcul de $\mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathbb{U}_1} s_{\mathrm{P}_1}(\mathbf{x},y)$ et $\mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathbb{U}_2} s_{\mathrm{P}_2}(\mathbf{x},y)$ avec l'Algorithme 6.4 ou 6.5
5: $(\mathrm{P}_1, \mathrm{P}_2) \leftarrow$ Mise à jour de $(\mathrm{P}_1, \mathrm{P}_2)$ avec $F_{\mathbb{U}_1,\mathbb{U}_2}(\mathrm{P})$ par descente de gradient
6: **retourner** $(\mathrm{P}_1, \mathrm{P}_2)$

Remarque sur le calcul des dérivées. Lorsque le risque est calculé via l'Algorithme 6.4, la somme est différentiable. Cependant, puisque le risque est obtenu par un échantillonnage de Monte Carlo, nous utilisons la technique de reparamétrisation implicite (Figurnov et al., 2018 ; Jankowiak et Obermeyer, 2018) pour obtenir les dérivées (implémentées dans `PyTorch` (Paszke et al., 2019) par exemple). Lorsque le risque est calculé via la solution en forme close du Corollaire 6.4.1 avec l'Algorithme 6.5, les risques dépendent de la fonction $I_{0.5}()$ qui est différentiable (voir Boik et Robinson-Cox, 1999).

6.5 Résumé des expériences

6.5.1 Les C-bornes PAC-Bayésiennes

Les expériences conduites ont d'abord été réalisées pour les algorithmes auto-certifiés de minimisation des C-bornes PAC-Bayésiennes de la Section 6.3.2 dans Viallard et al. (2021a). Il en ressort que l'Algorithme 6.3 apparaît être le meilleur algorithme parmi nos trois algorithmes auto-certifiés minimisant une C-borne PAC-Bayésienne. En outre, l'Algorithme 6.1, basé sur la C-borne la plus interprétable, produit les moins bons résultats. Globalement, les valeurs de bornes associées à l'Algorithme 6.3 sont plus précises que celles des Algorithmes 6.1 et 6.2. Nous pensons que l'Algorithme 6.3 amène à des valeurs de bornes plus faibles que l'Algorithme 6.2 car l'approche de Lacasse et al. majore conjointement l'erreur jointe et le désaccord.

Nous avons également comparé nos algorithmes à deux algorithmes de la littérature qui minimisent la C-borne empirique (`MinCq` de Roy et al. (2011) et `CB-Boost` de Bauvin et al. (2020)) ainsi qu'aux algorithmes auto-certifiés de minimisation des différentes relaxations du risque du vote de majorité : la méthode proposée par Masegosa et al. (2020) pour $\mathrm{R}_{\mathcal{D}}(\mathrm{MV}_\rho) \leq 4\mathrm{e}_{\mathcal{D}}(\rho)$, et un algorithme similaire à l'Algorithme 6.2 pour $\mathrm{R}_{\mathcal{D}}(\mathrm{MV}_\rho) \leq 2\mathrm{R}_{\mathcal{D}}(G_\rho)$, mais sans le numérateur de la C-borne. Comparé aux approches de l'état de l'art, l'algorithme de minimisation du risque de Gibbs est associé aux bornes les plus faibles parmi tous les algorithmes, au détriment d'un grand risque en test. Ce comportement illustre clairement la limitation de considérer *uniquement* le risque de Gibbs comme estimateur du risque du vote de majorité. En effet, comme discuté dans la Section 2.4.2, le risque de Gibbs n'est pas assez précis pour estimer le vote de majorité puisqu'une augmentation de la diversité des votants peut avoir un impact négatif sur le risque de Gibbs. De plus, comparé à l'approche de Masegosa et al., les résultats de nos algorithmes sont comparables. Ce comportement était attendu puisque la minimisation de la borne de Masegosa et al. (2020) ou d'une C-borne PAC-Bayésienne revient à minimiser un compromis entre le risque et le désaccord. Enfin, pour la classification binaire, notre Algorithme 6.3 produit de meilleurs résultats que `CB-Boost` et que `MinCq` pour lequel la différence est significative et les bornes sont proches de 1 (*i.e.*, non informative). L'optimisation de bornes sur le risque a tendance à produire de meilleures garanties, justifiant que la minimisation d'une C-borne empirique est souvent trop optimiste (comme pour `CB-Boost` ou `MinCq`).

En résumer, ces expériences indiquent que notre Algorithme 6.3 est celui qui amène au meilleur compromis entre avoir de bonnes performances en termes d'optimisation du risque tout en assurant des garanties en généralisation (avec des bornes informatives).

6.5.2 Le vote de majorité stochastique

Nous avons ensuite réalisé dans Zantedeschi et al. (2021) une série d'expériences complémentaire pour évaluer nos Algorithmes 6.6 et 6.7. En plus des algorithmes cités ci-dessus, nous avons considéré l'algorithme proposé par Lacasse et al. (2010) de minimisation de l'Équation (6.7) qui dépend du vote de majorité randomisé où N votants sont tirés de ρ.

Nous avons principalement remarqué que les Algorithmes 6.6 et 6.7 obtiennent des performances similaires en termes de risque stochastique en test et de valeur de borne. En outre, nos algorithmes sont capables d'obtenir des risques en test moyennés similaires aux risques des autres méthodes déterministes, ce qui démontre l'intérêt du vote de majorité stochastique. Nous pensons que cela est du au fait que les risques stochastiques dépendent de la $\frac{1}{2}$-marge de Laviolette et al. (2017).

6.6 Conclusion

Ce chapitre présente des algorithmes auto-certifiés (Freund, 1998) dans le cadre de l'apprentissage supervisé pour le vote de majorité. Nous avons proposé, dans un premier temps, des algorithmes de minimisation du risque du vote de majorité via des bornes en généralisation PAC-Bayésiennes pour la C-borne. Ensuite, pour s'attaquer à un des inconvénients de ces bornes, qui sont basées sur une relaxation du risque du vote de majorité, nous avons proposé un nouveau type de vote de majorité : le vote de majorité stochastique qui, pour chaque entrée $\mathbf{x}$, tire un vote de majorité MV_ρ à partir d'une probabilité de distribution *hyper-posterior* P puis renvoie $\mathrm{MV}_\rho(\mathbf{x})$. Comme dans le cas de la C-borne, nous avons dérivé des algorithmes auto-certifiés pour optimiser directement le risque de ce vote de majorité stochastique. Notre évaluation empirique a confirmé la qualité des modèles appris avec nos algorithmes, ainsi que la précision des bornes par rapport à l'état de l'art pour l'apprentissage de vote de majorité PAC-Bayésien.

Théorie PAC-Bayésienne pour la robustesse adversaire

7

Contexte

Ce chapitre a été réalisé durant la thèse de Paul Viallard et est, à notre connaissance, la première analyse PAC-Bayésienne pour la robustesse adversaire : sous l'angle du défenseur, l'objectif est d'avoir des garanties sur la capacité du vote de majorité PAC-Bayésien à être robuste aux attaques adversaires (souvent malveillantes). Ces travaux ont donné lieu à une publication à NeurIPS (Viallard et al., 2021b). À noter, qu'une autre de mes contributions (non présentée dans ce manuscrit), publiée à ECML-PKDD (Patracone et al., 2024), est en lien avec ce chapitre. Les travaux associés se placent cette fois-ci du côté de l'attaquant et ont l'intérêt de proposer à la fois une extension d'une forme d'attaque classique (les attaques universelles) et des garanties théoriques sous la forme d'une borne en généralisation (basée sur la complexité de Rademacher) sur la capacité des attaques à attaquer un modèle sur de nouvelles données.

7.1 Introduction

Dans ce chapitre, nous commençons par formaliser la notion de vote de majorité dans le cadre de la robustesse adversaire en Section 7.2. Pour cela, nous proposons une adaptation du vote de majorité où les entrées peuvent être légèrement modifiées ou perturbées afin de tromper la prédiction du vote de majorité (souvent à des fins malveillantes). Ce genre d'entrée modifiée est connu sous le nom d'*exemple adversaire* (Biggio et al., 2013 ; Szegedy et al., 2014) et est illustré par la Figure 7.1.

Pour assurer la sécurité des utilisateurs, il est important que les modèles soient robustes à ce genre de faibles perturbations. En effet, lorsque les modèles sont appliqués à des tâches réelles, comme les véhicules autonomes, les perturbations ne doivent pas compromettre la sécurité des utilisateurs. Les exemples perturbés sont obtenus à partir d'une *attaque adversaire* qui trompe le modèle, tandis que les techniques de *défense adversaire* renforcent la robustesse adversaire pour rendre les attaques inutiles (*e.g.*, Goodfellow et al., 2015 ; Papernot et al., 2016 ; Carlini et Wagner, 2017 ; Kurakin et al., 2017 ; Zantedeschi et al., 2017 ; Madry et al., 2018 ; Patracone et al., 2024). Cependant, les votes de majorité, comme d'autres modèles, manquent de garanties sur leur capacité à être robuste.

Prédiction : "Chat" | Perturbation | Prédiction : "Cheval" (exemple adversaire)

Figure 7.1. *Sur la gauche, l'image originale est classée correctement en "cat". Au centre, l'image correspond à la perturbation/au bruit ajouté à l'image originale pour obtenir l'image de droite. Pour l'œil humain, cette dernière apparaît identique à l'originale, mais la prédiction change radicalement. C'est cette image, avec la perturbation imperceptible, qui est appelée exemple adversaire.*

Nous formulons la robustesse adversaire au travers du prisme de la théorie PAC-Bayésienne ; nous appelons ce cadre *adversarially robust PAC-Bayes*. L'idée est de considérer un *risque de robustesse adversaire moyen* correspondant à la probabilité que le modèle commette une erreur de classification sur un exemple perturbé (cela peut être interprété comme un risque moyen sur les perturbations). Nous définissons également un *risque adversaire moyen-max* comme la probabilité qu'il existe au moins une perturbation entraînant une erreur de classification. Ces définitions ont l'avantage *(i)* d'être compatibles avec le cadre PAC-Bayésien et les votes de majorité, et *(ii)* d'être liées au risque de robustesse adversaire classique. Pour nos deux risques, nous dérivons une borne en généralisation PAC-Bayésienne valide pour tout type d'attaque. D'un point de vue algorithmique, ces bornes sont directement optimisables pour apprendre un vote de majorité robuste en moyenne sur les attaques. Nos algorithmes sont donc auto-certifiés (Freund, 1998). Nous illustrons empiriquement que notre cadre est capable de conduire à des garanties précises sur le risque adversaire tout en assurant une protection efficace face aux attaques adversaires.

7.2 Vote de majorité robuste aux attaques adversaires

7.2.1 Notations et contexte

Nous suivons principalement le cadre de la Section 2.4 pour des taches de classification binaire où $\mathbb{X} \subseteq \mathbb{R}^{\mathrm{d}}$ est l'espace d'entrée et $\mathbb{Y} = \{-1, +1\}$ l'espace de sortie. Soit $\mathcal{D}$ une distribution sur $\mathbb{X}\times\mathbb{Y}$ fixée, mais inconnue. Un exemple est noté par $(\mathbf{x}, y) \in \mathbb{X}\times\mathbb{Y}$. Soit $\mathbb{S} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{m}$ l'ensemble d'apprentissage composé de m exemples *i.i.d.* selon $\mathcal{D}$. Soit $\mathbb{H}$ un ensemble de votants à valeurs réelles de $\mathbb{X}$ vers $[-1, +1]$. Étant donné $\mathbb{H}$ et l'ensemble d'apprentissage $\mathbb{S}$, notre objectif est d'apprendre un vote de majorité PAC-Bayésien (Définition 2.4.1) défini par

$$\forall \mathbf{x} \in \mathbb{X}, \quad \mathsf{MV}_\rho(\mathbf{x}) = \operatorname{sign}\left(\mathop{\mathbb{E}}_{h\sim\rho} h(\mathbf{x})\right).$$

Nous souhaitons trouver un vote de majorité qui minimise le risque réel $\mathsf{R}_{\mathcal{D}}(\mathsf{MV}_\rho)$ sur $\mathcal{D}$ (Définition 2.4.2) défini par

$$\mathsf{R}_{\mathcal{D}}(\mathsf{MV}_\rho) = \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} \mathrm{I}\left[\mathsf{MV}_\rho(\mathbf{x}) \neq y\right].$$

Cependant, pour des applications réelles, une perturbation imperceptible de l'entrée peut avoir une mauvaise influence sur les performances en classification de nouvelles données (Szegedy et al., 2014) : les bornes en généralisation classiques ne sont alors plus valables. Ce type de perturbation peut être modélisé par un bruit additif (relativement faible), noté ϵ, appliqué à une entrée $\mathbf{x}$ et amenant à une entrée perturbée $\mathbf{x}+\epsilon$. Soit $b>0$ et $\|\cdot\|$ une norme [1] arbitraire, l'ensemble des bruits possibles $\mathbb{B}$ est défini par

$$\mathbb{B}=\Big\{\epsilon \in \mathbb{R}^{\mathrm{d}} \;\Big|\; \|\epsilon\| \leq b\Big\}.$$

L'apprenant a pour but de trouver un modèle aux attaques adversaires qui soit robuste en moyenne à tous les bruits de $\mathbb{B}$ sur $(\mathbf{x},y)\sim\mathcal{D}$. Plus formellement, le but est de minimiser le risque adversaire réel $\mathsf{A}_{\mathcal{D}}(\mathsf{MV}_\rho)$ défini comme suit.

Définition 7.2.1 (Risque adversaire)**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour toute distribution ρ sur $\mathbb{H}$, le *risque adversaire réel* est défini par

$$\mathsf{A}_{\mathcal{D}}(\mathsf{MV}_\rho) = \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} \max_{\epsilon\in\mathbb{B}} \mathrm{I}\left[\mathsf{MV}_\rho(\mathbf{x}+\epsilon) \neq y\right].$$

Le *risque adversaire empirique* associé et calculé avec $\mathbb{S}\sim\mathcal{D}^m$ est

$$\widehat{\mathsf{A}}_{\mathbb{S}}(\mathsf{MV}_\rho) = \frac{1}{m}\sum_{i=1}^{m} \max_{\epsilon\in\mathbb{B}} \mathrm{I}\left[\mathsf{MV}_\rho(\mathbf{x}_i+\epsilon) \neq y_i\right].$$

Dans ce chapitre, l'objectif est de rendre le vote de majorité MV_ρ robuste aux *attaques adversaires* qui cherchent un *exemple adversaire* $\mathbf{x}+\epsilon^*(\mathbf{x},y)$ pour tromper MV_ρ sur un exemple donné $(\mathbf{x},y)$, où $\epsilon^*(\mathbf{x},y)$ est

$$\epsilon^*(\mathbf{x},y) \;\in\; \mathop{\mathrm{argmax}}_{\epsilon\in\mathbb{B}} \mathrm{I}\left[\mathsf{MV}_\rho(\mathbf{x}+\epsilon) \neq y\right]. \tag{7.1}$$

En conséquence, les mécanismes de *défense adversaire* s'appuient souvent sur les attaques adversaires en remplaçant, lors de l'apprentissage, les exemples originaux $(\mathbf{x},y)$ par des exemples adversaires $(\mathbf{x}+\epsilon^*(\mathbf{x},y),y)$. Cette procédure est appelée apprentissage adversaire. Même s'il existe d'autres méthodes de défense, comme nous le verrons plus tard, l'apprentissage adversaire semble être l'un des mécanismes de défense les plus efficaces (Ren et al., 2020). Cependant, optimiser l'Équation (7.1) n'est pas possible à cause de la non-convexité de MV_ρ induite par la fonction $\mathrm{sign}()$. Dans la littérature, les méthodes d'attaques adversaires cherchent la perturbation optimale $\epsilon^*(\mathbf{x},y)$, mais, en pratique, c'est une approximation de cette perturbation qui est considérée.

7.2.2 Travaux connexes

Défenses/attaques adversaires. [2] De nombreuses méthodes existent pour résoudre ou approximer l'optimisation de l'Équation (7.1). Parmi celles-ci, *Fast Gradient Sign Method* (fgsm, Goodfellow et al., 2015) est une attaque qui génère un bruit ϵ dans la direction du gradient de la fonction perte par rapport à l'entrée $\mathbf{x}$. Kurakin et al. (2017) ont introduit ifgsm, une version itérative de fgsm où à chaque itération, on répète fgsm et ajoute

1. Les normes les plus utilisées sont les normes ℓ_1, ℓ_2 et ℓ_∞.
2. Voir Ren et al. (2020) pour un survey sur les défenses/attaques adversaires.

à $\mathbf{x}$ un bruit qui est le signe du gradient de la perte par rapport à $\mathbf{x}$. En suivant le même principe que IFGSM, MADRY et al. (2018) ont proposé une méthode basée sur le *Projected Gradient Descent* (PGD) incluant une initialisation aléatoire de $\mathbf{x}$ avant l'optimisation. Une autre technique appelée *Carlini and Wagner Attack* (CARLINI et WAGNER, 2017) a pour but de trouver des exemples adversaires $\mathbf{x}+\epsilon^*(\mathbf{x}, y)$ qui soient aussi proches que possible de l'entrée $\mathbf{x}$, *i.e.*, le but est d'avoir une attaque la plus imperceptible possible. En pratique, produire ce genre de perturbation entraîne un temps de calcul élevé. Contrairement aux méthodes les plus populaires qui cherchent un modèle avec un risque adversaire faible, notre contribution s'inscrit dans une autre ligne de recherche où l'idée est de relâcher cette mesure de risque en "pire cas" en considérant un risque adversaire *moyen* sur les bruits, plutôt qu'une formulation basée sur le $\max$ (*e.g.*, ZANTEDESCHI et al., 2017 ; HENDRYCKS et DIETTERICH, 2019). Notre formulation "en moyenne" est introduite dans la Section 7.2.3.

Bornes en généralisation pour la robustesse adversaire Avant la contribution de ce chapitre, quelques bornes en généralisation pour la robustesse adversaire ont été introduites (*e.g.*, KHIM et LOH, 2018 ; COHEN et al., 2019 ; MONTASSER et al., 2019 ; PINOT et al., 2019 ; SALMAN et al., 2019 ; YIN et al., 2019 ; MONTASSER et al., 2020 ; PINOT et al., 2022). Les résultats de KHIM et LOH, et YIN et al. sont des bornes basées sur la complexité de Rademacher. Le premier utilise une approximation du risque adversaire, le deuxième introduit des bornes dans le cas particulier des réseaux de neurones et des classifieurs linéaires et implique une dépendance polynomiale inévitable par rapport à la dimension de l'entrée. MONTASSER et al. étudient l'apprentissage PAC robuste pour les classes PAC-apprenables avec une dimension VC finie pour les votes de majorité non pondérés "robustifiés" à l'aide d'un algorithme de *boosting*. Cependant, leur algorithme considère toutes les perturbations adversaires possibles pour chaque exemple, ce qui n'est pas calculable en pratique. En outre, leur borne implique une constante élevée comme indiquée à la fin de la preuve du Th. 3.1 de MONTASSER et al. (2019). COHEN et al. ont démontré des bornes qui estiment le bruit minimum nécessaire pour obtenir un exemple adversaire (dans le cas de perturbations définies comme du bruit gaussien), alors que nos résultats donnent la probabilité que le modèle soit trompé par un exemple adversaire. SALMAN et al. exploitent la méthode de COHEN et al. ainsi que l'apprentissage adversaire pour obtenir des bornes plus précises. De plus, FARNIA et al. présentent des bornes sur le risque adversaire basées sur la marge pour des réseaux de neurones et attaques spécifiques (comme FGSM ou PGD). Bien qu'ils fassent appel à une borne PAC-Bayésienne classique, leur résultat n'est pas une analyse PAC-Bayésienne et s'inscrit dans la famille des bornes en convergence uniforme (pour les détails voir Ap. J NAGARAJAN et KOLTER, 2019b).

7.2.3 Le risque adversaire PAC-Bayésien

Au lieu de chercher le bruit de l'Équation (7.1) qui maximise la chance de tromper l'algorithme, nous modélisons la perturbation via une distribution dépendante de l'exemple. Cette distribution est utilisée pour définir nos nouvelles mesures de risque. Définissons d'abord $\mathcal{B}_{(x,y)}$, une distribution sur l'ensemble des bruits possibles $\mathbb{B}$, qui dépend d'un exemple $(\mathbf{x}, y)\in\mathbb{X}\times\mathbb{Y}$. Nous notons $\mathcal{E}$ une distribution sur $(\mathbb{X}\times\mathbb{Y})\times\mathbb{B}$ définie comme

$$\mathcal{E}((\mathbf{x}, y), \epsilon) = \mathcal{D}(\mathbf{x}, y) \cdot \mathcal{B}_{(x,y)}(\epsilon),$$

qui permet de générer les *exemples perturbés*. Étant donné un exemple $(\mathbf{x}_i, y_i)\sim\mathcal{D}$, nous définissons l'ensemble $\mathfrak{C}_i=\{\epsilon_j^i\}_{j=1}^n$ de n perturbations tirées selon $\mathcal{B}_{(x_i,y_i)}$. Nous considérons comme ensemble d'apprentissage l'ensemble $\widehat{\mathbb{S}}=\{((\mathbf{x}_i, y_i), \mathfrak{C}_i)\}_{i=1}^m\in(\mathbb{X}\times\mathbb{Y}\times\mathbb{B}^n)^m$ (de taille

$m\times n$). En d'autres termes, chaque $((\mathbf{x}_i, y_i), \mathfrak{C}_i) \in \widehat{\mathbb{S}}$ est tiré selon une distribution que nous notons $\mathcal{E}^n$ telle que

$$\mathcal{E}^n((\mathbf{x}_i, y_i), \mathfrak{C}_i) = \mathcal{D}(\mathbf{x}_i, y_i) \cdot \prod_{j=1}^{n} \mathcal{B}_{(x_i,y_i)}(\epsilon_j^i).$$

De plus, nous notons $(\mathcal{E}^n)^m$ la distribution empirique sur l'ensemble d'apprentissage perturbé constitué de m exemples et n perturbations pour chaque exemple. Inspirés par Zantedeschi et al. (2017) et Hendrycks et Dietterich (2019), nous définissons maintenant le *risque adversaire moyen (robustness averaged adversarial risk)*.

Définition 7.2.2 (Risque adversaire moyen)**.** Pour toute distribution $\mathcal{E}$ sur $(\mathbb{X}\times\mathbb{Y})\times\mathbb{B}$, pour toute distribution ρ sur $\mathbb{H}$, le *risque adversaire moyen réel* de MV_ρ est

$$\mathsf{R}_{\mathcal{E}}(\mathsf{MV}_\rho) \;=\; \mathop{\mathbb{P}}_{((\mathbf{x},y),\epsilon)\sim\mathcal{E}} (\mathsf{MV}_\rho(\mathbf{x}+\epsilon) \neq y) \;=\; \mathop{\mathbb{E}}_{((\mathbf{x},y),\epsilon)\sim\mathcal{E}} \mathrm{I}\left[\mathsf{MV}_\rho(\mathbf{x}+\epsilon) \neq y\right].$$

Le *risque adversaire moyen empirique*, sur l'ensemble $\widehat{\mathbb{S}} = \{((\mathbf{x}_i, y_i), \mathfrak{C}_i)\}_{i=1}^m$, est

$$\widehat{\mathsf{R}}_{\widehat{\mathbb{S}}}(\mathsf{MV}_\rho) = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} \mathrm{I}\left[\mathsf{MV}_\rho(\mathbf{x}_i + \epsilon_j^i) \neq y_i\right].$$

Comme nous le verrons dans la Proposition 7.3.1, le risque $\mathsf{R}_{\mathcal{E}}(\mathsf{MV}_\rho)$ est vu comme un risque optimisé au regard de $\epsilon^*(\mathbf{x}, y)$ de l'Équation (7.1). En effet, au lieu d'être choisi comme maximisant la perte, ϵ est tiré selon une distribution. Cela peut conduire à un risque non informatif si ϵ n'est pas assez informatif pour tromper le modèle. Pour contourner ce problème, nous proposons une extension de ce risque que nous appelons le *risque adversaire moyen-max*.

Définition 7.2.3 (Risque adversaire moyen-max)**.** Pour toute distribution $\mathcal{E}$ sur $(\mathbb{X}\times\mathbb{Y})\times\mathbb{B}$, pour toute distribution ρ sur $\mathbb{H}$, le *risque adversaire moyen-max réel* de MV_ρ est

$$\mathsf{A}_{\mathcal{E}^n}(\mathsf{MV}_\rho) \;=\; \mathop{\mathbb{P}}_{((\mathbf{x},y),\mathfrak{C})\sim\mathcal{E}^n} \Big(\exists\, \epsilon \in \mathfrak{C},\, \mathsf{MV}_\rho(\mathbf{x}+\epsilon) \neq y\Big).$$

Le risque *adversaire moyen-max empirique*, sur l'ensemble $\widehat{\mathbb{S}} = \{((\mathbf{x}_i, y_i), \mathfrak{C}_i)\}_{i=1}^m$, est

$$\widehat{\mathsf{A}}_{\widehat{\mathbb{S}}}(\mathsf{MV}_\rho) = \frac{1}{m} \sum_{i=1}^{m} \max_{\epsilon\in\mathfrak{C}_i} \mathrm{I}\left[\mathsf{MV}_\rho(\mathbf{x}_i + \epsilon) \neq y_i\right].$$

Pour un exemple $(\mathbf{x}, y)\sim\mathcal{D}$, au lieu de vérifier si un exemple perturbé $\mathbf{x}+\epsilon$ est adversaire, nous tirons n exemples perturbés $\mathbf{x}+\epsilon_1, \ldots, \mathbf{x}+\epsilon_n$ et nous vérifions si au moins un des exemples est adversaire.

7.3 PAC-Bayes robuste aux attaques adversaires

7.3.1 Relations entre les risques adversaires

La Proposition 7.3.1 ci-dessous montre les relations intrinsèques entre le risque adversaire classique $\mathsf{A}_{\mathcal{D}}(\mathsf{MV}_\rho)$ et nos deux relaxations $\mathsf{R}_{\mathcal{E}}(\mathsf{MV}_\rho)$ et $\mathsf{A}_{\mathcal{E}^n}(\mathsf{MV}_\rho)$. En particulier, nous mon-

trons que plus le nombre d'exemples perturbés n est grand, plus la probabilité d'obtenir un exemple adversaire est élevée, donc plus on se rapproche du risque adversaire $\mathrm{A}_{\mathcal{D}}(\mathrm{MV}_\rho)$.

Proposition 7.3.1 (Relations entre les risques adversaires)**.** Pour toute distribution $\mathcal{E}$ sur $(\mathbb{X}\times\mathbb{Y})\times\mathbb{B}$, pour toute distribution ρ sur $\mathbb{H}$, pour tout $(n, n')\in\mathbb{N}^2$ avec $1\leq n'\leq n$, on a

$$\mathrm{R}_{\mathcal{E}}(\mathrm{MV}_\rho) \;\leq\; \mathrm{A}_{\mathcal{E}^{n'}}(\mathrm{MV}_\rho) \;\leq\; \mathrm{A}_{\mathcal{E}^{n}}(\mathrm{MV}_\rho) \;\leq\; \mathrm{A}_{\mathcal{D}}(\mathrm{MV}_\rho). \tag{7.2}$$

La partie gauche de l'Équation (7.2) confirme que le risque adversaire moyen $\mathrm{R}_{\mathcal{E}}(\mathrm{MV}_\rho)$ est optimiste par rapport au risque adversaire classique $\mathrm{A}_{\mathcal{D}}(\mathrm{MV}_\rho)$. La Proposition 7.3.2 estime à quel point $\mathrm{R}_{\mathcal{E}}(\mathrm{MV}_\rho)$ peut être proche de $\mathrm{A}_{\mathcal{D}}(\mathrm{MV}_\rho)$.

Proposition 7.3.2 (Risque classique et risque adversaire moyen)**.** Pour toute distribution $\mathcal{E}$ sur $(\mathbb{X}\times\mathbb{Y})\times\mathbb{B}$, pour toute distribution ρ sur $\mathbb{H}$, on a

$$\mathrm{A}_{\mathcal{D}}(\mathrm{MV}_\rho) - \mathrm{TV}(\gamma\|\Gamma) \;\leq\; \mathrm{R}_{\mathcal{E}}(\mathrm{MV}_\rho),$$

où Γ et γ sont des distributions sur $\mathbb{X}\times\mathbb{Y}$ et $\mathrm{TV}(\gamma\|\Gamma) = \mathbb{E}_{(\mathbf{x}',y')\sim\Gamma}\,\frac{1}{2}\left|\frac{\gamma(\mathbf{x}',y')}{\Gamma(\mathbf{x}',y')}-1\right|$, est la distance en variation totale *(Total Variation, TV)* entre γ et Γ.
La densité $\Gamma(\mathbf{x}', y')$ est la probabilité de tirer un exemple perturbé $(\mathbf{x}', y') = (\mathbf{x}+\epsilon, y)$ avec $((\mathbf{x}, y), \epsilon)\sim\mathcal{E}$, *i.e.*, on a

$$\Gamma(\mathbf{x}', y') \;=\; \Pr_{((\mathbf{x},y),\epsilon)\sim\mathcal{E}}\left[\mathbf{x}+\epsilon = \mathbf{x}' \;,\;\; y = y'\right].$$

La densité $\gamma(\mathbf{x}', y')$ est la probabilité de tirer un exemple adversaire $(\mathbf{x}', y') = (\mathbf{x}+\epsilon^*(\mathbf{x}, y), y)$ avec $(\mathbf{x}, y)\sim\mathcal{D}$, *i.e.*, on a

$$\gamma(\mathbf{x}', y') \;=\; \Pr_{(\mathbf{x},y)\sim\mathcal{D}}\left[\mathbf{x}+\epsilon^*(\mathbf{x}, y) = \mathbf{x}' \;,\;\; y = y'\right].$$

Notons que $\epsilon^*(\mathbf{x}, y)$ dépend de ρ, car γ dépend de ρ. D'après la Proposition 7.3.2, avec les distributions Γ et γ, les risques $\mathrm{A}_{\mathcal{D}}(\mathrm{MV}_\rho)$ et $\mathrm{R}_{\mathcal{E}}(\mathrm{MV}_\rho)$ peuvent être réécrits comme

$$\mathrm{R}_{\mathcal{E}}(\mathrm{MV}_\rho) = \Pr_{(\mathbf{x}',y')\sim\Gamma}\left[\mathrm{MV}_\rho(\mathbf{x}') \neq y'\right], \quad \text{et} \quad \mathrm{A}_{\mathcal{D}}(\mathrm{MV}_\rho) = \Pr_{(\mathbf{x}',y')\sim\gamma}\left[\mathrm{MV}_\rho(\mathbf{x}') \neq y'\right].$$

Enfin, les Propositions 7.3.1 et 7.3.2 lient le risque adversaire $\mathrm{R}_{\mathcal{E}}(\mathrm{MV}_\rho)$ au risque adversaire "standard" $\mathrm{A}_{\mathcal{D}}(\mathrm{MV}_\rho)$. En effet, des deux propositions, on obtient

$$\mathrm{A}_{\mathcal{D}}(\mathrm{MV}_\rho) - \mathrm{TV}(\gamma\|\Gamma) \;\leq\; \mathrm{R}_{\mathcal{E}}(\mathrm{MV}_\rho) \;\leq\; \mathrm{A}_{\mathcal{E}^n}(\mathrm{MV}_\rho) \;\leq\; \mathrm{A}_{\mathcal{D}}(\mathrm{MV}_\rho). \tag{7.3}$$

Ainsi, plus la distance $\mathrm{TV}(\gamma\|\Gamma)$ est petite, plus le risque adversaire moyen $\mathrm{R}_{\mathcal{E}}(\mathrm{MV}_\rho)$ est proche de $\mathrm{A}_{\mathcal{D}}(\mathrm{MV}_\rho)$ et plus il est probable qu'un exemple $((\mathbf{x}, y), \epsilon) \sim \mathcal{E}$ soit adversaire, *i.e.*, lorsque notre exemple adversaire "moyen" ressemble à un exemple adversaire "spécifique". De plus, l'Équation (7.3) justifie que le point de vue PAC-Bayésien a du sens pour l'apprentissage adversaire avec des garanties théoriques : les garanties PAC-Bayésiennes que nous dérivons dans la section suivante impliquent des garanties sur le risque adversaire $\mathrm{A}_{\mathcal{D}}(\mathrm{MV}_\rho)$.

7.3.2 Bornes PAC-Bayes pour le vote de majorité robuste

Pour dériver des bornes en généralisation PAC-Bayésiennes pour $\mathrm{R}_{\mathcal{E}}(\mathrm{MV}_\rho)$ et $\mathrm{A}_{\mathcal{E}^n}(\mathrm{MV}_\rho)$, nous considérons une des relaxations classiques, le risque de Gibbs (Définition 2.4.5) qui est défini pour nos risques adversaires dans les Équations (7.4) et (7.5).

Définition 7.3.1 (Risques de Gibbs adversaires)**.** Pour toute distribution $\mathcal{E}$ sur $(\mathbb{X}\times\mathbb{Y})\times\mathbb{B}$, pour tout ensemble d'hypothèses $\mathbb{H}$, pour toute distribution ρ sur $\mathbb{H}$, on a

$$\mathrm{R}_{\mathcal{E}}(G_\rho) = \mathop{\mathbb{E}}_{((\mathbf{x},y),\epsilon)\sim\mathcal{E}} \frac{1}{2}\left[1 - \mathop{\mathbb{E}}_{h\sim\rho} yh(\mathbf{x}+\epsilon)\right], \tag{7.4}$$

$$\text{et}\quad \mathrm{A}_{\mathcal{E}^n}(G_\rho) = \mathop{\mathbb{E}}_{((\mathbf{x},y),\mathfrak{C})\sim\mathcal{E}^n} \frac{1}{2}\left[1 - \min_{\epsilon\in\mathfrak{C}}\left(y \mathop{\mathbb{E}}_{h\sim\rho} h(\mathbf{x}+\epsilon)\right)\right]. \tag{7.5}$$

Ces relaxations sont exprimées comme l'espérance sur ρ des risques individuels des votants de $\mathbb{H}$. D'un point de vue algorithmique, $\mathrm{R}_{\mathcal{E}}(G_\rho)$ et $\mathrm{A}_{\mathcal{E}^n}(G_\rho)$ présentent les avantages *(i)* d'être différentiables, contrairement à $\mathrm{R}_{\mathcal{E}}(\mathrm{MV}_\rho)$ et $\mathrm{A}_{\mathcal{E}^n}(\mathrm{MV}_\rho)$, et *(ii)* de majorer $\mathrm{R}_{\mathcal{E}}(\mathrm{MV}_\rho)$ et $\mathrm{A}_{\mathcal{E}^n}(\mathrm{MV}_\rho)$ comme suit.

Théorème 7.3.1 (Relaxations du risque adversaire)**.** Pour toute distribution $\mathcal{E}$ sur $(\mathbb{X}\times\mathbb{Y})\times\mathbb{B}$ et ρ sur $\mathbb{H}$, pour tout $n>1$, on a

$$\mathrm{R}_{\mathcal{E}}(\mathrm{MV}_\rho) \le 2\,\mathrm{R}_{\mathcal{E}}(G_\rho), \qquad \text{et} \qquad \mathrm{A}_{\mathcal{E}^n}(\mathrm{MV}_\rho) \le 2\,\mathrm{A}_{\mathcal{E}^n}(G_\rho).$$

Une borne en généralisation pour $\mathrm{R}_{\mathcal{E}}(G_\rho)$, respectivement pour $\mathrm{A}_{\mathcal{E}^n}(G_\rho)$, implique donc une borne pour $\mathrm{R}_{\mathcal{E}}(\mathrm{MV}_\rho)$, respectivement pour $\mathrm{A}_{\mathcal{E}^n}(\mathrm{MV}_\rho)$. Les Théorèmes 7.3.2 and 7.3.3 énoncent nos bornes en généralisation PAC-Bayésiennes pour respectivement $\mathrm{R}_{\mathcal{E}}(G_\rho)$ et $\mathrm{A}_{\mathcal{E}^n}(G_\rho)$.

Théorème 7.3.2 (Borne PAC-Bayésienne pour $\mathrm{R}_{\mathcal{E}}(G_\rho)$)**.** Pour toute distribution $\mathcal{E}$ sur $(\mathbb{X}\times\mathbb{Y})\times\mathbb{B}$, pour tout ensemble de votants $\mathbb{H}$, pour toute distribution *prior* $\pi\in\mathbb{M}^*(\mathbb{H})$, pour tout $n\in\mathbb{N}_*$, pour tout $\delta\in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\widehat{\mathbb{S}}\sim(\mathcal{E}^n)^m}\left[\forall\rho\in\mathbb{M}(\mathbb{H}),\ \mathrm{kl}(\widehat{\mathrm{R}}_{\widehat{\mathbb{S}}}(G_\rho)\|\mathrm{R}_{\mathcal{E}}(G_\rho)) \le \frac{1}{m}\left(\mathrm{KL}(\rho\|\pi)+\ln\frac{m+1}{\delta}\right)\right] \ge 1-\delta, \tag{7.6}$$

$$\text{et}\ \mathop{\mathbb{P}}_{\widehat{\mathbb{S}}\sim(\mathcal{E}^n)^m}\left[\forall\rho\in\mathbb{M}(\mathbb{H}),\ \mathrm{R}_{\mathcal{E}}(G_\rho) \le \widehat{\mathrm{R}}_{\widehat{\mathbb{S}}}(G_\rho)+\sqrt{\frac{1}{2m}\left(\mathrm{KL}(\rho\|\pi)+\ln\frac{m+1}{\delta}\right)}\right] \ge 1-\delta, \tag{7.7}$$

$$\text{où}\ \ \widehat{\mathrm{R}}_{\widehat{\mathbb{S}}}(G_\rho) = \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}\frac{1}{2}\left[1-y_i \mathop{\mathbb{E}}_{h\sim\rho} h(\mathbf{x}_i+\epsilon_j^i)\right].$$

Il est important de préciser que le risque empirique associé à $\mathrm{R}_{\mathcal{E}}(G_\rho)$ est calculé avec $\widehat{\mathbb{S}}$ qui contient des exemples non *i.i.d.*, ce qui signifie que l'application de la technique de preuve "classique" ne peut s'appliquer. L'astuce est d'utiliser un résultat de Ralaivola et al. (2010) appelé *chromatic PAC-Bayesian bound* qui permet de considérer des données non indépendantes. Les bornes du Théorème 7.3.2 ne dépendent pas du nombre d'exemples

perturbés n, mais dépendent uniquement du nombre d'exemples originaux m. Ce comportement étonnant provient du faire que les n exemples perturbés sont inter-dépendants. Notons que l'Équation (7.6) a la forme d'une borne à la Seeger (2002) et est plus précise, mais moins interprétable que l'Équation (7.7) à la McAllester (1998). Ces bornes impliquent un compromis classique entre le risque empirique $\widehat{\mathsf{R}}_{\widehat{\mathbb{S}}}(G_\rho)$ et $\mathrm{KL}(\rho\|\pi)$.

Nous présentons maintenant une borne en généralisation pour $\mathsf{A}_{\mathcal{E}^n}(G_\rho)$. Puisque ce risque implique un *min*, nous ne pouvons pas utiliser la même astuce que pour le résultat précédent. Pour contourner ce problème, nous considérons la distance TV entre deux distributions artificielles sur $\mathfrak{C}_i$: une distribution arbitraire Θ_i sur $\mathfrak{C}_i$ pour tout $((\mathbf{x}_i,y_i),\mathfrak{C}_i)\in\widehat{\mathbb{S}}$ et une distribution de Dirac θ_i^h sur $\mathfrak{C}_i$ pour tout $h\in\mathbb{H}$ telle que $\theta_i^h(\epsilon)=1$ si $\epsilon=\operatorname{argmax}_{\epsilon\in\mathfrak{C}_i}\frac{1}{2}\big[1-y_ih(\mathbf{x}_i+\epsilon)\big]$ (*i.e.*, si ϵ maximise la perte linéaire) et 0 sinon.

Théorème 7.3.3 (Borne PAC-Bayésienne pour $\mathsf{A}_{\mathcal{E}^n}(G_\rho)$)**.** Pour toute distribution $\mathcal{E}$ sur $(\mathcal{X}\times\mathcal{Y})\times\mathbb{B}$, pour tout ensemble de votants $\mathbb{H}$, pour toute distribution $\pi\in\mathbb{M}^*(\mathbb{H})$, pour tout $n\in\mathbb{N}_*$, pour tout $\delta\in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\widehat{\mathbb{S}}\sim(\mathcal{E}^n)^m}\left[\begin{array}{l}\forall\rho\in\mathbb{M}(\mathbb{H}),\ \forall i\in\{1,\ldots,m\},\ \forall\Theta_i\text{ sur }\mathfrak{C}_i\text{ indépendante de }h\in\mathbb{H},\\ \mathsf{A}_{\mathcal{E}^n}(G_\rho)\ \le\ \dfrac{1}{m}\mathop{\mathbb{E}}_{h\sim\rho}\sum_{i=1}^m\max_{\epsilon\in\mathfrak{C}_i}\left[\dfrac{1-y_ih(\mathbf{x}_i+\epsilon)}{2}\right]+\sqrt{\dfrac{\mathrm{KL}(\rho\|\pi)+\ln\frac{2\sqrt{m}}{\delta}}{2m}}\\ \qquad\qquad\le\ \widehat{\mathsf{A}}_{\widehat{\mathbb{S}}}(G_\rho)+\dfrac{1}{m}\sum_{i=1}^m\mathop{\mathbb{E}}_{h\sim\rho}\mathrm{TV}(\theta_i^h\|\Theta_i)+\sqrt{\dfrac{\mathrm{KL}(\rho\|\pi)+\ln\frac{2\sqrt{m}}{\delta}}{2m}}\end{array}\right]\ge1-\delta,\tag{7.8}$$

avec la distance TV $\mathrm{TV}(\theta\|\Theta)=\mathbb{E}_{\epsilon\sim\Theta}\frac{1}{2}\left|\left[\frac{\theta(\epsilon)}{\Theta(\epsilon)}\right]-1\right|$

et le risque empirique $\widehat{\mathsf{A}}_{\widehat{\mathbb{S}}}(G_\rho)=\frac{1}{m}\sum_{i=1}^m\frac{1}{2}\Big[1-\min_{\epsilon\in\mathfrak{C}_i}\Big(y_i\mathop{\mathbb{E}}_{h\sim\rho}h(\mathbf{x}_i+\epsilon)\Big)\Big]$.

Pour minimiser le risque moyen-max réel $\mathsf{A}_{\mathcal{E}^n}(G_\rho)$ via l'Équation (7.8, 1ère borne), nous devons minimiser le compromis entre le risque empirique $\frac{1}{m}\,\mathbb{E}_h\sum_{i=1}^m\max_\epsilon\left[\frac{1}{2}\left(1-y_ih(\mathbf{x}_i+\epsilon)\right)\right]$ et $\mathrm{KL}(\rho\|\pi)$. Cependant, pour le calcul du risque empirique, la perte pour tous les votants et pour toutes les perturbations doit être calculée, ce qui peut être coûteux en temps. Nous proposons une alternative, avec l'Équation (7.8, 2ème borne), efficacement optimisable en utilisant $\frac{1}{m}\sum_{i=1}^m\mathbb{E}_h\,\mathrm{TV}(\theta_i^h\|\Theta_i)$ et le risque empirique moyen-max $\widehat{\mathsf{A}}_{\widehat{\mathbb{S}}}(G_\rho)$. Intuitivement, nous interprétons l'Équation (7.8, 2ème borne) comme un compromis entre le risque empirique (qui reflète de la robustesse) et de deux termes de pénalisation (KL et TV). La divergence $\mathrm{KL}(\rho\|\pi)$ contrôle à quel point la distribution *posterior* ρ diffère du *prior* π, tandis que la distance $\mathbb{E}_h\,\mathrm{TV}(\theta_i^h\|\Theta_i)$ contrôle la diversité des votants, *i.e.*, la capacité des votants d'être trompés par un même exemple adversaire. D'un point de vue algorithmique, un comportement intéressant est que la borne de l'Équation (7.8, 2ème borne) est valide pour toutes les distributions Θ_i sur $\mathfrak{C}_i$. Étant donné $(\mathbf{x}_i,y_i)$, cela suggère que nous cherchons Θ_i qui minimise $\mathbb{E}_h\,\mathrm{TV}(\theta_i^h\|\Theta_i)$. Idéalement, ce terme tend vers 0 lorsque Θ_i est proche[3] de θ_i^h et que tous les votants voient leur perte maximisée par la même perturbation $\epsilon\in\mathfrak{C}_i$.

3. Comme θ_i^h est une distribution de Dirac, on a $\mathbb{E}_h\,\mathrm{TV}(\theta_i^h\|\Theta_i)=\frac{1}{2}\left[1-\mathbb{E}_h\,\Theta_i(\epsilon_h^*)+\mathbb{E}_h\sum_{\epsilon\neq\epsilon_h^*}\Theta_i(\epsilon)\right]$, avec $\epsilon_h^*=\operatorname{argmax}_{\epsilon\in\mathfrak{C}_i}\frac{1}{2}\big[1-y_ih(\mathbf{x}_i+\epsilon)\big]$.

Pour apprendre un vote de majorité performant, une solution est alors de minimiser la partie droite de ces bornes pour trouver un bon compromis entre un risque empirique faible et une divergence faible entre les poids *prior* et les poids *posterior* appris. Cependant, telles quelles, les Équations (7.6) et (7.8, 1ère borne) ne sont pas pertinentes pour l'optimisation. L'Équation (7.6) n'est pas directement optimisable, car elle majore la $\mathrm{kl}()$ entre le risque réel et le risque empirique. Pour obtenir une borne optimisable, nous pouvons utiliser la fonction $\overline{\mathrm{kl}}()$ introduite dans la Définition 2.4.9. De plus, d'un point de vue algorithmique, le prior π est fixé et ne peut pas dépendre de l'ensemble d'apprentissage $\mathbb{S}$. Pour contourner ce problème, nous utilisons une borne de l'union en considérant T distributions *a priori* qui peuvent être sélectionnées *a posteriori* avec $\mathbb{S}$. Nous obtenons les deux bornes suivantes.

Corollaire 7.3.1 (Borne PAC-Bayésienne pour $\mathsf{R}_{\mathcal{E}}(G_\rho)$)**.** Pour toute distribution $\mathcal{E}$ sur $(\mathbb{X}\times\mathbb{Y})\times\mathbb{B}$, pour tout ensemble de votants $\mathbb{H}$, pour tout $T\in\mathbb{N}_*$, pour tout ensemble de distributions *prior* $\{\pi_1,\ldots,\pi_T\}\in\mathbb{M}^*(\mathbb{H})^T$, pour tout $n\in\mathbb{N}_*$, pour tout $\delta\in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\widehat{\mathbb{S}}\sim(\mathcal{E}^n)^m}\left[\forall\rho\in\mathbb{M}(\mathbb{H}),\ \mathsf{R}_{\mathcal{E}}(G_\rho)\leq\overline{\mathrm{kl}}\left(\widehat{\mathsf{R}}_{\widehat{\mathbb{S}}}(G_\rho)\middle|\frac{1}{m}\left[\mathrm{KL}(\rho\|\pi)+\ln\frac{T(m+1)}{\delta}\right]\right)\right]\geq 1-\delta. \quad (7.9)$$

Corollaire 7.3.2 (Borne PAC-Bayésienne pour $\mathsf{A}_{\mathcal{E}^n}(G_\rho)$)**.** Pour toute distribution $\mathcal{E}$ sur $(\mathbb{X}\times\mathbb{Y})\times\mathbb{B}$, pour tout ensemble de votants $\mathbb{H}$, pour tout $n\in\mathbb{N}_*$, pour toute distribution *prior* $\pi\in\mathbb{M}^*(\mathbb{H})$, pour tout $\delta\in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\widehat{\mathbb{S}}\sim(\mathcal{E}^n)^m}\left[\begin{array}{l}\forall\rho\in\mathbb{M}(\mathbb{H}),\ \forall i\in\{1,\ldots,m\},\ \forall\Theta_i \text{ sur } \mathfrak{C}_i \text{ indépendante de } h\in\mathbb{H},\\ \mathsf{A}_{\mathcal{E}^n}(G_\rho)\leq\widehat{\mathsf{A}}_{\widehat{\mathbb{S}}}(G_\rho)+\dfrac{1}{m}\displaystyle\sum_{i=1}^{m}\mathop{\mathbb{E}}_{h\sim\rho}\mathrm{TV}(\theta_i^h\|\Theta_i)+\sqrt{\dfrac{\mathrm{KL}(\rho\|\pi)+\ln\frac{2T\sqrt{m}}{\delta}}{2m}}\end{array}\right]\geq 1-\delta. \quad (7.10)$$

7.3.3 Des bornes à un algorithme

Nous dérivons maintenant un algorithme auto-certifié (Freund, 1998) d'apprentissage qui minimise soit la borne de l'Équation (7.9) soit celle de l'Équation (7.10). Soit un ensemble fini de votants $\mathbb{H}$ qui sont différentiables et où chaque $h\in\mathbb{H}$ est paramétrisé par un vecteur de poids $\mathbf{w}^h$. Inspiré par Masegosa et al., 2020, les votants de $\mathbb{H}$ et la distribution *prior* π dépendante des données sont appris avec un ensemble d'apprentissage $\mathbb{S}'$ indépendant de $\mathbb{S}$. Cette approche est classique en PAC-Bayes (Parrado-Hernández et al., 2012b ; Lever et al., 2013 ; Dziugaite et Roy, 2018 ; Dziugaite et al., 2021). Ensuite, la distribution *posterior* est apprise avec l'ensemble $\mathbb{S}$ en minimisant les bornes des Corollaires 7.3.1 and 7.3.2. Plus spécifiquement, nous minimisons une fonction objectif approximée par mini-lots $\mathbb{U}\subseteq\mathbb{S}$. La fonction objectif pour optimiser l'Équation (7.9), *resp.* Équation (7.10), est

$$F_{\mathbb{U}}(\rho)=\overline{\mathrm{kl}}\left(\widehat{\mathsf{R}}_{\mathbb{U}}(\rho)\middle|\frac{1}{m}\left[\mathrm{KL}(\rho\|\pi)+\ln\frac{T(m+1)}{\delta}\right]\right),$$

resp. $$F_{\mathbb{U}}(\rho)=\widehat{\mathsf{A}}_{\mathbb{U}}(\rho)+\sqrt{\frac{1}{2m}\left[\mathrm{KL}(\rho\|\pi)+\ln\frac{2T\sqrt{m}}{\delta}\right]}.$$

La distance TV n'apparaît pas dans la fonction objectif puisque nous faisons le choix de fixer $n=1$, *i.e.*, nous tirons un seul bruit par exemple. En effet, si $n=1$, la valeur de la distance TV est 0 (avec $n>1$ nous devrions la minimiser).

Nous décrivons maintenant notre algorithme d'apprentissage adversaire en deux étapes (résumé dans l'Algorithme 7.1). La première étape construit un ensemble de votants $\mathbb{H}$ et la distribution *prior* π associée. La seconde est dédiée à l'apprentissage de ρ. Ces étapes sont détaillées ci-après.

Algorithme 7.1 Algorithme d'apprentissage adversaire moyen avec garantie

Entrées : Ensemble d'apprentissage disjoints $\mathbb{S}$ et $\mathbb{S}'$, *prior* initial π_0 sur $\mathbb{H}_0$ (avec $\mathbf{w}_0$), fonction objectif $F()$, nombre d'*epochs* T et T', fonction d'attaque

Étape 1 – Construction du *prior* et de l'ensemble des votants

pour $t \leftarrow 1$ to T' **faire**
  $\pi_t \leftarrow \pi_{t-1}$ et $\mathbb{H}_t \leftarrow \mathbb{H}_{t-1}$ ($\mathbf{w}_t \leftarrow \mathbf{w}_{t-1}$)
  pour tout mini-lot $\mathbb{U} \subseteq \mathbb{S}'$ **faire**
    $\mathbb{U} \leftarrow$ Attaque de MV_{π_t} avec $(\mathbf{x}, y)$ de $\mathbb{U}$
    Mise à jour de π_t avec $\nabla_{\pi_t} \widehat{\mathsf{R}}_{\mathbb{U}}(\pi_t)$
    Mise à jour de $\mathbf{w}_t$ avec $\nabla_{\mathbf{w}_t} \widehat{\mathsf{R}}_{\mathbb{U}}(\pi_t)$
  $\mathbb{S}_t \leftarrow$ Attaque de MV_{π_t} avec les exemples de $\mathbb{S}$
$(\pi, \mathbb{H}) \leftarrow (\pi_{t^*}, \mathbb{H}_{t^*})$ avec $t^* \leftarrow \operatorname{argmin}_{t' \in \{1,\ldots,t\}} \widehat{\mathsf{R}}_{\mathbb{S}_{t'}}(\pi_{t'})$

Étape 2 – Minimisation de la borne

$\rho_0 \leftarrow \pi$
pour $t \leftarrow 1$ à T **faire**
  $\rho_t \leftarrow \rho_{t-1}$
  pour tout mini-lot $\mathbb{U} \subseteq \mathbb{S}$ **faire**
    $\mathbb{U} \leftarrow$ Attaque de MV_{π} avec les exemples $(\mathbf{x}, y)$ de $\mathbb{U}$
    Mise à jour de ρ_t avec $\nabla_{\rho_t} F_{\mathbb{U}}(\rho_t)$
  $\mathbb{S}_t \leftarrow$ Attaque de MV_{π} avec les exemples de $\mathbb{S}$
$\rho \leftarrow \rho_{t^*}$ avec $t^* \leftarrow \operatorname{argmin}_{t' \in \{1,\ldots,t\}} F_{\mathbb{S}_{t'}}(\rho_{t'})$

Pour attaquer les exemples. Les attaques qui interviennent dans l'Algorithme 7.1 diffèrent des attaques qui génèrent l'ensemble perturbé $\widehat{\mathbb{S}}$. À chaque itération (des deux étapes), nous attaquons un exemple avec le modèle courant alors que $\widehat{\mathbb{S}}$ est généré avec le vote de majorité *prior* MV_{π} (*i.e.*, la sortie de l'Étape 1).

Étape 1. À partir d'une distribution *prior* initiale π_0 (*e.g.*, la distribution uniforme) et d'un ensemble initial de votants $\mathbb{H}_0$ où chaque votant h est paramétré par un vecteur de poids $\mathbf{w}_0^h$, l'objectif est de construire l'ensemble d'hypothèses $\mathbb{H}$ et la distribution *prior* π à fournir en entrée de l'Étape 2 pour minimiser la borne. Pour ce faire, à chaque *epoch* t de l'Étape 1, nous apprenons à partir de $\mathbb{S}'$ un *prior* "intermédiaire" π_t sur un ensemble de votants "intermédiaire" $\mathbb{H}_t$ constitué de votants h paramétrés par des poids $\mathbf{w}_t^h$; l'optimisation est réalisée par rapport à $\mathbf{w}_t = \{\mathbf{w}_t^h\}_{h \in \mathbb{H}_t}$. À chaque itération de l'optimiseur, pour chaque $(\mathbf{x}, y)$ du mini-lot courant $\mathbb{U}$, nous attaquons le vote de majorité MV_{π_t} pour obtenir un exemple perturbé $\mathbf{x}+\epsilon$. Ensuite, nous effectuons une *forward pass* dans le vote de majorité

avec les exemples perturbés et mettons à jour les poids $\mathbf{w}_t$ et le *prior* π_t. Pour résumer, à la fin de l'Étape 1, le *prior* π et l'ensemble de votants $\mathbb{H}$ construit pour l'Étape 2 sont ceux associés à la meilleure *epoch* $t^* \in \{1, \ldots, T'\}$ qui permet de minimiser $\widehat{\mathrm{R}}_{\mathbb{S}_t}(\mathrm{MV}_{\pi_t})$ où $\mathbb{S}_t$ est l'ensemble perturbé obtenu en attaquant MV_{π_t} avec les exemples de $\mathbb{S}$. Cette sélection du *prior* π avec $\mathbb{S}$, bien que pouvant ressembler à une forme de triche, est une stratégie valide puisque les Équations (7.10) et (7.9) sont valides pour tout *prior* $\pi \in \{\pi_1, \ldots, \pi_{T'}\}$.

Étape 2. À partir du *prior* π sur $\mathbb{H}$ et de l'ensemble $\mathbb{S}$, nous réalisons la même procédure que pour l'Étape 1 à la différence près que la fonction objectif correspond à la borne que nous souhaitons optimiser, notée $F()$. Notons que les distributions *prior* "intermédiaires" ne dépendent pas de $\mathbb{S}$ puisqu'elles sont apprises avec $\mathbb{S}'$: les bornes sont donc valides.

7.4 Résumé des expériences

Dans l'article VIALLARD et al. (2021b), nous avons illustré empiriquement que notre cadre PAC-Bayésien pour la robustesse adversaire est capable de fournir des garanties en généralisation précises pour le risque adversaire. Pour cela, nous avons considéré les attaques PGD et IFGSM avec les normes ℓ_2 et ℓ_∞, ainsi que leurs variantes spécifiques au PAC-Bayes notées PGD_U et $\mathrm{IFGSM}_\mathrm{U}$. Nous avons étudié différents scénarios de défense/attaque pour différentes tâches (les votants étant ici des arbres de décisions différentiables). Nous avons étudié deux situations pour $\mathbb{H}$: l'ensemble $\mathbb{H}$ tel qu'obtenu à l'Étape 1 et l'ensemble $\mathbb{H}^{\mathrm{SIGN}} = \{h'(\cdot) = \mathrm{sign}(h(\cdot)) \mid h \in \mathbb{H}\}$ (en supprimant la fonction $\mathrm{sign}()$ lors de l'attaque pour être différentiable par rapport à l'entrée).

Il s'est avéré que $\mathbb{H}^{\mathrm{SIGN}}$ permet d'obtenir les meilleures bornes et résultats. D'une part, les bornes obtenues avec $\mathbb{H}^{\mathrm{SIGN}}$ sont toutes informatives (*i.e.*, plus petite que 1) et apportent des garanties pour les modèles (ce qui n'est pas le cas pour toutes les bornes avec $\mathbb{H}$ dont les bornes associées à l'Équation (7.8, $2^{\text{ème}}$ borne) qui dépassent 1 alors que les risques sont comparables aux risques obtenus avec $\mathbb{H}^{\mathrm{SIGN}}$. En fait, lors de l'optimisation, considérer $\mathbb{H}^{\mathrm{SIGN}}$ aide au contrôle de la distance TV. Les performances obtenues avec $\mathbb{H}^{\mathrm{SIGN}}$ peuvent être expliquées par le fait que la fonction signe "sature" la sortie des votants, rendant le vote de majorité plus robuste aux bruits.

Comme attendu, les bornes obtenues associées au Théorème 7.3.2 sont plus précises que celles du Théorème 7.3.3. En effet, nous avons montré que le risque adversaire moyen-max $\mathrm{A}_{\mathcal{E}^n}(\mathrm{MV}_\rho)$ est moins optimiste que le risque adversaire moyen associé $\mathrm{R}_{\mathcal{E}}(\mathrm{MV}_\rho)$. De plus, les valeurs de la borne de l'Équation (7.8, $1^{\text{ère}}$ borne) sont plus précises que celles de l'Équation (7.8, $2^{\text{ème}}$ borne), ce qui était également attendu puis l'Équation (7.8, $1^{\text{ère}}$ borne) est une borne inférieure de l'Équation (7.8, $2^{\text{ème}}$ borne).

Un point intéressant est que nos bornes apportent des garanties à la fois pour nos risques $\mathrm{R}_{\mathcal{E}}(\mathrm{MV}_\rho)$ et $\mathrm{A}_{\mathcal{E}^n}(\mathrm{MV}_\rho)$, mais également pour le risque adversaire classique $\mathrm{A}_{\mathcal{D}}(\mathrm{MV}_\rho)$. En effet, malgré le pessimisme du risque classique, ce dernier reste cohérent avec nos bornes, *i.e.*, il est plus faible que les bornes. En outre, les écarts observés entre le risque classique et nos risques sont faibles, ce qui signifie que notre relaxation "moyennée" n'est pas trop optimiste.

Enfin, nos résultats nous ont permis de confirmer que nous sommes capables d'apprendre des modèles robustes face aux attaques testées avec des garanties théoriques.

7.5 Conclusion

Ce travail est, à notre connaissance, le premier qui étudie et formalise la robustesse adversaire dans le cadre PAC-Bayésien pour le vote de majorité pondéré. À partir de cette formalisation, nous avons pu dériver des bornes en généralisation sur le risque adversaire du vote de majorité basé sur deux mesures de risques adversaire moyennées. Ces bornes *(i)* sont suffisamment générales pour être valides pour tout type d'attaque adversaire, *(ii)* sont précises, et *(iii)* peuvent être directement minimisée via un algorithme auto-certifié pour obtenir un vote de majorité robuste à des perturbations imperceptibles des données d'entrée. Une des limitations de ce travail est qu'il est principalement théorique et ne se concentre que sur le vote de majorité, sans nécessairement s'intéresser à obtenir les meilleures performances.

Quatrième partie

De bornes PAC-Bayésiennes désintégrées à de nouvelles bornes

Un cadre général pour la désintégration des bornes PAC-Bayésienne

8

Contexte

Les chapitres précédents se placent tous dans le cadre "classique" des bornes PAC-Bayésiennes où les garanties en généralisation portent sur une espérance sur l'ensemble des hypothèses $\mathbb{H}$. Ce chapitre introduit, quant à lui, une nouvelle approche de désintégration des bornes PAC-Bayésiennes pour obtenir des bornes qui s'appliquent à une hypothèse de $\mathbb{H}$. L'intérêt de nos travaux est que les bornes obtenues sont facilement optimisables et amènent à des algorithmes auto-certifiés. Ces travaux, réalisés durant la thèse de Paul Viallard, ont donné lieu à une publication dans le journal *Machine Learning Journal* (Viallard et al., 2024b) également présenté durant la conférence ECML-PKDD 2024.

8.1 Introduction

Comme nous l'avons vu dans ce manuscrit, la théorie PAC-Bayésienne est un outil puissant pour majorer le risque réel de modèles stochastiques tels que le vote de majorité stochastique considéré dans le Chapitre 6. Cependant, la grande majorité des méthodes d'apprentissage requièrent des garanties pour des modèles déterministes. Dans ce cas, une étape de "dérandomisation" doit être appliquée à une borne PAC-Bayésienne "stochastique" pour pouvoir s'appliquer à un modèle déterministe. Plusieurs approches de dérandomisation existent pour des cadres spécifiques. Par exemple, Langford et Shawe-Taylor (2002) ont proposé une dérandomisation pour des distributions *posterior* gaussiennes sur les classifieurs linéaires : la symétrie gaussienne permet d'obtenir une borne sur le risque du classifieur *maximum a posteriori* (déterministe) à partir de la borne sur le risque (stochastique) de Gibbs. En s'appuyant également sur des *posteriors* gaussiens, Letarte et al. (2019a) ont dérivé une borne PAC-Bayésienne pour une architecture de réseau déterministe très spécifique utilisant des fonctions de signe comme activations (cette approche a été étendue par Biggs et Guedj (2021, 2022)). Une autre ligne de travaux consiste à dérandomiser les réseaux de neurones (Neyshabur et al., 2018 ; Nagarajan et Kolter, 2019b). Bien que techniquement différente, cette approche part de garanties PAC-Bayésiennes sur le risque de Gibbs et utilise une borne de "perturbation de sortie" pour convertir une borne "stochastique" en

une borne pour le classifieur moyen. Le fait que ces travaux existants soient spécifiques et relativement diverses illustre le manque de cadre général pour la dérandomisation des bornes PAC-Bayésiennes classiques.

Ce chapitre se focalise sur un autre type de dérandomisation qui s'effectue via une *désintégration des bornes PAC-Bayes*, proposé par Catoni (2007, Th.1.2.7) et Blanchard et Fleuret (2007) (rappelées dans la Section 2.5). Malgré l'intérêt de ces résultats pour la dérandomisation, ces bornes ont été peu étudiées dans la littérature et n'ont jamais été utilisées en pratique. Motivés par des objectifs pratiques, nous dérivons de nouvelles bornes PAC-Bayésiennes désintégrées précises et utilisables *(i)* qui dérandomisent tout classifieur sans étape additionnelle et avec *presque* aucun impact sur les garanties, et *(ii)* qui peuvent être facilement optimisables pour apprendre des classifieurs avec des garanties. Pour ce faire, nous proposons un cadre général de désintégration basée sur la divergence de Rényi qui permet d'atteindre l'objectif pratique d'un apprentissage efficace. D'un point de vue théorique, le terme de divergence de Rényi fait que notre résultat est censé être moins précis que celui de Rivasplata et al. (2020, Th.1*(i)*) dans lequel le terme de divergence est "désintégré" mais dépend uniquement de l'hypothèse tirée. Cependant, comme nous l'avons montré lors de notre évaluation empirique sur des réseaux de neurones, leur terme "désintégré" est, en pratique, sujet à une forte variance, rendant leur borne plus difficile à optimiser. En effet, dans notre cas, notre terme de divergence de Rényi n'est pas influencé par l'hypothèse tirée. Notre borne a donc l'avantage de conduire à un algorithme plus stable avec de meilleurs résultats empiriques.

8.2 Cadre et bornes PAC-Bayésiennes

Nous nous plaçons dans le cadre de la classification supervisée décrite dans la Section 2.2 avec $\mathbb{X}$ l'espace d'entrée, $\mathbb{Y}$ l'espace de sortie et $\mathcal{D}$ une distribution inconnue sur $\mathbb{X}\times\mathbb{Y}$. L'ensemble d'apprentissage est noté $\mathbb{S} = \{(\mathbf{x}_i, y_i)\}_{i=1}^m \sim \mathcal{D}^m$. L'ensemble d'hypothèses $\mathbb{H}$ est composé de fonctions $h : \mathbb{X} \to \mathbb{Y}$. Étant donné $\mathbb{S}$ et $\mathbb{H}$, l'apprenant a pour objectif de trouver l'hypothèse $h \in \mathbb{H}$ qui minimise le risque réel $\mathrm{R}^{\ell}_{\mathcal{D}}(h) = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\, \ell(h,(\mathbf{x},y))$, où $\ell : \mathbb{H} \times (\mathbb{X}\times\mathbb{Y}) \to [0,1]$ est une fonction perte. Le risque empirique associé est défini par $\widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h) = \frac{1}{m}\sum_{i=1}^m \ell(h,(\mathbf{x}_i,y_i))$.

Pour faciliter la lecture de ce chapitre, nous rappelons la borne générale de Bégin et al. (2016) qui est au cœur de notre contribution (rappelée également dans la Section 2.4.5) Cette borne dépend de la divergence de Rényi (de paramètre $\lambda > 1$) entre ρ et π définie par

$$\mathrm{D}_\lambda(\rho\|\pi) = \frac{1}{\lambda-1}\ln\left[\mathop{\mathbb{E}}_{h\sim\pi}\left[\frac{\rho(h)}{\pi(h)}\right]^\lambda\right].$$

Théorème 2.4.9 (Borne PAC-Bayésienne générale de Bégin et al. (2016))**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble d'hypothèses $\mathbb{H}$, pour toute distribution $\pi\in\mathbb{M}^*(\mathbb{H})$, pour toute fonction mesurable $\varphi : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y})^m \to \mathbb{R}^+_*$, pour tout $\lambda > 1$, pour tout $\delta \in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\begin{array}{l}\forall\rho\in\mathbb{M}(\mathbb{H}),\\ \dfrac{\lambda}{\lambda-1}\ln\left(\mathop{\mathbb{E}}_{h\sim\rho}\varphi(h,\mathbb{S})\right) \le \mathrm{D}_\lambda(\rho\|\pi) + \ln\left(\dfrac{1}{\delta}\mathop{\mathbb{E}}_{\mathbb{S}'\sim\mathcal{D}^m}\mathop{\mathbb{E}}_{h'\sim\pi}\varphi(h',\mathbb{S}')^{\frac{\lambda}{\lambda-1}}\right)\end{array}\right] \ge 1-\delta.$$

Une notion clé est que les bornes PAC-Bayésiennes s'appliquent sur l'espérance des risques individuels des classifieurs de $\mathbb{H}$, *i.e.*, le risque de Gibbs. Pour des tâches d'apprentissage classiques, une difficulté est donc de "dérandomiser" ces bornes pour obtenir des garanties pour un classifieur déterministe (*i.e.*, en "supprimant" l'espérance sur $\mathbb{H}$). Dans certains cas, cette dérandomisation est le résultat de la structure des hypothèses, comme pour les classifieurs linéaires stochastiques qui peuvent directement être exprimés comme un classifieur linéaire déterministe (Germain et al., 2009) (comme nous l'avons fait dans le Chapitre 4). Cependant, dans d'autres situations, cette dérandomisation est plus complexe et est spécifique à la classe des hypothèses (*e.g.*, pour les réseaux de neurones, Neyshabur et al. (2018), Nagarajan et Kolter (2019a, Ap. J), Biggs et Guedj (2022)).

8.3 Théorèmes PAC-Bayésiens désintégrés

Dans cette section, nous présentons notre contribution principale : un cadre général de dérandomisation basé sur la divergence de Rényi pour désintégrer les bornes PAC-Bayes.

8.3.1 Rappel de la forme des bornes PAC-Bayes désintégrées

Nous rappelons la forme générale des bornes désintégrées utilisées dans ce chapitre.

Définition 2.5.1 (Borne en généralisation PAC-Bayésienne désintégrée)**.** Soit une mesure de l'écart en généralisation $\phi : [0,1]^2 \to [0,1]$. Une borne PAC-Bayésienne désintégrée est définie telle que pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble d'hypothèses $\mathbb{H}$ avec $\ell : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y}) \to [0,1]$ une fonction perte, pour toute distribution *prior* $\pi \in \mathbb{M}^*(\mathbb{H})$, pour tout algorithme $A : (\mathbb{X}\times\mathbb{Y})^m \times \mathbb{M}^*(\mathbb{H}) \to \mathbb{M}(\mathbb{H})$, il existe une fonction $\Phi : \mathbb{M}(\mathbb{H})\times\mathbb{M}^*(\mathbb{H})\times]0,1] \to \mathbb{R}$ telle que pour tout $\delta \in]0,1]$ on a

$$\mathop{\mathbb{P}}_{\substack{\mathbb{S}\sim\mathcal{D}^m \\ h\sim\rho_{\mathbb{S}}}} \left[\phi(\mathsf{R}^\ell_{\mathcal{D}}(h), \widehat{\mathsf{R}}^\ell_{\mathbb{S}}(h)) \;\leq\; \Phi\Big(\rho_{\mathbb{S}}, \pi, \delta\Big) \right] \;\geq\; 1-\delta,$$

où $\rho_{\mathbb{S}} = A(\mathbb{S}, \pi)$ est la sortie de l'algorithme déterministe A et $\phi()$ est, par exemple, $\phi(\mathsf{R}^\ell_{\mathcal{D}}(h), \widehat{\mathsf{R}}^\ell_{\mathbb{S}}(h)) = |\mathsf{R}^\ell_{\mathcal{D}}(h) - \widehat{\mathsf{R}}^\ell_{\mathbb{S}}(h)|$.

Ici, le *posterior* $\rho_{\mathbb{S}} = A(\mathbb{S}, \pi)$ est construit par un algorithme *déterministe* choisi *a priori* et noté $A : (\mathbb{X}\times\mathbb{Y})^m \times \mathbb{M}^*(\mathbb{H}) \to \mathbb{M}(\mathbb{H})$. Cet algorithme *(i)* prend en entrée un ensemble d'apprentissage $\mathbb{S} \in (\mathbb{X}\times\mathbb{Y})^m$ et une distribution *prior* $\pi \in \mathbb{M}^*(\mathbb{H})$, et *(ii)* renvoie une distribution dépendante des données $\rho_{\mathbb{S}} = A(\mathbb{S}, \pi)$. Ce genre de borne permet de dérandomiser les bornes PAC-Bayes classiques de la manière suivante. Au lieu de considérer une borne valide pour toutes les distributions *posterior* sur $\mathbb{H}$ (le "$\forall\rho$" du Théorème 2.4.9), nous considérons uniquement la distribution *posterior* $\rho_{\mathbb{S}}$ obtenue via l'algorithme A. La borne ci-dessus est alors valide pour l'hypothèse unique h tirée selon $\rho_{\mathbb{S}}$ au lieu du classifieur stochastique de Gibbs : les risques individuels ne sont plus moyennés par rapport à $\rho_{\mathbb{S}}$; c'est la *désintégration d'une borne PAC-Bayésienne*. La dépendance en la probabilité $\rho_{\mathbb{S}}$ signifie que la borne est valide avec une probabilité d'au moins $1-\delta$ sur le choix aléatoire de l'échantillon d'apprentissage $\mathbb{S} \sim \mathcal{D}^m$ et de l'hypothèse $h \sim \rho_{\mathbb{S}}$.

Selon ce principe, nous introduisons dans les Théorèmes 8.3.1 et 8.3.2 ci-dessous deux bornes PAC-Bayésiennes désintégrées générales. Un atout clé de nos résultats est que les bornes sont instanciables à des contextes spécifiques comme pour les bornes PAC-Bayésiennes

"classiques" (*e.g.*, avec des données *i.i.d.*/non-*i.i.d.*, des pertes non bornées, etc). L'instanciation d'une telle borne permet d'obtenir une borne pour être optimisée, conduisant alors à un algorithme auto-certifié (Freund, 1998) avec des garanties théoriques. Pour illustrer l'intérêt de nos résultats, nous présentons, dans la Section 8.4, une telle instanciation pour les réseaux de neurones.

8.3.2 Bornes désintégrées avec la divergence de Rényi

8.3.2.1 Une borne désintégrée générale

Dans le même esprit que le Théorème 2.4.9, notre résultat principal, énoncé dans le Théorème 8.3.1 ci-dessous, est une borne générale basée sur la divergence de Rényi $\mathrm{D}_\lambda(\rho_{\mathbb{S}}\|\pi)$ d'ordre $\lambda>1$.

Théorème 8.3.1 (Borne PAC-Bayésienne désintégrée générale)**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble d'hypothèses $\mathbb{H}$, pour toute distribution *prior* $\pi\in\mathbb{M}^*(\mathbb{H})$, pour toute fonction mesurable $\varphi:\mathbb{H}\times(\mathbb{X}\times\mathbb{Y})^m\to\mathbb{R}_+^*$, pour tout $\lambda>1$, pour tout $\delta\in]0,1]$, pour tout algorithme $A:(\mathbb{X}\times\mathbb{Y})^m\times\mathbb{M}^*(\mathbb{H})\to\mathbb{M}(\mathbb{H})$, on a

$$\mathop{\mathbb{P}}_{\substack{\mathbb{S}\sim\mathcal{D}^m\\ h\sim\rho_{\mathbb{S}}}}\left[\frac{\lambda}{\lambda-1}\ln\left[\varphi(h,\mathbb{S})\right]\le\frac{2\lambda-1}{\lambda-1}\ln\frac{2}{\delta}+\mathrm{D}_\lambda(\rho_{\mathbb{S}}\|\pi)+\ln\left(\mathop{\mathbb{E}}_{\mathbb{S}'\sim\mathcal{D}^m}\mathop{\mathbb{E}}_{h'\sim\pi}\varphi(h',\mathbb{S}')^{\frac{\lambda}{\lambda-1}}\right)\right]\ge1-\delta,$$

où $\rho_{\mathbb{S}}=A(\mathbb{S},\pi)$ est la sortie de l'algorithme déterministe A.

Comme pour les bornes PAC-Bayésiennes générales classiques, ce théorème est vu comme le point de départ pour dériver des bornes en généralisation en fonction du choix de $\varphi()$ (comme dans le Corollaire 8.4.1 ci-après). D'un point de vue technique, dans la preuve de notre théorème, l'utilisation de l'inégalité Hölder se fait de manière différente que dans les preuves PAC-Bayésiennes classiques. En effet, dans la preuve de Bégin et al. (2016, Th. 8) l'étape de changement de mesure basée sur l'inégalité de Hölder est l'étape clé pour obtenir des bornes pour toute distribution *posterior* ρ, alors que notre borne est valide pour une seule distribution *posterior* $\rho_{\mathbb{S}}$ qui dépend de l'ensemble d'apprentissage $\mathbb{S}$ et de la distribution *prior* π. En fait, nous utilisons l'inégalité de Hölder pour introduire une distribution *prior* π indépendante de l'ensemble $\mathbb{S}$: un point crucial pour la borne que nous instancions dans le Corollaire 8.4.1.

En comparaison du Théorème 2.4.9, notre borne requiert un terme[1] supplémentaire $\ln 2+\frac{\lambda}{\lambda-1}\ln\frac{2}{\delta}$. Cependant, en fixant $\varphi(h,\mathbb{S})=m\,\mathrm{kl}(\widehat{\mathrm{R}}_{\mathbb{S}}^{\ell}(h)\|\mathrm{R}_{\mathcal{D}}^{\ell}(h))$ et $\lambda{=}2$, le terme $\ln\frac{8}{\delta^2}$ est multiplié par $\frac{1}{m}$ qui correspond à un coût raisonnable pour "dérandomiser" une borne. Par exemple, si $m=5\,000$ et $\delta=0.05$, on a $\frac{1}{m}\ln\frac{8}{\delta^2}\approx0.002$.

Nous instancions maintenant le Théorème 8.3.1 avec $\lambda\to1^+$ et $\lambda\to+\infty$. Ce résultat montre que la borne converge quand $\lambda\to1^+$ et $\lambda\to+\infty$.

1. Au lieu du terme "classique" $\ln\frac{1}{\delta}$, notre borne fait intervenir $\frac{2\lambda-1}{\lambda-1}\ln\frac{2}{\delta}$, la différence entre ces deux termes est donc $\frac{2\lambda-1}{\lambda-1}\ln\frac{2}{\delta}-\ln\frac{1}{\delta}=\ln 2+\frac{\lambda}{\lambda-1}\ln\frac{2}{\delta}$.

Corollaire 8.3.1 (Cas limites du Théorème 8.3.1)**.** Sous les hypothèses du Théorème 8.3.1, lorsque $\lambda \to 1^+$, on a

$$\mathop{\mathbb{P}}_{\substack{\mathbb{S}\sim\mathcal{D}^m \\ h\sim\rho_{\mathbb{S}}}}\left[\ln\left[\varphi(h,\mathbb{S})\right] \;\leq\; \ln\frac{2}{\delta} + \ln\left[\operatorname*{esssup}_{\mathbb{S}'\in(\mathbb{X}\times\mathbb{Y}),h'\in\mathbb{H}} \varphi(h',\mathbb{S}')\right]\right] \geq 1-\delta,$$

lorsque $\lambda \to +\infty$, on a

$$\mathop{\mathbb{P}}_{\substack{\mathbb{S}\sim\mathcal{D}^m \\ h\sim\rho_{\mathbb{S}}}}\left[\ln\left[\varphi(h,\mathbb{S})\right] \;\leq\; \ln\left[\operatorname*{esssup}_{h'\in\mathbb{H}} \frac{\rho_{\mathbb{S}}(h')}{\pi(h')}\right] + \ln\left[\frac{4}{\delta^2}\mathop{\mathbb{E}}_{\mathbb{S}'\sim\mathcal{D}^m}\mathop{\mathbb{E}}_{h'\sim\pi}\varphi(h',\mathbb{S}')\right]\right] \geq 1-\delta,$$

où esssup est le supremum essentiel défini comme le supremum sur un ensemble avec des mesures de probabilité non nulles, *i.e.*,

$$\operatorname*{esssup}_{\mathbb{S}'\in(\mathbb{X}\times\mathbb{Y}),h'\in\mathbb{H}} \varphi(h',\mathbb{S}') = \inf\left\{\tau\in\mathbb{R}, \mathop{\mathbb{P}}_{\substack{\mathbb{S}\sim\mathcal{D}^m \\ h\sim\rho_{\mathbb{S}}}}\Big[\varphi(h,\mathbb{S}) > \tau\Big] = 0\right\},$$

$$\text{et}\quad \operatorname*{esssup}_{h'\in\mathbb{H}} \frac{\rho_{\mathbb{S}}(h')}{\pi(h')} = \inf\left\{\tau\in\mathbb{R}, \mathop{\mathbb{P}}_{h\sim\rho_{\mathbb{S}}}\left[\frac{\rho_{\mathbb{S}}(h)}{\pi(h)} > \tau\right] = 0\right\}.$$

Ce corollaire montre que le paramètre λ contrôle le compromis entre la divergence de Rényi $\mathrm{D}_\lambda(\rho_{\mathbb{S}}\|\pi)$ et $\ln\left[\mathbb{E}_{\mathbb{S}'}\,\mathbb{E}_{h'\sim\pi}\,\varphi(h',\mathbb{S}')^{\frac{\lambda}{\lambda-1}}\right]$. En effet, si $\lambda\to1^+$, on a $\mathrm{D}_\lambda(\rho_{\mathbb{S}}\|\pi)\to0$ alors que les autres termes convergent vers $\ln\left[\operatorname{esssup}_{\mathbb{S}',h'}\varphi(h',\mathbb{S}')\right]$, *i.e.*, la valeur maximale possible du second terme. Si $\lambda\to+\infty$, la divergence de Rényi augmente et tend vers $\ln\operatorname{esssup}_{h'}\frac{\rho_{\mathbb{S}}(h')}{\pi(h')}$ et les autres termes diminuent et tendent vers $\ln[\mathbb{E}_{\mathbb{S}'\sim\mathcal{D}^m}\,\mathbb{E}_{h'\sim\pi}\,\varphi(h',\mathbb{S}')]$.

8.3.2.2 Comparaison avec la borne de Rivasplata et al. (2020)

Pour la comparaison, nous rappelons dans le Théorème 2.5.1 la borne de Rivasplata et al. (2020, Th.1(i)), qui est plus générale que les bornes de Blanchard et Fleuret (2007) et de Catoni (2007, Th.1.2.7).

Théorème 2.5.1 (Borne désintégrée générale de Rivasplata et al. (2020))**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble d'hypothèses $\mathbb{H}$, pour toute distribution *prior* $\pi\in\mathbb{M}^*(\mathbb{H})$, pour toute fonction mesurable $\varphi:\mathbb{H}\times(\mathbb{X}\times\mathbb{Y})^m\to\mathbb{R}$, pour tout $\delta\in]0,1]$, pour tout algorithme $A:(\mathbb{X}\times\mathbb{Y})^m\times\mathbb{M}^*(\mathbb{H})\to\mathbb{M}(\mathbb{H})$, on a

$$\mathop{\mathbb{P}}_{\substack{\mathbb{S}\sim\mathcal{D}^m \\ h\sim\rho_{\mathbb{S}}}}\left[\varphi(h,\mathbb{S}) \leq \underbrace{\ln\left[\frac{\rho_{\mathbb{S}}(h)}{\pi(h)}\right] + \ln\left[\frac{1}{\delta}\mathop{\mathbb{E}}_{\mathbb{S}'\sim\mathcal{D}^m}\mathop{\mathbb{E}}_{h'\sim\pi}\exp\left(\varphi(h',\mathbb{S}')\right)\right]}_{\Phi(\rho_{\mathbb{S}},\pi,\delta)}\right] \geq 1-\delta,$$

où $\rho_{\mathbb{S}} = A(\mathbb{S},\pi)$ est la sortie de l'algorithme déterministe A.

Notons que la borne peut être réécrite avec le logarithme. On a

$$\mathop{\mathbb{P}}_{\substack{\mathbb{S}\sim\mathcal{D}^m,\\ h\sim\rho_{\mathbb{S}}}}\left[\ln\left(\varphi(h,\mathbb{S})\right) \leq \ln\left[\frac{\rho_{\mathbb{S}}(h)}{\pi(h)}\right]+\ln\left[\frac{1}{\delta}\mathop{\mathbb{E}}_{\mathbb{S}'\sim\mathcal{D}^m}\mathop{\mathbb{E}}_{h'\sim\pi}\varphi(h',\mathbb{S}')\right]\right]\geq 1-\delta.$$

Le terme $\ln\frac{\rho_{\mathbb{S}}(h)}{\pi(h)}$ (présent également dans les résultats de Blanchard et Fleuret (2007) et Catoni (2007)) peut être interprété comme une "KL-divergence désintégrée[2]" qui dépend uniquement de $h\sim\rho_{\mathbb{S}}$. En revanche, notre borne fait apparaître la divergence de Rényi entre les distributions *prior* π et *posterior* $\rho_{\mathbb{S}}$. Cela a pour conséquence que notre borne ne comporte qu'un seul terme dépendant de l'hypothèse tirée (le risque) : la valeur de la divergence est donc la même quelle que soit l'hypothèse. En théorie, notre borne est moins précise à cause de la divergence de Rényi, (voir Erven et Harremoës, 2014) et de la dépendance en δ moins favorable que celle du Théorème 2.5.1. Cependant, notre terme de divergence est le principal avantage de notre borne. En effet, confirmé par nos expériences, le terme $\mathrm{D}_\lambda(\rho_{\mathbb{S}}\|\pi)$ rend l'apprentissage (avec notre algorithme auto-certifié) plus stable et plus efficace en comparaison à l'optimisation de la borne du Théorème 2.5.1 où $\ln\frac{\rho_{\mathbb{S}}(h)}{\pi(h)}$ est sujet à une grande variabilité.

8.3.2.3 Une borne désintégrée générale paramétrable

En PAC-Bayes, les bornes paramétrées ont été introduites pour contrôler le compromis entre le risque empirique et la divergence (Catoni, 2007 ; Thiemann et al., 2017). Nous présentons une version paramétrée de notre borne afin d'élargir son champ d'application pratique.

Théorème 8.3.2 (Borne PAC-Bayes désintégrée paramétrable)**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble d'hypothèses $\mathbb{H}$, pour toute distribution *prior* $\pi\in\mathbb{M}^*(\mathbb{H})$, pour toute fonction mesurable $\varphi:\mathbb{H}\times(\mathbb{X}\times\mathbb{Y})^m\to\mathbb{R}_+^*$, pour tout $\delta\in]0,1]$, pour tout algorithme $A:(\mathbb{X}\times\mathbb{Y})^m\times\mathbb{M}^*(\mathbb{H})\to\mathbb{M}(\mathbb{H})$, on a

$$\mathop{\mathbb{P}}_{\substack{\mathbb{S}\sim\mathcal{D}^m\\ h\sim\rho_{\mathbb{S}}}}\left[\forall\lambda>0,\ \ln\left(\varphi(h,\mathbb{S})\right)\leq\ln\left(\frac{\lambda}{2}e^{\mathrm{D}_2(\rho_{\mathbb{S}}\|\pi)}+\frac{8}{2\lambda\delta^3}\mathop{\mathbb{E}}_{\mathbb{S}'\sim\mathcal{D}^m}\mathop{\mathbb{E}}_{h'\sim\pi}\left[\varphi(h',\mathbb{S}')^2\right]\right)\right]\geq 1-\delta,$$

où $\rho_{\mathbb{S}}=A(\mathbb{S},\pi)$ est la sortie de l'algorithme déterministe A.

Le terme $e^{\mathrm{D}_2(\rho_{\mathbb{S}}\|\pi)}$ est lié à la distance χ^2. En effet, on a : $\chi^2(\rho_{\mathbb{S}}\|\pi)=\mathbb{E}_{h\sim\pi}\left[\frac{\rho_{\mathbb{S}}(h)}{\pi(h)}\right]^2-1=e^{\mathrm{D}_2(\rho_{\mathbb{S}}\|\pi)}-1$. Un atout du Théorème 8.3.2 est le paramètre λ de contrôle du compromis entre l'exponentielle de la divergence de Rényi $e^{\mathrm{D}_2(\rho_{\mathbb{S}}\|\pi)}$ et $\frac{1}{\delta^3}\mathbb{E}_{\mathbb{S}'}\mathbb{E}_{h'\sim\pi}\varphi(h',\mathbb{S}')^2$. Notre borne est valide pour tout $\lambda>0$, ainsi, d'un point de vue pratique, nous pouvons apprendre le paramètre λ pour minimiser la borne et contrôler l'instabilité numérique possible due à $\mathrm{D}_2(\rho_{\mathbb{S}}|\pi)$. En effet, si $\mathrm{D}_2(\rho_{\mathbb{S}}|\pi)$ est élevé, le calcul peut conduire à une valeur infinie en raison de la précision arithmétique limitée. Il est important de préciser que, tout comme pour les autres bornes paramétrées (*e.g.*, Thiemann et al., 2017), il existe une solution en forme close pour le paramètre optimal λ. Cette solution est dérivée dans la Proposition 8.3.1 et montre que la borne optimale du Théorème 8.3.2 correspond à celle du Théorème 8.3.1.

2. La KL-divergence est dîtes *désintégrée*, car le log n'est pas moyenné contrairement à la KL-divergence.

Proposition 8.3.1 (Borne optimale de Théorème 8.3.2)**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble d'hypothèses $\mathbb{H}$, pour toute distribution *prior* $\pi\in\mathbb{M}^*(\mathbb{H})$, pour toute fonction mesurable $\varphi:\mathbb{H}\times(\mathbb{X}\times\mathbb{Y})^m\to\mathbb{R}_+^*$, pour tout $\delta\in]0,1]$, pour tout algorithme $A:(\mathbb{X}\times\mathbb{Y})^m\times\mathbb{M}^*(\mathbb{H})\to\mathbb{M}(\mathbb{H})$, soit

$$\lambda^*=\operatorname*{argmin}_{\lambda>0}\ln\left[\frac{\lambda}{2}e^{\mathrm{D}_2(\rho_{\mathbb{S}}\|\pi)}+\frac{\mathop{\mathbb{E}}_{\mathbb{S}'\sim\mathcal{D}^m}\mathop{\mathbb{E}}_{h'\sim\pi}\left[8\varphi(h',\mathbb{S}')^2\right]}{2\lambda\delta^3}\right],$$

alors, on a $\overbrace{2\ln\left[\frac{\lambda^*}{2}e^{\mathrm{D}_2(\rho_{\mathbb{S}}\|\pi)}+\mathop{\mathbb{E}}_{\mathbb{S}'\sim\mathcal{D}^m}\mathop{\mathbb{E}}_{h'\sim\pi}\left(\frac{8\varphi(h',\mathbb{S}')^2}{2\lambda^*\delta^3}\right)\right]}^{\text{Théorème 8.3.2}}$

$$=\underbrace{\mathrm{D}_2(\rho_{\mathbb{S}}\|\pi)+\ln\left[\mathop{\mathbb{E}}_{\mathbb{S}'\sim\mathcal{D}^m}\mathop{\mathbb{E}}_{h'\sim\pi}\left(\frac{8\varphi(h',\mathbb{S}')^2}{\delta^3}\right)\right]}_{\text{Théorème 8.3.1 avec }\lambda=2},$$

où $\lambda^*=\sqrt{\dfrac{\mathbb{E}_{\mathbb{S}'\sim\mathcal{D}^m}\mathbb{E}_{h'\sim\pi}\left[8\varphi(h',\mathbb{S}')^2\right]}{\delta^3\exp(\mathrm{D}_2(\rho_{\mathbb{S}}\|\pi))}}$.

Le λ^* optimal amène donc à la même borne pour les Théorèmes 8.3.1 et 8.3.2.

8.4 La désintégration en action

Cette section présente un exemple d'instanciation du Théorème 8.3.1 pour les réseaux de neurones, montrant l'utilité de nos résultats en comparaison des bornes classiques.

8.4.1 Spécialisation aux réseaux de neurones

L'objectif est d'apprendre les poids d'un réseau de neurones (NN, pour *Neural Networks*) qui minimisent le risque réel $\mathrm{R}_{\mathcal{D}}(h)=\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\mathrm{I}\left[h(\mathbf{x})\neq y\right]$. L'ensemble d'hypothèses $\mathbb{H}$ est un ensemble de NNs avec des poids différents pour une architecture donnée. Les utilisateurs procèdent généralement par *epochs* et obtiennent un NN "intermédaire" après chaque *epoch*. Puis, ils sélectionnent le NN intermédiaire associé au risque en validation le plus faible. Nous proposons de traduire cette pratique dans notre cadre PAC-Bayésien en considérant un *prior* par *epoch*. Avec T *epochs*, on a donc T *priors* $\mathbb{P}=\{\pi_t\}_{t=1}^T$, où $\forall t\in\{1,\ldots,T\},\pi_t=\mathcal{N}(\mathbf{v}_t,\sigma^2\mathbf{I}_D)$ est une distribution gaussienne centrée en $\mathbf{v}_t$ (le vecteur de poids associé au t-ième NN intermédiaire) de matrice de covariance $\sigma^2\mathbf{I}_D$ (où $\mathbf{I}_D$ est la matrice identité de dimension $D\times D$). En supposant que les T *priors* soient appris avec un ensemble $\mathbb{S}_{\text{prior}}$ tel que $\mathbb{S}_{\text{prior}}\bigcap\mathbb{S}=\emptyset$, alors les Corollaires 8.4.1 et 8.4.2 nous guident pour apprendre un *posterior* $\rho_{\mathbb{S}}=\mathcal{N}(\mathbf{w},\sigma^2\mathbf{I}_D)$ à partir du *prior* $\pi\in\mathbb{P}$ qui minimise le risque empirique sur $\mathbb{S}$ (plus de détails sont donnés après les énoncés des corollaires). Le fait de considérer des distributions Gaussiennes a l'avantage de simplifier l'expression de la KL-divergence, ce qui est fréquemment utilisé en PAC-Bayes pour les NNs (*e.g.*, Dziugaite et Roy, 2017 ; Letarte et al., 2019a ; Zhou et al., 2019) [3]

3. Nous rappelons que l'utilisation des distributions Gaussiennes a d'abord été étudiée pour les classifieurs linéaires (*e.g.*, Ambroladze et al., 2006 ; Germain et al., 2009). Dans ce contexte, la symétrie de la distribution Gaussienne permet de simplifier la dérandomisation.

Nous instancions le Théorème 8.3.1 à ce cadre dans le Corollaire 8.4.1.

Corollaire 8.4.1 (Instanciation du Théorème 8.3.1 aux NNs)**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble $\mathbb{P}=\{\pi_1,\ldots,\pi_T\}$ de T *priors* sur $\mathbb{H}$ où $\pi_t=\mathcal{N}(\mathbf{v}_t,\sigma^2\mathbf{I}_D)$, pour tout algorithme $A:(\mathbb{X}\times\mathbb{Y})^m\times\mathbb{M}^*(\mathbb{H})\to\mathbb{M}(\mathbb{H})$, pour toute perte $\ell:\mathbb{H}\times(\mathbb{X}\times\mathbb{Y})\to[0,1]$, pour tout $\delta\in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\substack{\mathbb{S}\sim\mathcal{D}^m\\ h\sim\rho_{\mathbb{S}}}}\left[\forall\pi_t\in\mathbb{P},\ \mathrm{kl}(\widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h)\|\mathsf{R}^{\ell}_{\mathcal{D}}(h))\ \leq\ \frac{1}{m}\left(\frac{\|\mathbf{w}-\mathbf{v}_t\|_2^2}{\sigma^2}+\ln\frac{16T\sqrt{m}}{\delta^3}\right)\right]\ \geq\ 1-\delta,$$

où $\mathrm{kl}(a\|b)=a\ln\frac{a}{b}+(1-a)\ln\frac{1-a}{1-b}$, et $\rho_{\mathbb{S}}=\mathcal{N}(\mathbf{w},\sigma^2\mathbf{I}_D)$, et où l'hypothèse $h\sim\rho_{\mathbb{S}}$ est paramétrée par $\mathbf{w}+\boldsymbol{\epsilon}$.

Par souci de comparaison, le Corollaire 8.4.2 instancie d'autres bornes désintégrées de la littérature : l'Équation (8.1) correspond à la borne de Rivasplata et al. (2020) (Théorème 2.5.1), l'Équation (8.2) à la borne de Blanchard et Fleuret (2007) et l'Équation (8.3) à la borne de Catoni (2007).

Corollaire 8.4.2 (Instanciation de bornes de la littérature pour les NNs)**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble $\mathbb{P}=\{\pi_1,\ldots,\pi_T\}$ de T *priors* sur $\mathbb{H}$ où $\pi_t=\mathcal{N}(\mathbf{v}_t,\sigma^2\mathbf{I}_D)$, pour tout algorithme $A:(\mathbb{X}\times\mathbb{Y})^m\times\mathbb{M}^*(\mathbb{H})\to\mathbb{M}(\mathbb{H})$, pour toute perte $\ell:\mathbb{H}\times(\mathbb{X}\times\mathbb{Y})\to\{0,1\}$, pour tout $\delta\in]0,1]$, avec une probabilité d'au moins $1-\delta$ sur $\mathbb{S}\sim\mathcal{D}^m$ et sur $h\sim\rho_{\mathbb{S}}$ paramétrée par $\mathbf{w}+\boldsymbol{\epsilon}$, on a pour $\forall\pi_t\in\mathbb{P}$

$$\mathrm{kl}(\widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h)\|\mathsf{R}^{\ell}_{\mathcal{D}}(h))\leq\frac{1}{m}\left[\frac{\|\mathbf{w}+\boldsymbol{\epsilon}-\mathbf{v}_t\|_2^2-\|\boldsymbol{\epsilon}\|_2^2}{2\sigma^2}+\ln\frac{2T\sqrt{m}}{\delta}\right],\tag{8.1}$$

$$\forall b\in\mathbb{b},\ \mathrm{kl}_+(\widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h)\|\mathsf{R}^{\ell}_{\mathcal{D}}(h))\leq\frac{1}{m}\left[\frac{b+1}{b}\left[\frac{\|\mathbf{w}+\boldsymbol{\epsilon}-\mathbf{v}_t\|_2^2-\|\boldsymbol{\epsilon}\|_2^2}{2\sigma^2}\right]_+\!+\ln\frac{(b+1)T\,\mathrm{card}(\mathbb{b})}{\delta}\right],\tag{8.2}$$

$$\forall c\in\mathbb{c},\ \mathsf{R}^{\ell}_{\mathcal{D}}(h)\leq\frac{1-\exp\left(-c\widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h)-\frac{1}{m}\left[\frac{\|\mathbf{w}+\boldsymbol{\epsilon}-\mathbf{v}_t\|_2^2-\|\boldsymbol{\epsilon}\|_2^2}{2\sigma^2}+\ln\frac{T\,\mathrm{card}(\mathbb{c})}{\delta}\right]\right)}{1-e^{-c}},\tag{8.3}$$

où $[x]_+=\max(x,0)$, et $\mathrm{kl}_+(\widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h)\|\mathsf{R}^{\ell}_{\mathcal{D}}(h))=\mathrm{kl}(\widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h)\|\mathsf{R}^{\ell}_{\mathcal{D}}(h))$ si $\widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h)<\mathsf{R}^{\ell}_{\mathcal{D}}(h)$ et 0 sinon. De plus, $\boldsymbol{\epsilon}\sim\mathcal{N}(\mathbf{0},\sigma^2\mathbf{I}_D)$ est un bruit gaussien tel que $\mathbf{w}+\boldsymbol{\epsilon}$ sont les poids de $h\sim\rho_{\mathbb{S}}$ avec $\rho_{\mathbb{S}}=\mathcal{N}(\mathbf{w},\sigma^2\mathbf{I}_D)$, et $\mathbb{c}$, $\mathbb{b}$ sont deux ensembles d'hyperparamètres fixés *a priori*.

Comme λ du Théorème 8.3.2, le paramètre c contrôle le compromis entre le risque empirique $\widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h)$ et $\frac{1}{m}\left[\frac{\|\mathbf{w}+\boldsymbol{\epsilon}-\mathbf{v}_t\|_2^2-\|\boldsymbol{\epsilon}\|_2^2}{2\sigma^2}+\ln\frac{T\,\mathrm{card}(\mathbb{c})}{\delta}\right]$. Le paramètre b contrôle quant à lui la précision de la borne. Ces paramètres peuvent être ajustés pour minimiser les Équations (8.2) et (8.3). Cependant, il n'existe pas de solution en forme close pour l'expression du minimum de ces bornes. Ainsi, pour apprendre la distribution $\rho_{\mathbb{S}}$ associée à la plus petite valeur de ces bornes, notre protocole expérimental met en œuvre une minimisation des bornes par descente de gradient pour tout $b\in\mathbb{b}$ ou $c\in\mathbb{c}$. Pour obtenir une borne précise, la divergence entre un *prior* $\pi_t\in\mathbb{P}$ et $\rho_{\mathbb{S}}$ doit être faible, *i.e.*, $\|\mathbf{w}-\mathbf{v}_t\|_2^2$ ou $\|\mathbf{w}+\boldsymbol{\epsilon}-\mathbf{v}_t\|_2^2-\|\boldsymbol{\epsilon}\|_2^2$) doit être faible. Une solution est de découper l'ensemble d'apprentissage en deux sous-ensembles d'intersection vide $\mathbb{S}_{\text{prior}}$ et $\mathbb{S}$. L'ensemble $\mathbb{S}_{\text{prior}}$ est utilisé pour apprendre le *prior* alors que $\mathbb{S}$ est utilisé

pour apprendre le *posterior* et pour calculer la borne. Ainsi, si nous "pré-apprenons" avec $\mathbb{S}_{\text{prior}}$ un *prior* suffisamment bon $\pi_t \in \mathbb{P}$, alors nous pouvons nous attendre à avoir une valeur faible de $\|\mathbf{w}-\mathbf{v}_t\|_2$.

Procédure d'apprentissage

L'ensemble d'apprentissage originale est découpé en deux ensembles distincts : $\mathbb{S}_{\text{prior}}$ et $\mathbb{S}$ (*resp.* de taille m_{prior} et m). L'apprentissage s'effectue en deux étapes.
1) Le *prior* π est "pré-appris" avec un algorithme d'apprentissage arbitraire A_{prior} avec $\mathbb{S}_{\text{prior}}$ et est sélectionné par *early stopping* avec $\mathbb{S}$ comme ensemble de validation.
2) Étant donné $\mathbb{S}$ et π, nous apprenons le *posterior* $\rho_{\mathbb{S}}$ avec l'algo. A (défini *a priori*).

À première vue, la sélection des poids *a priori* avec $\mathbb{S}$ par *early stopping* peut ressembler à de la "triche". Cependant, cette procédure peut être vue comme : *(i)* construction de $\mathbb{P}$ à partir des T NNs "intermédiaires" appris à chaque *epoch* à partir de $\mathbb{S}_{\text{prior}}$, puis *(ii)* optimisation de la borne avec le *prior* associé au meilleur risque sur $\mathbb{S}$. Cette procédure donne un résultat statistiquement valide : le Corollaire 8.4.1 est valide pour tout $\pi_t \in \mathbb{P}$, ce qui signifie que nous pouvons sélectionner celui que nous voulons, en particulier celui qui minimise $\widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h)$ pour une hypothèse $h \sim \pi_t$. Cette heuristique a du sens puisqu'elle permet de détecter si un *prior* est concentré autour d'hypothèses qui sur-apprennent potentiellement l'ensemble d'apprentissage $\mathbb{S}_{\text{prior}}$. Habituellement, les utilisateurs considèrent ce "*prior optimal*" comme le NN final. Dans notre cas, l'avantage est que nous raffinons ce "*prior optimal*" avec $\mathbb{S}$ pour apprendre le *posterior* $\rho_{\mathbb{S}}$. Notons que Pérez-Ortiz et al. (2021) ont introduit des bornes en généralisation précises avec des *priors* dépendants des données pour des NNs stochastiques (*i.e.*, non-derandomisés). Notre méthode pour déterminer le *prior* diffère puisque *(i)* nous apprenons T NNs (*i.e.*, T *prior*) au lieu d'un seul et *(ii)* nous fixons la variance de la Gaussienne du *posterior* $\rho_{\mathbb{S}}$. À notre connaissance, notre méthode d'apprentissage pour le *prior* est nouvelle.

8.4.2 Algorithme auto-certifié pour les réseaux de neurones

Nous utilisons la méthode d'apprentissage décrite ci-dessus dans laquelle nous intégrons la minimisation directe de toutes les bornes. Pour optimiser une borne par descente de gradient, nous remplaçons la perte 0-1 non différentiable par une approximation : la cross-entropie bornée (Dziugaite et Roy, 2018). Nous effectuons ce remplacement, d'une part, car la minimisation de la cross-entropie fonctionne bien pour les NNs (Goodfellow et al., 2016) et, d'autre part, car elle est bornée entre 0 et 1 (hypothèse requise pour la fonction $\mathrm{kl}()$). La cross-entropie est définie dans le cadre multiclasse avec $y \in \mathbb{Y}$ par $\ell(h,(\mathbf{x},y)) = -\frac{1}{Z}\ln\left[e^{-Z}+(1-2e^{-Z})h[y]\right] \in [0,1]$ où $h[y]$ est la y-ième sortie du NN ; nous fixons $Z=4$ (la valeur par défaut de Dziugaite et Roy, 2018).

Pour apprendre un *prior* $\pi \in \mathbb{P}$ suffisamment bon et le *posterior* $\rho_{\mathbb{S}}$, nous appliquons notre méthode d'apprentissage avec deux algorithmes de descente de gradient stochastique notés A_{prior} et A. Notons que l'aléa dans l'algorithme de descente de gradient stochastique est fixé pour obtenir des algorithmes déterministes. Durant l'étape **1)** étant donné $\mathbb{S}_{\text{prior}}$, l'algorithme A_{prior} apprend les T *priors* $\{\pi_1,\ldots,\pi_T\}=\mathbb{P}$ (*i.e.*, lors des T *epochs*) en minimisant la perte de cross-entropie bornée. En d'autres termes, à la fin de l'*epoch* t, les poids $\mathbf{w}_t$ du classifieur sont utilisés pour définir le *prior* $\pi_t = \mathcal{N}(\mathbf{w}_t, \sigma^2\mathbf{I}_D)$. Ensuite, le meilleur *prior* $\pi \in \mathbb{P}$ est sélectionné par *early stopping* sur $\mathbb{S}$. Durant l'étape **2)**, étant donné $\mathbb{S}$ et π, l'algorithme A intègre l'optimisation directe des bornes avec la perte de cross-entropie bornée.

8.4.3 À propos des réseaux de neurones stochastiques

À cause de sa nature stochastique, le PAC-Bayes a été utilisé pour étudier les NNs stochastiques (*e.g.*, Dziugaite et Roy, 2017, 2018 ; Zhou et al., 2019 ; Pérez-Ortiz et al., 2021)). Pour comparer les NNs déterministes et les NNs stochastiques, nous instancions la borne du Théorème 2.4.5 pour les NNs stochastiques dans le Corollaire 8.4.3. Nous rappelons que, dans ce chapitre, un NN déterministe est un *unique* h tiré selon la distribution *posterior* $\rho_{\mathbb{S}}=\mathcal{N}(\mathbf{w},\sigma^2\mathbf{I}_D)$ (la sortie de l'algorithme A). Pour chaque exemple, la prédiction est donc effectuée par le même NN déterministe : celui paramétré par les poids $(\mathbf{w}+\boldsymbol{\epsilon})\in\mathbb{R}^D$. À l'inverse, le NN stochastique associé à un *posterior* $\rho=\mathcal{N}(\mathbf{w},\sigma^2\mathbf{I}_D)$ prédit l'étiquette d'un exemple donné *(i)* en tirant h selon ρ, puis *(ii)* en renvoyant l'étiquette prédite par h. Le risque du NN stochastique est donc le risque réel moyenné $\mathbb{E}_{h\sim\rho}\mathsf{R}^{\ell}_{\mathcal{D}}(h)$, où l'espérance est calculée sur *tous* les h tirés selon ρ. Nous calculons le risque empirique du NN stochastique avec une approximation Monte Carlo : *(i)* nous tirons K vecteurs de poids, et *(ii)* nous calculons la moyenne des risques sur les K NNs associés ; nous notons ρ^K la distribution de ces K tirages. Dans ce contexte, nous obtenons la borne PAC-Bayésienne suivante.

Corollaire 8.4.3 (Borne PAC-Bayésienne pour les NNs stochastiques)**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble $\mathbb{P}=\{\pi_1,\ldots,\pi_T\}$ de T *priors* sur $\mathbb{H}$ où $\pi_t=\mathcal{N}(\mathbf{v}_t,\sigma^2\mathbf{I}_D)$, pour toute perte $\ell:\mathbb{H}\times(\mathbb{X}\times\mathbb{Y})\to\{0,1\}$, pour tout $\delta\in]0,1]$, avec une probabilité d'au moins $1-\delta$ sur $\mathbb{S}\sim\mathcal{D}^m$ et $\{h_1,\ldots,h_K\}\sim\rho^K$, on a simultanément pour tout $\pi_t\in\mathbb{P}$,

$$\mathrm{kl}\left(\mathop{\mathbb{E}}_{h\sim\rho}\widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h)\,\middle\|\,\mathop{\mathbb{E}}_{h\sim\rho}\mathsf{R}^{\ell}_{\mathcal{D}}(h)\right)\leq\frac{1}{m}\left[\frac{\|\mathbf{w}-\mathbf{v}_t\|_2^2}{2\sigma^2}+\ln\frac{4T\sqrt{m}}{\delta}\right],\tag{8.4}$$

$$\text{et}\quad\mathrm{kl}\left(\frac{1}{K}\sum_{i=1}^{K}\widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h_i)\,\middle\|\,\mathop{\mathbb{E}}_{h\sim\rho}\widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h)\right)\leq\frac{1}{n}\ln\frac{4}{\delta},\tag{8.5}$$

où $\rho=\mathcal{N}(\mathbf{w},\sigma^2\mathbf{I}_D)$ et l'hypothèse $h\sim\rho$ est paramétrée par $\mathbf{w}+\boldsymbol{\epsilon}$ avec $\boldsymbol{\epsilon}\sim\mathcal{N}(\mathbf{0},\sigma^2\mathbf{I}_D)$.

Ce résultat met en évidence deux caractéristiques qui justifient de le considérer comme référence pour une comparaison équitable entre les bornes PAC-Bayes désintégrées et classiques (donc entre les NNs déterministes et stochastiques). Premièrement, il met en jeu les mêmes termes que le Corollaire 8.4.1. Deuxièmement, il est proche de la borne de Pérez-Ortiz et al. (2021, Sec. 6.2), puisque *(i)* nous adaptons la KL-divergence à notre cadre (*i.e.*, $\mathrm{KL}(\rho|\pi)=\frac{1}{2\sigma^2}\|\mathbf{w}-\mathbf{v}_t\|_2^2$), *(ii)* la borne est valable pour T *priors* grâce à une borne de l'union.

8.5 Résumé des expériences

Nous avons mené nos expériences sur les trois jeux de données suivants : MNIST (LeCun et al., 1998), Fashion-MNIST (Xiao et al., 2017), et CIFAR-10 (Krizhevsky, 2009). Nous avons utilisé notre méthode proposée dans la Section 8.4.2 pour comparer les différentes bornes énoncées dans ce chapitre.

Il est important de préciser, nous n'avons pas cherché à obtenir les performances de l'état de l'art. En fait, nous avons tout d'abord confirmé que le découpage de l'ensemble d'apprentissage original en $50\%/50\%$ pour $(\mathbb{S}_{\mathsf{prior}},\mathbb{S})$, qui est le plus courant en PAC-Bayes (Germain et al., 2009 ; Pérez-Ortiz et al., 2021), est un choix pertinent. Ensuite, nous avons mis en évidence que notre borne désintégrée associée au NN déterministe est plus précise

que la borne moyennée classique associée au NN stochastique (Corollaire 8.4.3). Enfin, nous avons montré que notre borne désintégrée (Corollaire 8.4.1) est plus précise et plus stable que les bornes basées sur Rivasplata et al. (2020), Blanchard et Fleuret (2007) et Catoni (2007) (Corollaire 8.4.2).

En résumé, nos expériences ont montré que notre borne désintégrée est, en pratique, plus précise que celles de la littérature. Cette précision nous permet de borner avec exactitude le risque réel $\mathrm{R}_{\mathcal{D}}(h)$, rendant ainsi la sélection de modèle à partir de la borne désintégrée efficace. De plus, nous avons montré que notre borne est plus facile à optimiser. Cela est principalement dû à la KL-divergence désintégrée qui dépend de l'hypothèse tirée h selon les poids $(\boldsymbol{\omega} + \boldsymbol{\epsilon})$ (ce terme n'apparaît pas dans notre borne). En effet, les gradients de la KL-divergence désintégrée par rapport à $\boldsymbol{\omega}$ dépendent du bruit $\boldsymbol{\epsilon}$, rendant le gradient imprécis (particulièrement avec un "haut" *learning rate* et une petite variance σ^2).

8.6 Conclusion

Dans ce chapitre, nous avons présenté une nouvelle borne PAC-Bayésienne désintégrée (Théorème 8.3.1) lorsque l'étape de dérandomisation consiste en *(i)* l'apprentissage d'une distribution *posterior* $\rho_{\mathbb{S}}$ sur l'ensemble des classifieurs (étant donné un algorithme, un ensemble d'apprentissage $\mathbb{S}$ et une distribution *prior* π) et *(ii)* le tirage d'une hypothèse h à partir de cette distribution $\rho_{\mathbb{S}}$. Bien que notre borne puisse être plus lâche que celles de Blanchard et Fleuret (2007), Catoni (2007) et Rivasplata et al. (2020), elle offre de belles opportunités pour l'apprentissage de classifieurs déterministes.

En effet, notre borne peut être utilisée non seulement pour étudier les garanties théoriques des classifieurs déterministes, mais aussi pour dériver des algorithmes auto-certifiés plus stables et efficaces que ceux issus des bornes de la littérature. Concrètement, les bornes de Blanchard et Fleuret (2007), Catoni (2007) et Rivasplata et al. (2020) dépendent de deux termes liés au classifieur tiré : le risque et la "KL-divergence désintégrée". En revanche, dans notre borne, le terme de divergence de Rényi dépend de l'ensemble des hypothèses, ce qui implique que la divergence reste inchangée, quel que soit le classifieur choisi. En ce sens, notre borne est plus stable, car l'algorithme d'apprentissage de minimisation de la borne permet, en pratique, de choisir une meilleure hypothèse que celles obtenues avec les bornes de Blanchard et Fleuret (2007), Catoni (2007) et Rivasplata et al. (2020). Nous avons illustré l'intérêt de notre borne sur des réseaux de neurones.

Dans le prochain chapitre, nous exploitons la KL-divergence désintégrée et la borne de Rivasplata et al. (2020) pour obtenir des bornes en généralisation avec une mesure de complexité arbitraire.

Bornes en généralisation avec des mesures de complexité arbitraires

9

Contexte

Alors que l'ensemble des contributions des Chapitres précédents se placent directement dans le cadre de la théorie PAC-Bayésienne, ce chapitre introduit un cadre théorique (général) original. En effet, ce cadre permet de dériver des bornes en généralisation qui permettent d'utiliser des mesures de complexité générales mieux corrélées avec l'écart en généralisation. Nous pensons que ces travaux sont d'un intérêt important pour la communauté puisqu'ils offrent une nouvelle direction pour comprendre théoriquement les capacités en généralisation des modèles. Ces travaux, réalisés durant la thèse de Paul Viallard, ont donné lieu à une publication à la conférence AISTATS (Viallard et al., 2024a).

9.1 Introduction

Comme nous l'avons vu dans le Chapitre 2, la théorie de l'apprentissage statistique propose divers cadres théoriques pour évaluer la capacité en généralisation en estimant à quel point le risque empirique est représentatif du risque réel via une majoration de l'écart en généralisation. Cet écart en généralisation représente la différence entre le risque réel et le risque empirique. La majoration de cet écart est généralement exprimée comme une fonction de deux quantités principales : *(i)* la taille de l'ensemble d'apprentissage et *(ii)* une mesure de complexité qui capture à quel point le modèle sur-apprend les données d'apprentissage. C'est le cas de mesures classiques telles que la dimension VC (Définition 2.3.2) ou la complexité de Rademacher (Définition 2.3.3) qui considèrent tout l'ensemble d'hypothèses. Comme nous l'avons rappelé dans le Chapitre 2, des mesures de complexité dépendantes de l'algorithme existent et permettent de ne considérer que l'hypothèse apprise avec l'algorithme, c'est le cas des paramètres liés à stabilité uniforme (Définition 2.3.5) ou à la robustesse algorithmique (Définition 2.3.6). D'un point de vue plus général, Lee et al. (2020, Prop. 1) ont établi un lien entre des mesures de complexité arbitraires et leur utilisation dans les bornes en généralisation. En effet, si l'on interprète leur résultat, la borne associée indique que l'écart

en généralisation est majoré, avec une grande probabilité, par une mesure de complexité définie par l'utilisateur, si cette mesure de complexité est proche de l'écart en généralisation. Cependant, cette borne est inapplicable, car elle repose sur une mesure de proximité entre la mesure de complexité et l'écart en généralisation.

Pour étudier les capacités en généralisation, une ligne de recherche récente dans le contexte des réseaux de neurones se consacre à l'étude empirique de différentes mesures de complexité pour identifier les mieux corrélées à l'écart en généralisation (Jiang et al., 2019 ; Dziugaite et al., 2020 ; Jiang et al., 2021). Ces résultats, extrêmement importants pour comprendre la généralisation, sont incomplets puisque ce sont des résultats uniquement empiriques. D'autre part, les bornes en généralisation de la littérature sont restreintes par le fait que l'utilisateur ne peut pas utiliser sa propre mesure de complexité dans la borne. En d'autres termes, à notre connaissance, il n'existe pas de bornes en généralisation permettant de considérer des mesures de complexité arbitraires identifiées comme bons indicateurs de l'écart en généralisation.

Dans ce chapitre, nous dérivons donc des bornes utilisant des mesures de complexité définies par l'utilisateur. Nous pensons que cette direction est d'un intérêt important pour faire progresser la compréhension de la généralisation, car l'écart en généralisation peut être borné théoriquement par un terme qui dépend d'une mesure spécifiée par l'utilisateur. Pour prouver de telles bornes, nous nous appuyons sur la borne PAC-Bayésienne désintégrée de Rivasplata et al. (2020), rappelée dans le Théorème 2.5.1. Comme vu dans le chapitre précédent, une telle borne permet de dériver des garanties théoriques pour des modèles qui dépendent du modèle tiré et de l'ensemble d'apprentissage, ce qui est peu courant en théorie de l'apprentissage statistique. Ainsi, nos résultats novateurs fournissent une base théorique aux nombreuses régularisations utilisées en pratique pour la sélection de modèles (*e.g.*, la régularisation L2).

9.2 Préliminaires

Pour faciliter la lecture de ce chapitre, nous rappelons quelques notions clés introduites précédemment et nécessaires à la bonne compréhension de notre contribution.

9.2.1 Contexte

Nous nous plaçons dans le cadre de la classification supervisée de la Section 2.5, où $\mathbb{X}$ est l'espace d'entrée et $\mathbb{Y}$ l'espace des étiquettes. Un exemple $(\mathbf{x}, y) \in \mathbb{X}\times\mathbb{Y}$ est tiré selon une distribution inconnue $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$. L'ensemble d'apprentissage $\mathbb{S}=\{(\mathbf{x}_i, y_i)\}_{i=1}^m$ contient m exemples tirés *i.i.d.* selon $\mathcal{D}$. Soit $\mathbb{H}$ un ensemble (potentiellement infini) d'hypothèses $h : \mathbb{X}\to\mathbb{Y}$. Étant donné $\mathbb{S}$, le but est de trouver $h \in \mathbb{H}$ qui minimise le risque réel $\mathrm{R}^{\ell}_{\mathcal{D}}(h) = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\, \ell(h, (\mathbf{x}, y))$. En pratique, puisque $\mathcal{D}$ est inconnue, nous estimons le risque réel par le risque empirique $\widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h) = \frac{1}{m}\sum_{i=1}^m \ell(h, (\mathbf{x}, y))$. Nous notons $\phi : [0, 1]^2 \to \mathbb{R}$ l'écart en généralisation habituellement défini par $\phi(\mathrm{R}^{\ell}_{\mathcal{D}}(h), \widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h)) = |\mathrm{R}^{\ell}_{\mathcal{D}}(h) - \widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h)|$.

Pour pouvoir incorporer des mesures de complexité arbitraires dans les bornes, nous nous basons sur le cadre des bornes PAC-Bayes désintégrées (rappelées dans la Section 9.2.2) où nous majorons l'écart en généralisation pour une hypothèse h tirée selon $\rho_{\mathbb{S}} \in \mathbb{M}(\mathbb{H})$ avec une fonction qui dépend d'une mesure de complexité arbitraire. Pour ce faire, nous avons besoin d'une connaissance *a priori* sur les hypothèses de $\mathbb{H}$ qui est modélisée par une distribution *prior* $\pi \in \mathbb{M}^*(\mathbb{H})$. Le but est alors d'apprendre, à partir de $\mathbb{S}$ et π, une distribution *posterior* $\rho_{\mathbb{S}}$

qui assigne une grande probabilité aux meilleures hypothèses de $\mathbb{H}$. L'hypothèse h est ensuite tirée selon $\rho_{\mathbb{S}}$ pour obtenir une garantie qui dépend d'une mesure de complexité arbitraire.

9.2.2 Rappel sur les bornes PAC-Bayes désintégrées

Comme expliqué dans le Chapitre 8, les bornes PAC-Bayésiennes désintégrées n'ont été que très peu utilisées dans la littérature. Elles n'ont reçu que récemment un intérêt pour dériver des bornes précises en pratique. Nous rappelons que les premières bornes PAC-Bayes désintégrées ont été introduites par Catoni (2007, Th. 1.2.7) et Blanchard et Fleuret (2007, Prop. 3.1). La forme de ces bornes est la suivante.

Définition 2.5.1 (Borne en généralisation PAC-Bayésienne désintégrée)**.** Soit une mesure de l'écart en généralisation $\phi : [0,1]^2 \to [0,1]$. Une borne PAC-Bayésienne désintégrée est définie telle que pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble d'hypothèses $\mathbb{H}$ avec $\ell : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y}) \to [0,1]$ une fonction perte, pour toute distribution *prior* $\pi \in \mathbb{M}^*(\mathbb{H})$, pour tout algorithme $A : (\mathbb{X}\times\mathbb{Y})^m \times \mathbb{M}^*(\mathbb{H}) \to \mathbb{M}(\mathbb{H})$, il existe une fonction $\Phi : \mathbb{M}(\mathbb{H})\times\mathbb{M}^*(\mathbb{H})\times]0,1] \to \mathbb{R}$ telle que pour tout $\delta \in]0,1]$ on a

$$\mathop{\mathbb{P}}_{\substack{\mathbb{S}\sim\mathcal{D}^m \\ h\sim\rho_{\mathbb{S}}}} \left[\phi(\mathsf{R}^{\ell}_{\mathcal{D}}(h), \widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h)) \;\leq\; \Phi\Big(\rho_{\mathbb{S}}, \pi, \delta\Big) \right] \;\geq\; 1-\delta,$$

où $\rho_{\mathbb{S}} = A(\mathbb{S}, \pi)$ est la sortie de l'algorithme déterministe A et $\phi()$ est, par exemple, $\phi(\mathsf{R}^{\ell}_{\mathcal{D}}(h), \widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h)) = |\mathsf{R}^{\ell}_{\mathcal{D}}(h) - \widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h)|$.

Étant donné $\mathbb{S} \sim \mathcal{D}^m$, nous pouvons apprendre la distribution $\rho_{\mathbb{S}}$ à l'aide de $\mathbb{S}$, puis nous tirons l'hypothèse h selon $\rho_{\mathbb{S}}$ pour obtenir une borne avec grande probabilité sur les choix aléatoires de $\mathbb{S}$ et h. Dans ce chapitre, nous nous focalisons principalement sur la borne de Rivasplata et al. (2020) rappelée ci-dessous.

Théorème 2.5.1 (Borne désintégrée générale de Rivasplata et al. (2020))**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble d'hypothèses $\mathbb{H}$, pour toute distribution *prior* $\pi \in \mathbb{M}^*(\mathbb{H})$, pour toute fonction mesurable $\varphi : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y})^m \to \mathbb{R}$, pour tout $\delta \in]0,1]$, pour tout algorithme $A : (\mathbb{X}\times\mathbb{Y})^m \times \mathbb{M}^*(\mathbb{H}) \to \mathbb{M}(\mathbb{H})$, on a

$$\mathop{\mathbb{P}}_{\substack{\mathbb{S}\sim\mathcal{D}^m \\ h\sim\rho_{\mathbb{S}}}} \left[\varphi(h,\mathbb{S}) \leq \underbrace{\ln\left[\frac{\rho_{\mathbb{S}}(h)}{\pi(h)}\right] + \ln\left[\frac{1}{\delta} \mathop{\mathbb{E}}_{\mathbb{S}'\sim\mathcal{D}^m} \mathop{\mathbb{E}}_{h'\sim\pi} \exp\left(\varphi(h',\mathbb{S}')\right)\right]}_{\Phi(\rho_{\mathbb{S}},\pi,\delta)} \right] \geq 1-\delta,$$

où $\rho_{\mathbb{S}} = A(\mathbb{S}, \pi)$ est la sortie de l'algorithme déterministe A.

Ici, $\varphi(h,\mathbb{S}) = m\,\phi(\mathsf{R}^{\ell}_{\mathcal{D}}(h), \widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h))$ mesure l'écart entre les risques réel et empirique. De plus, $\Phi(\rho_{\mathbb{S}}, \pi, \delta)$ est composé de deux termes : *(i)* la KL-divergence désintégrée $\ln \frac{\rho_{\mathbb{S}}(h)}{\pi(h)}$ qui mesure à quel point les distributions *prior* et *posterior* diffèrent pour un unique h, et *(ii)* le terme $\ln\Big[\frac{1}{\delta}\,\mathbb{E}_{\mathbb{S}'}\,\mathbb{E}_{h'}\exp\left(\varphi(h',\mathbb{S}')\right)\Big]$ qui est constant par rapport à $h \in \mathbb{H}$ et $\mathbb{S} \in (\mathbb{X}\times\mathbb{Y})^m$ et qui est généralement majoré pour instancier la borne. Dans ce qui suit, nous désignons la partie droite de la borne, $\Phi()$, comme la mesure de complexité. Cela contraste légèrement

avec la définition standard de la complexité où le terme *(ii)* (lié à δ et à m) n'est pas inclus. En fait, ce terme additionnel est constant par rapport à $h \sim \rho_{\mathbb{S}}$ et $\mathbb{S} \sim \mathcal{D}^m$.

Dans la borne du Théorème 2.5.1, le terme de complexité $\Phi()$ dépend de la KL-divergence désintégrée et souffre de quelques inconvénients. En effet, la complexité est imposée par le cadre et peut, en pratique, être sujette à une forte variance (voir le Chapitre 8). Cependant, il est important de préciser que la KL-divergence désintégrée a un avantage : elle dépend uniquement de l'hypothèse h et de $\mathbb{S}$, au lieu de dépendre de tout l'ensemble d'hypothèses (comme c'est souvent le cas, *e.g.*, avec la KL-divergence en PAC-Bayes ou avec la dimension VC). Cet avantage peut permettre une meilleure corrélation entre l'écart en généralisation et certaines mesures de complexité. Dans la section suivante, nous utilisons la KL-divergence désintégrée pour notre contribution : une borne générale qui dépend d'une mesure de complexité arbitraire.

9.3 Intégrer une mesure de complexité arbitraire dans une borne en généralisation

Nous commençons avec une brève présentation de notre résultat afin de donner quelques intuitions préliminaires et d'introduire la notion de distribution de Gibbs, qui est un élément clé de notre contribution. Nous présentons ensuite formellement notre résultat théorique dans la Section 9.3.3.

9.3.1 Notre résultat en quelques mots

Le principe pour introduire notre notion de mesure de complexité est de paramétrer la complexité à l'aide d'une fonction additionnelle "personnalisable" $\mu : \mathbb{H} \times (\mathbb{X} \times \mathbb{Y})^m \to \mathbb{R}$, que nous appelons *fonction paramétrique*. Grâce à la fonction $\mu()$, nous définissons $\Phi^r_\mu(h, \mathbb{S}, \delta)$ comme une fonction à valeurs réelles paramétrée par $\mu()$ et une variable aléatoire externe $r \sim \mathcal{R}$. Cette fonction prend en argument une hypothèse $h \in \mathbb{H}$, un échantillon d'apprentissage $\mathbb{S} \in (\mathbb{X} \times \mathbb{Y})^m$, et δ. La borne que nous dérivons dans le Théorème 9.3.1 dépend de $\Phi^r_\mu(h, \mathbb{S}, \delta)$.

Définition 9.3.1 (Borne en généralisation avec une mesure de complexité)**.** Soit l'écart en généralisation $\phi : \mathbb{R}^2 \to \mathbb{R}$. Étant donné un ensemble d'hypothèses $\mathbb{H}$, une fonction perte $\ell : \mathbb{H} \times (\mathbb{X} \times \mathbb{Y}) \to [0, 1]$ et une fonction paramétrique $\mu : \mathbb{H} \times (\mathbb{X} \times \mathbb{Y})^m \to \mathbb{R}$, Une borne en généralisation avec une mesure de complexité arbitraire est définie telle que pour toute distribution $\mathcal{D}$ sur $\mathbb{X} \times \mathbb{Y}$, pour toute distribution $\mathcal{R}$ représentant l'aléa, il existe une fonction à valeur réelle $\Phi^r_\mu : \mathbb{H} \times (\mathbb{X} \times \mathbb{Y})^m \times]0, 1] \to \mathbb{R}$ telle que pour tout $\delta \in]0, 1]$, on a

$$\mathop{\mathbb{P}}_{r \sim \mathcal{R},\ \mathbb{S} \sim \mathcal{D}^m,\ h \sim \rho_{\mathbb{S}}} \left[\phi(\mathsf{R}^\ell_{\mathcal{D}}(h), \widehat{\mathsf{R}}^\ell_{\mathbb{S}}(h)) \ \leq \ \Phi^r_\mu(h, \mathbb{S}, \delta) \right] \ \geq \ 1 - \delta, \tag{9.1}$$

où $\rho_{\mathbb{S}}$ est la distribution *posterior*.

L'astuce pour obtenir un tel résultat repose sur l'utilisation d'une distribution *posterior* $\rho_{\mathbb{S}}$ qui dépend de $\mu()$. Pour cela, nous définissons $\rho_{\mathbb{S}}$ comme la distribution de Gibbs

$$\rho_{\mathbb{S}}(h) \propto \exp\left[-\mu(h, \mathbb{S}) \right]. \tag{9.2}$$

Bien que cette équation puisse sembler restrictive, elle est en réalité suffisamment flexible pour représenter n'importe quelle fonction de densité de probabilité à condition qu'une mesure

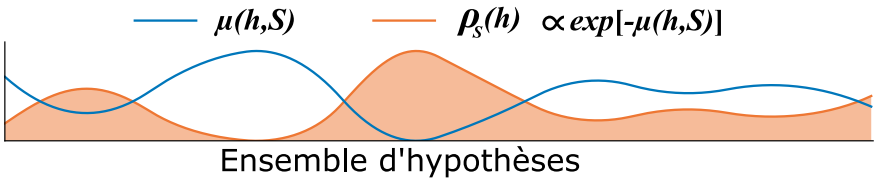


Figure 9.1. *Illustration du comportement de la distribution de Gibbs $\rho_{\mathbb{S}}$ avec une fonction paramétrique $\mu()$. L'axe des abscisses représente un ensemble d'hypothèse (continu) et l'axe des ordonnées représente les valeurs de $\rho_{\mathbb{S}}$ et $\mu()$. La distribution $\rho_{\mathbb{S}}$ affecte une plus grande probabilité aux hypothèses avec une valeur de $\mu()$ faible.*

de complexité pertinente soit sélectionnée. Par exemple, soit $\rho'_{\mathbb{S}}$ une distribution sur $\mathbb{H}$, *e.g.*, une distribution Gaussienne ou de Laplace, alors en choisissant $\mu(h,\mathbb{S}) = -\ln \rho'_{\mathbb{S}}(h)$, on peut retrouver la distribution $\rho'_{\mathbb{S}}$. De plus, du point de vue de l'optimisation, la distribution de Gibbs $\rho_{\mathbb{S}}$ est intéressante : étant donné un ensemble d'apprentissage fixe $\mathbb{S}$, une hypothèse h a plus de chances d'être tirée lorsque $\mu(h,\mathbb{S})$ est faible (voir la Figure 9.1). En fait, la fonction $h \mapsto \mu(h,\mathbb{S})$ peut être considérée comme une fonction objectif. Par exemple, pour minimiser le risque réel $\mathrm{R}^{\ell}_{\mathcal{D}}(h)$, dans l'idéal, on voudrait définir $\mu(h,\mathbb{S}) = \alpha \mathrm{R}^{\ell}_{\mathcal{D}}(h)$, qui est associée à une distribution de Gibbs échantillonnant des hypothèses avec un faible risque réel et se concentrant autour des faibles risques lorsque $\alpha \in \mathbb{R}^*_+$ augmente. Cependant, comme le risque réel est inconnu, il doit être remplacé par une fonction $\mu()$ calculable. Par exemple, la fonction $\mu()$ peut correspondre au risque empirique, défini par $\mu(h,\mathbb{S}) = \alpha \widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h)$.

9.3.2 À propos de la distribution de Gibbs

Dans cette section, nous souhaitons mettre en lumière deux grandes lignes de travaux liées à notre cadre : *(i)* l'utilisation de la distribution de Gibbs en théorie PAC-Bayes "classique" et *(ii)* le lien entre cette distribution et l'optimisation.

Distribution de Gibbs en PAC-Bayes. La distribution de Gibbs a commencé à être étudiée en PAC-Bayes par Catoni (2004, 2007). De plus, Alquier et al. (2016, Th. 4.2 et 4.3) ont dérivé des bornes en généralisation PAC-Bayésiennes avec la distribution de Gibbs de l'Équation (9.2) avec $\mu(h,\mathbb{S}) = 0$ comme *posterior*. Cependant, leurs théorèmes analysent l'espérance des risques réels $\mathbb{E}_{h\sim\rho_{\mathbb{S}}} \mathrm{R}^{\ell}_{\mathcal{D}}(h)$ alors que nous sommes intéressés par une *unique* hypothèse h tirée selon $\rho_{\mathbb{S}}$. En outre, leurs bornes mettent en jeu la KL-divergence, non calculable et dépendante de l'ensemble des hypothèses, entre la distribution de Gibbs et une distribution *prior*. La KL-divergence doit donc être majorée pour permettre l'instanciation de la borne en pratique. Comme nous le verrons dans la suite, les bornes du Théorème 9.3.1 et du Corollaire 9.3.1 n'ont pas cet inconvénient puisqu'elles requièrent uniquement de connaître l'expression de la densité (à une constante de normalisation prêt) pour $h \sim \rho_{\mathbb{S}}$ et $h' \sim \pi$.

Lien entre optimisation et distribution de Gibbs. Étant donné une fonction paramétrique différentiable définie par $\mu(h,\mathbb{S}) = \alpha\nu(h,\mathbb{S})$ (avec α un paramètre de concentration), la distribution de Gibbs associée peut être reliée à l'algorithme *Stochastic Gradient Langevin Dynamics* (SGLD, Welling et Teh, 2011) qui apprend une hypothèse $h \in \mathbb{H}$ en exécutant des itérations de la forme

$$h_t \longleftarrow h_{t-1} - \eta \nabla \nu(h_t,\mathbb{S}) + \sqrt{\frac{2\eta}{\alpha}}\epsilon_t, \quad \text{avec} \quad \epsilon_t \sim \mathcal{N}(\mathbf{0},\mathbf{I}_D), \tag{9.3}$$

où h_t est l'hypothèse apprise à l'itération t, le paramètre η est le *learning rate* et α est le paramètre de concentration de la distribution de Gibbs. Lorsque α augmente, le bruit ϵ_t a

moins d'influence sur l'itération suivante obtenue avec SGLD, car $\sqrt{\frac{2\eta}{\alpha}}\epsilon_t$ diminue, ce qui permet de mieux minimiser la fonction ν. De plus, lorsque le *learning rate* η tend vers zéro, SGLD devient un processus en temps continu appelé diffusion de Langevin, défini par l'équation différentielle stochastique dans l'Équation (9.4). En effet, l'Équation (9.3) peut être vue comme une discrétisation de Euler-Maruyama (voir Raginsky et al., 2017) de l'Équation (9.4) définie pour $t \geq 0$ par

$$dh_t = -\nabla\nu(h_t, \mathbb{S})dt + \sqrt{\frac{2}{\alpha}}\mathbf{B}(t), \tag{9.4}$$

où $\mathbf{B}(t)$ est le mouvement Brownien. Sous certaines hypothèses sur la fonction n, Chiang et al. (1987) ont montré que la distribution invariante de la diffusion de Langevin est la distribution de Gibbs $\rho_{\mathbb{S}}$ avec $\mu(h, \mathbb{S}) = \alpha\nu(h, \mathbb{S})$.

9.3.3 Une borne en généralisation avec mesure de complexité

La définition de $\rho_{\mathbb{S}}$ nous permet d'énoncer notre résultat principal : une borne sur l'écart en généralisation qui est valide pour des hypothèses tirées selon la distribution *posterior* $\rho_{\mathbb{S}}(h) \propto \exp\left[-\mu(h, \mathbb{S})\right]$

Théorème 9.3.1 (Borne en généralisation avec mesure de complexité)**.** Soit une fonction perte $\ell : \mathbb{H} \times (\mathbb{X}\times\mathbb{Y}) \to [0,1]$ et l'écart en généralisation $\phi : [0,1]^2 \to \mathbb{R}$. Pour tout $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout $\mathbb{H}$, pour toute distribution $\pi \in \mathbb{M}^*(\mathbb{H})$, pour tout $\mu : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y})^m \to \mathbb{R}$, pour tout $\delta \in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\substack{h'\sim\pi \\ \mathbb{S}\sim\mathcal{D}^m \\ h\sim\rho_{\mathbb{S}}}}\left[\phi(\mathsf{R}^\ell_{\mathcal{D}}(h), \widehat{\mathsf{R}}^\ell_{\mathbb{S}}(h)) \leq \underbrace{\mu(h', \mathbb{S}) - \mu(h, \mathbb{S}) + \ln\frac{\pi(h')}{\pi(h)} + \ln\left(\frac{4}{\delta^2}\mathop{\mathbb{E}}_{\mathbb{S}'\sim\mathcal{D}^m}\mathop{\mathbb{E}}_{g\sim\pi}\exp\left[\phi(\mathsf{R}^\ell_{\mathcal{D}}(g), \widehat{\mathsf{R}}^\ell_{\mathbb{S}'}(g))\right]\right)}_{\Phi^r_\mu(h,\mathbb{S},\delta)}\right] \geq 1-\delta.$$

La borne $\Phi^r_\mu(h, \mathbb{S}, \delta)$ dépend de trois termes : *(i)* la différence $\mu(h', \mathbb{S}) - \mu(h, \mathbb{S})$, *(ii)* le log ratio $\ln(\pi(h')/\pi(h))$, et *(iii)* une constante $\ln[\frac{4}{\delta^2}\,\mathbb{E}_{\mathbb{S}'}\,\mathbb{E}_g \exp[\phi(\mathsf{R}^\ell_{\mathcal{D}}(g), \widehat{\mathsf{R}}^\ell_{\mathbb{S}'}(g))]]$. Comparé au Théorème 2.5.1, nous majorons la KL-divergence désintégrée $\ln\frac{\rho_{\mathbb{S}}(h)}{\pi(h)}$ par la différence $\mu(h', \mathbb{S}) - \mu(h, \mathbb{S})$ et le log ratio $\ln(\pi(h')/\pi(h))$. L'avantage de ces deux termes est qu'ils sont facilement calculables, tant que nous pouvons calculer $\mu(h', \mathbb{S})$, $\mu(h, \mathbb{S})$ et la densité de π (à une constante de normalisation près). Cela contraste avec le résultat de Lee et al. (2020), qui est une borne valable pour tout $\epsilon > 0$ de la forme :

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m, h\sim\rho_{\mathbb{S}}}\left[|\mathsf{R}^\ell_{\mathcal{D}}(h) - \widehat{\mathsf{R}}^\ell_{\mathbb{S}}(h)| \leq \mu(h, \mathbb{S}) + \epsilon\right] \geq 1 - \delta'(\epsilon),$$

où $\delta'(\epsilon)$ dépend de la distribution inconnue $\mathcal{D}$, de n ensemble d'apprentissage $\mathbb{S}_1, \ldots, \mathbb{S}_n$ et de n hypothèses $h_1 \sim \rho_{\mathbb{S}_1}, \ldots, h_n \sim \rho_{\mathbb{S}_n}$. Bien que dans ce résultat il n'y ait pas de restriction sur la forme de la distribution $\rho_{\mathbb{S}}$, sa dépendance en $\mathcal{D}$ rend le terme $\delta'(\epsilon)$ non calculable (contrairement à notre borne).

Le terme *(iii)* est en général négligeable comparé à *(i)* et *(ii)*, et peut être majoré quand l'écart en généralisation est instancié. Pour obtenir une borne qui converge quand m augmente, il est suffisant de fixer $\phi()$ comme une fonction de m. L'importance du terme *(ii)*

dépend de l'instanciation de π. Enfin, *(i)* dépend du choix de $\mu()$ qui a une forte influence sur l'hypothèse tirée $h\sim\rho_{\mathbb{S}}$ et donc sur l'écart $\phi(\mathrm{R}^{\ell}_{\mathcal{D}}(h),\widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h))$. Par exemple, si $\mu(h,\mathbb{S})=0$ alors la différence $\mu(h',\mathbb{S})-\mu(h,\mathbb{S})=0$, mais la distribution $\rho_{\mathbb{S}}$ est dans ce cas uniforme, ce qui empêche de tirer une hypothèse qui minimise le risque réel $\mathrm{R}^{\ell}_{\mathcal{D}}(h)$. Il y a donc un compromis à trouver pour minimiser cette différence et tirer une hypothèse qui minimise l'écart en généralisation $\phi(\mathrm{R}^{\ell}_{\mathcal{D}}(h),\widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h))$ et $\mathrm{R}^{\ell}_{\mathcal{D}}(h)$. Nous verrons par la suite comment instancier la fonction paramétrique $\mu()$. Notons qu'il est possible de retrouver des bornes classiques basées sur la convergence uniforme ou dépendantes d'un algorithme.

Pour obtenir une borne utilisable en pratique, le défi restant est de trouver une borne supérieure pour $\ln[\frac{4}{\delta^2}\,\mathbb{E}_{\mathbb{S}'}\,\mathbb{E}_g\exp[\phi(\mathrm{R}^{\ell}_{\mathcal{D}}(g),\widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(g))]]$ et $\mathbb{E}_{h'\sim\pi}\ln\frac{\pi(h')}{\pi(h)}$. En guise d'illustration, nous présentons dans le corollaire suivant une instanciation du Théorème 9.3.1 avec l'écart en généralisation $\phi(\mathrm{R}^{\ell}_{\mathcal{D}}(h),\widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h))=m\,\mathrm{kl}[\widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h)\|\mathrm{R}^{\ell}_{\mathcal{D}}(h)]$ où π est la distribution uniforme sur un ensemble d'hypothèses bornées $\mathbb{H}$.

Corollaire 9.3.1 (Borne en généralisation pratique avec mesure de complexité)**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble d'hypothèses bornées $\mathbb{H}$, pour toute fonction perte $\ell:\mathbb{H}\times(\mathbb{X}\times\mathbb{Y})\to[0,1]$, étant donné la distribution *prior* uniforme π sur $\mathbb{H}$, pour tout $\mu:\mathbb{H}\times(\mathbb{X}\times\mathbb{Y})^m\to\mathbb{R}$, pour tout $\delta\in]0,1]$, on a

$$\mathop{\mathbb{P}}_{\substack{h'\sim\pi\\ \mathbb{S}\sim\mathcal{D}^m\\ h\sim\rho_{\mathbb{S}}}}\left[\mathrm{kl}\left[\widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h)\|\mathrm{R}^{\ell}_{\mathcal{D}}(h)\right]\leq\frac{1}{m}\left[\mu(h',\mathbb{S})-\mu(h,\mathbb{S})+\frac{8\sqrt{m}}{\delta^2}\right]_+\right]\geq 1-\delta. \tag{9.5}$$

Le Corollaire 9.3.1 fournit une borne sur $\mathrm{kl}[\widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h)\|\mathrm{R}^{\ell}_{\mathcal{D}}(h)]$ où tous les termes sauf $\mathrm{R}^{\ell}_{\mathcal{D}}(h)$ sont calculables. Pour évaluer l'Équation (9.5), nous pouvons réarranger les termes pour obtenir une borne en généralisation sur le risque réel $\mathrm{R}^{\ell}_{\mathcal{D}}(h)$. En effet, on a

$$\mathrm{R}^{\ell}_{\mathcal{D}}(h)\leq\overline{\mathrm{kl}}\left(\widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h)\;\middle|\;\frac{1}{m}\left[\mu(h',\mathbb{S})-\mu(h,\mathbb{S})+\frac{8\sqrt{m}}{\delta^2}\right]_+\right). \tag{9.6}$$

Ce résultat est utilisé dans la Section 9.4 pour illustrer les garanties en généralisation avec différentes valeurs de fonctions paramétriques $\mu()$. Dans certains cas triviaux, le taux de convergence peut être arbitraire, *e.g.*, lorsque $\mu(h,\mathbb{S})=m\widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h)$. Par exemple, pour un risque empirique élevé[1] $\widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h')$, la partie droite de l'Équation (9.5) se simplifie en $\Phi^{h'}_{\mu}(h,\mathbb{S},\delta)=[(\widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h')-\widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h))+\frac{1}{m}\ln(2\sqrt{m}/\delta)]_+$ et est élevé, pour tout m. Pour que la borne soit significative, nous devons fixer $\mu()$ de sorte que la distribution $\rho_{\mathbb{S}}$ permette de tirer un h qui minimise le risque empirique $\widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h)$ et l'écart en généralisation, et nous voulons que la mesure de complexité $\Phi^{h'}_{\mu}(h,\mathbb{S},\delta)$ soit précise (avec $h'\sim\pi$).

La précision des bornes peut être améliorée avec un *prior* dépendant des données comme souvent en PAC-Bayes (*e.g.*, Parrado-Hernández et al., 2012b ; Dziugaite et al., 2021 ; Pérez-Ortiz et al., 2021). Nous suivons une stratégie similaire en définissant le *prior* π par

$$\pi(h)\propto\exp\left|-\omega(h)\right|, \tag{9.7}$$

où $\omega:\mathbb{H}\to\mathbb{R}$ peut dépendre de $\mathcal{D}$. Ainsi, π peut dépendre d'un ensemble $\mathbb{S}'\sim\mathcal{D}^{m'}$. Dans ce cas, nous obtenons le corollaire suivant.

1. Lorsque h' est tiré selon la loi uniforme sur $\mathbb{H}$, il est fréquent que $\widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h')$ soit élevé.

Corollaire 9.3.2 (Borne en généralisation pratique avec mesure de complexité et un *prior* dépendant des données)**.** Pour toute distribution $\mathcal{D}$ sur $\mathbb{X}\times\mathbb{Y}$, pour tout ensemble d'hypothèses $\mathbb{H}$, pour toute fonction perte $\ell : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y})\to[0,1]$, pour toute fonction $\mu : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y})^m\to\mathbb{R}$, pour tout $\omega : \mathbb{H}\to\mathbb{R}$, pour tout $\delta\in]0,1]$,

$$\mathop{\mathbb{P}}_{\substack{\mathbb{S}\sim\mathcal{D}^m\\ h'\sim\pi\\ h\sim\rho_{\mathbb{S}}}}\left[\mathrm{kl}\left[\widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h)\|\mathrm{R}^{\ell}_{\mathcal{D}}(h)\right]\leq\frac{1}{m}\left([\mu(h',\mathbb{S})-\omega(h')]-[\mu(h,\mathbb{S})-\omega(h)]+\ln\frac{8\sqrt{m}}{\delta^2}\right)\right]\geq 1-\delta, \quad (9.8)$$

avec π définie dans l'Équation (9.7).

9.4 Utilisation d'une complexité arbitraire en pratique

Le Corollaire 9.3.1 n'est pas utilisable tel quel : il reste à s'occuper du tirage de h selon la distribution de Gibbs $\rho_{\mathbb{S}}$ de l'Équation (9.2). Nous nous attaquons à ce tirage dans la Section 9.4.1. Ensuite, nous utilisons en pratique la solution proposée pour évaluer la borne.

9.4.1 Échantillonnage à partir de la distribution de Gibbs

Le tirage aléatoire de h selon la distribution de Gibbs de l'Équation (9.2) est une tâche complexe. Naïvement, nous devons évaluer la fonction $h\mapsto -\alpha\widehat{\mathrm{R}}^{\ell}_{\mathbb{S}}(h)-\mu(h,\mathbb{S})$ pour tout $h\in\mathbb{H}$, ce qui n'est pas possible lorsque $\mathbb{H}$ est infini ou trop grand. Nous proposons une solution à ce problème pour les modèles sur-paramétrés que nous avons considérés pour les expériences menées. Soit un $\mathbb{H}$ un ensemble d'hypothèses, $h_{\mathbf{w}}$ paramétrées par $\mathbf{w}\in\mathbb{R}^D$ une distribution manipulable notée $\mathrm{P}^{\mathbf{w}}_{\mathbb{U}}$ (*e.g.*, une distribution gaussienne) telle que sa densité approxime celle de $\rho_{\mathbb{S}}$. Dans ce cas, pour apprendre une telle distribution auxiliaire, nous proposons dans l'Algorithme 9.1 une version stochastique de l'algorithme *Metropolis Adjusted Langevin* (MALA, Besag, 1994)[2]. Son but est de générer des tirages selon $\rho_{\mathbb{S}}$ en affinant itérativement la distribution définie par

$$\mathrm{P}^{\mathbf{w}}_{\mathbb{U}}=\mathcal{N}\left(\mathbf{w}-\eta\nabla\left[\mathrm{R}^{\ell}_{\mathbb{U}}(\mathbf{w})+\frac{1}{\alpha}\,\mu(\mathbf{w},\mathbb{U})\right],\ \tfrac{2\eta}{\alpha}\mathbf{I}\right), \quad (9.9)$$

où $\mathrm{R}^{\ell}_{\mathbb{U}}(\mathbf{w})=\mathbb{E}_{(\mathbf{x},y)\in\mathbb{U}}\,\ell(h_{\mathbf{w}},(\mathbf{x},y))$ est le risque empirique sur le mini-lot $\mathbb{U}\subseteq\mathbb{S}$ et $\ell()$ est une fonction perte. Concrètement, nous initialisons les paramètres $\mathbf{w}$ du modèle avec la sortie d'un algorithme d'optimisation (*Vanilla SGD* dans notre cas). Nous les affinons ensuite : à chaque itération de l'Algorithme 9.1, étant donné les poids courants $\mathbf{w}$ et un mini-lot $\mathbb{U}\subseteq\mathbb{S}$ (Ligne 2), nous tirons un vecteur candidat $\mathbf{w}'$ (Ligne 3) selon la distribution $\mathrm{P}^{\mathbf{w}}_{\mathbb{U}}$. Puis (Ligne 4 à 7) nous décidons de rejeter ou d'accepter ce nouveau candidat pour qu'il devienne nos poids courants $\mathbf{w}$, en vérifiant son ratio $\tau=\min\left(1,\frac{\rho_{\mathbb{U}}(\mathbf{w}')\mathrm{P}^{\mathbf{w}'}_{\mathbb{U}}(\mathbf{w})}{\rho_{\mathbb{U}}(\mathbf{w})\mathrm{P}^{\mathbf{w}}_{\mathbb{U}}(\mathbf{w}')}\right)$ qui doit être supérieur à une valeur de contrôle u tirée selon la distribution uniforme sur $[0,1]$. Sous l'hypothèse que $\rho_{\mathbb{S}}$ est absolument continue par rapport à $\mathrm{P}^{\mathbf{w}}_{\mathbb{S}}$ (voir Chib et Greenberg, 1995, pour les détails), lorsque le nombre d'itérations tend vers $+\infty$ et lorsque $\mathbb{U}=\mathbb{S}$, le vecteur $\mathbf{w}$ retourné est tiré selon $\rho_{\mathbb{S}}$ (Smith et Roberts, 1993). Cette hypothèse requière que la distribution $\mathrm{P}^{\mathbf{w}}_{\mathbb{S}}$ ait une densité strictement positive lorsque $\rho_{\mathbb{S}}$ est, elle aussi, strictement positive (voir Chib et Greenberg, 1995).

2. Voir Chib et Greenberg (1995) pour une introduction à l'algorithme *Metropolis-Hastings* sur lequel MALA est basé.

Algorithme 9.1 Stochastic MALA

Entrées : Ensemble d'apprentissage $\mathbb{S}$, poids $\mathbf{w}$, fonction $\mu()$, fonction perte $\ell()$, nombre d'itération T, *learning rate* η, paramère α

pour $t \leftarrow 1 \dots T$ **faire**
  $\mathbb{U} \leftarrow$ tirage sans replacement d'un mini-lot depuis $\mathbb{S}$
  $\mathbf{w}' \leftarrow$ tirage selon a distribution $\mathrm{P}^{\mathbf{w}}_{\mathbb{U}}$
  $\tau \leftarrow \min\left(1, \frac{\rho_{\mathbb{U}}(\mathbf{w}')\mathrm{P}^{\mathbf{w}'}_{\mathbb{U}}(\mathbf{w})}{\rho_{\mathbb{U}}(\mathbf{w})\mathrm{P}^{\mathbf{w}}_{\mathbb{U}}(\mathbf{w}')}\right)$
  $u \leftarrow$ Tirage selon $\mathrm{Uni}(0,1)$
  si $u \leq \tau$ **alors**
    $\mathbf{w} \leftarrow \mathbf{w}'$
retourner $\mathbf{w}$

9.4.2 Résumé des expériences

Nous avons étudié la précision des bornes des Corollaires 9.3.1 et 9.3.2 sur deux jeux de données : MNIST (LeCun et al., 1998) et FashionMNIST (Xiao et al., 2017). Plus précisément, nous avons considéré les bornes sur le risque réel et le risque empirique associés à la fonction de perte 01.

Expériences sur le risque empirique. Nous avons comparé nos bornes avec des bornes similaires issues de la littérature avec la distribution de Gibbs $\rho_{\mathbb{S}}$ définie avec la fonction paramétrique $\mu(h, \mathbb{S}) = \alpha \mathsf{R}^{\ell'}_{\mathbb{S}}(h)$. En effet, cette distribution de Gibbs a déjà été étudiée pour des bornes PAC-Bayes classiques et désintégrées, mais a conduit à des bornes incalculables. Nous avons proposé une adaptation de ces bornes pour les rendre calculables et s'y comparer. Plus précisément, nous comparons nos bornes à la borne suivante (similaire à celle de Lever et al., 2013) ; avec une probabilité d'au moins $1-\delta$, on a

$$\mathrm{kl}[\widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h) \| \mathsf{R}^{\ell}_{\mathcal{D}}(h)] \leq \frac{1}{m}\left[\frac{\alpha^2}{8m} + \sqrt{\frac{\alpha^2}{2m}\ln\frac{6\sqrt{m}}{\delta}} + \ln\frac{6\sqrt{m}}{\delta}\right]. \tag{9.10}$$

Nous adaptons également la technique de preuve de Dziugaite et Roy (2018) pour obtenir avec une probabilité d'au moins $1-\delta$

$$\mathrm{kl}[\widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h) \| \mathsf{R}^{\ell}_{\mathcal{D}}(h)] \leq \frac{1}{m}\left(\alpha\left[\mathsf{R}^{\ell'}_{\mathbb{S}}(h') - \mathsf{R}^{\ell'}_{\mathbb{S}}(h)\right] + \alpha'\left[\mathsf{R}^{\ell'}_{\mathbb{S}}(h) - \mathsf{R}^{\ell'}_{\mathbb{S}}(h')\right] + 2\alpha' + \ln\tfrac{8\sqrt{m}}{\delta^2}\right), \tag{9.11}$$

où $h' \sim \pi$ et $\pi(h) \propto \exp[-\alpha' \mathsf{R}^{\ell'}_{\mathbb{S}}(h)]$. Pour le Corollaire 9.3.2, nous définissons le *prior* π avec la fonction $\omega(h) = \alpha \mathsf{R}^{\ell'}_{\mathbb{S}'}(h)$ où $\mathbb{S}'$ satisfait le ratio $\frac{m'}{m+m'} = 0.5$. Pour toutes les bornes, α et α' sont uniformément espacés sur une échelle logarithmique entre $\sqrt{m}$ et m.

Comme attendu, avec le *prior* qui minimise la borne, nous avons observé que les écart-types sont faibles uniquement pour de grandes valeurs de α, car ce paramètre contrôle la concentration de la distribution de Gibbs. Une valeur de α grande a tendance à impliquer un risque en test plus faible. Cependant, les bornes deviennent grandes à mesure que α augmente sauf pour la borne du Corollaire 9.3.2. Ce comportement est attendu dans le cas de l'Équation (9.10) puisque la borne augmente lors α diminue. La borne du Corollaire 9.3.1 est grande quand $\alpha[\mathsf{R}^{\ell'}_{\mathbb{S}}(h') - \mathsf{R}^{\ell'}_{\mathbb{S}}(h)]$ est grand. C'est le cas puisque $\mathsf{R}^{\ell'}_{\mathbb{S}}(h')$ est élevé, étant donné que h' est tiré selon $\rho_{\mathbb{S}}(h) \propto \exp[-\alpha \mathsf{R}^{\ell'}_{\mathbb{S}}(h)]$. Le même phénomène se produit avec l'Équation (9.11) car $\mathsf{R}^{\ell'}_{\mathbb{S}}(h')$ est grand quand α' est petit, *i.e.*, la concentration n'est pas suffisante pour minimiser le risque empirique. La précision du Corollaire 9.3.2 vient du fait

que le risque empirique de $h' \sim \pi$ et celui de $h \sim \rho_{\mathbb{S}}$ sont faibles, tout comme la borne lorsque les risques $\mathsf{R}^{\ell'}_{\mathbb{S}}(h')$ et $\mathsf{R}^{\ell'}_{\mathbb{S}'}(h)$ sont également faibles. De plus, pour de petites valeurs de α, les risques en test et les valeurs des bornes sont plus élevées par rapport aux autres. Cela est dû au fait que nous utilisons la moitié des données ($\frac{m'}{m+m'}{=}0.5$) pour apprendre le *prior* dépendant des données. En effet, α est deux fois plus petit que pour les autres bornes, ce qui rend les valeurs plus élevées puisque la distribution de Gibbs est moins concentrée.

Expérience pour des risques régularisés. Pour rendre plus précises les bornes des Corollaires 9.3.1 et 9.3.2, nous avons comparé différents risques empiriques régularisés avec les normes optimisables étudiées par Jiang et al. (2019, Sec. C). L'idée, qui semble naturelle, est de sélectionner une hypothèse avec un compromis faible entre son risque empirique et une norme. Contre toute attente, nous avons observé que régulariser le risque empirique avec une fonction paramétrique n'aide pas à rendre plus précises les bornes. Cela suggère donc que les normes ne sont pas de bons prédicteurs/estimateurs de l'écart en généralisation.

Par souci de clarification, les risques empiriques régularisés avec les normes optimisables que nous avons considérés sont

- $\textsc{DistFro}^{\boldsymbol{R}}_{\beta}(h, \mathbb{S}) = \alpha\Big[\beta \mathsf{R}^{\ell'}_{\mathbb{S}}(h) + \bar{\beta}\textsc{DistFro}(h, \mathbb{S})\Big]$, où $\textsc{DistFro}(h, \mathbb{S}) = \sum_{i=1}^{L} \|\mathbf{w}_i - \mathbf{v}_i\|_2$,
- $\textsc{DistL}_2{}^{\boldsymbol{R}}_{\beta}(h, \mathbb{S}) = \alpha\Big[\beta \mathsf{R}^{\ell'}_{\mathbb{S}}(h) + \bar{\beta}\textsc{DistL}_2(h, \mathbb{S})\Big]$, où $\textsc{DistL}_2(h, \mathbb{S}) = \|\mathbf{w} - \mathbf{v}\|_2$,
- $\textsc{ParNorm}^{\boldsymbol{R}}_{\beta}(h, \mathbb{S}) = \alpha\Big[\beta \mathsf{R}^{\ell'}_{\mathbb{S}}(h) + \bar{\beta}\textsc{ParNorm}(h, \mathbb{S})\Big]$, où $\textsc{ParNorm}(h, \mathbb{S}) = \sum_{i=1}^{L} \|\mathbf{w}_i\|_2^2$,
- $\textsc{PathNorm}^{\boldsymbol{R}}_{\beta}(h, \mathbb{S}) = \alpha\Big[\beta \mathsf{R}^{\ell'}_{\mathbb{S}}(h) + \bar{\beta}\textsc{PathNorm}(h, \mathbb{S})\Big]$, où $\textsc{PathNorm}(h, \mathbb{S}) = \sum_{y \in \mathcal{Y}} h_{\mathbf{w}^2}(\mathbf{1})[y]$,
- $\textsc{SumFro}^{\boldsymbol{R}}_{\beta}(h, \mathbb{S}) = \alpha\Big[\beta \mathsf{R}^{\ell'}_{\mathbb{S}}(h) + \bar{\beta}\textsc{SumFro}(h, \mathbb{S})\Big]$, où $\textsc{SumFro}(h, \mathbb{S}) = L\left[\prod_{i=1}^{L} \|\mathbf{w}_i\|_2^2\right]^{\frac{1}{L}}$,

avec $\bar{\beta} = 1 - \beta$.

Expériences sur les complexités neuronales. À la lumière de nos résultats, nous avons également étudié le comportement de nos bornes lorsqu'elles sont calculées avec un meilleur prédicteur de l'écart en généralisation. En effet, nous nous sommes idéalement intéressés à concentrer la mesure de probabilité associée à $\rho_{\mathbb{S}}$ sur les hypothèses avec un faible écart en généralisation. Pour ce faire, la fonction paramétrique pour $\rho_{\mathbb{S}}$ peut dépendre d'une estimation de cet écart. Dans cette section, nous considérons la borne du Corollaire 9.3.1 (avec *prior* uniforme) et nous étudions les fonctions paramétriques $\mu()$ suivantes :

$$\mu(h, \mathbb{S}) = f^{\boldsymbol{D}}(h, \mathbb{S}) = \alpha |f(h, \mathbb{S}) - f(h_{\mathsf{SGD}}, \mathbb{S})|,$$

où $f \in \{\textsc{DistFro}, \textsc{DistL}_2, \textsc{ParNorm}, \textsc{PathNorm}, \textsc{SumFro}\}$ et $\alpha = m$, et où h_{SGD} est obtenu par descente de gradient stochastique (SGD). Ce choix particulier de $\mu()$ permet de tirer des hypothèses proches de la valeur de la fonction $f()$ évaluée sur h_{SGD}. Nous évaluons également une fonction paramétrique $\textsc{Neural}^{\boldsymbol{D}}$ constituée d'un réseau de neurones appris pour prédire l'écart en généralisation. Plus précisément, nous apprenons la fonction $\textsc{Neural}(h, \mathbb{S})$ qui devient la sortie d'un réseau *feed-forward* (appris à partir de $\mathbb{S}$), prenant les paramètres $\mathbf{w}$ du modèle h et renvoyant un réel positif qui doit représenter l'écart en généralisation. La fonction $\textsc{Neural}^{\boldsymbol{D}}$ compare donc la sortie du réseau *feed-forward* neural associé à $h \sim \rho_{\mathbb{S}}$ et h_{SGD}. Notons que l'apprentissage d'un réseau de neurones pour prédire l'écart en généralisation a été proposé par Lee et al. (2020).

Nous avons observé que les bornes obtenues dans cette situation se comportent différemment des bornes avec une mesure de complexité basée sur une norme. En effet, les valeurs moyennes des bornes pour les mesures basées sur les normes sont toutes non informatives (*i.e.*, elles sont supérieures à 1). Dans ce cas, les fonctions paramétriques évaluées sur h sont proches de zéro, alors que l'évaluation sur h' est grande, impliquant que les bornes sont non informatives. Cela met en évidence un inconvénient des études empiriques de Jiang et al. (2019) et Dziugaite et al. (2020) qui portent sur la corrélation entre les normes et l'écart en généralisation sur des réseaux de neurones *appris*. Or, considérer une norme comme approximation de l'écart en généralisation est impossible dans ce cas. En effet, redimensionner les poids des réseaux (par un scalaire) donne exactement les mêmes prédictions et conserve le même écart en généralisation tout en modifiant la norme ; cela est dû à l'utilisation de fonctions d'activation homogènes non négatives, telles que la (Leaky) RELU standard (*e.g.*, Neyshabur et al., 2015 ; Dinh et al., 2017). En revanche, les fonctions paramétriques Neural et Neural^{D} fournissent des bornes précisent et sont proches des bornes idéales. Cela illustre que l'apprentissage d'une fonction paramétrique (et donc d'une mesure de complexité) peut aider à obtenir des bornes en généralisation plus précises. Notons que les bornes avec Neural et Neural^{D} sont précises même sans *prior* dépendant des données, pourtant généralement nécessaire pour obtenir des bornes précises pour les réseaux de neurones (*e.g.*, Dziugaite et Roy, 2017 ; Dziugaite et al., 2021 ; Pérez-Ortiz et al., 2021 ; Viallard et al., 2024b). Ce résultat est prometteur et encourageant pour s'affranchir du besoin d'un *prior* dépendant des données en PAC-Bayes.

9.5 Retrouver des bornes en convergence uniforme et dépendantes d'un algorithme

"La boucle est bouclée"

Cette section marque en quelque sorte la fin de ce manuscrit puisque les résultats de ce chapitre sont suffisamment généraux pour retrouver deux types de bornes classiques rappelées dans le Chapitre 2 dans les Sections 2.3.1 et 2.3.2. Les Corollaires 9.5.1 et 9.5.3 ci-dessous ne présentent pas de nouvelles bornes, mais montrent comment obtenir des bornes existantes en intégrant une mesure de complexité spécifique $\mu()$.

Borne en généralisation basée sur la convergence uniforme. À partir du Théorème 9.3.1, nous pouvons obtenir la borne générale en convergence uniforme suivante.

Corollaire 9.5.1 (Borne en convergence uniforme)**.** Soit $\ell : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y})\to[0,1]$ une fonction perte et $\phi : [0,1]^2\to\mathbb{R}$ l'écart en généralisation. On suppose qu'il existe une fonction $\Phi_{\mathrm{u}} :]0,1]\to\mathbb{R}$ vérifiant la Définition 2.3.1. En appliquant le Théorème 9.3.1 avec la fonction paramétrique $\mu()$ définie pour tout $(h,\mathbb{S})\in\mathbb{H}\times(\mathbb{X}\times\mathbb{Y})^m$ par

$$\mu(h,\mathbb{S}) = -\phi(\mathsf{R}^{\ell}_{\mathcal{D}}(h), \widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h)) - \Phi_{\mathrm{u}}(\tfrac{\delta}{2}) - \ln \pi(h),$$

pour tout $\delta\in]0,1]$, on obtient la borne :

$$\mathop{\mathbb{P}}_{\substack{\mathbb{S}\sim\mathcal{D}^m\\ h'\sim\pi}}\left[\begin{array}{l}\sup_{h\in\mathbb{H}}\ \phi\Big(\mathsf{R}^{\ell}_{\mathcal{D}}(h), \widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h)\Big)\ \le \\ \quad \Phi_{\mathrm{u}}(\frac{\delta}{2}) + \ln\left(\dfrac{16}{\delta^2}\mathop{\mathbb{E}}_{\mathbb{S}'\sim\mathcal{D}^m}\mathop{\mathbb{E}}_{f\sim\pi}\exp\left[\phi\Big(\mathsf{R}^{\ell}_{\mathcal{D}}(f), \widehat{\mathsf{R}}^{\ell}_{\mathbb{S}'}(f)\Big) - \phi\Big(\mathsf{R}^{\ell}_{\mathcal{D}}(h'), \widehat{\mathsf{R}}^{\ell}_{\mathbb{S}}(h')\Big)\right]\right)\end{array}\right] \ge 1-\delta.$$

Ce corollaire souligne que notre cadre est suffisamment général pour pouvoir obtenir une borne basée sur la convergence uniforme. En effet, si nous sommes en mesure de trouver (avec une grande probabilité) une borne supérieure de l'écart en généralisation dans le pire des cas $\sup_h \phi\Big(\mathrm{R}^\ell_{\mathcal{D}}(h), \widehat{\mathrm{R}}^\ell_{\mathbb{S}}(h)\Big)$, notée $\Phi_{\mathtt{u}}(\delta)$, alors notre cadre permet de dériver une borne dépendant de $\Phi_{\mathtt{u}}(\delta)$. Par exemple, considérons la borne $\Phi_{\mathtt{u}}(\delta) = \mathtt{rad}(\mathbb{H}) + \sqrt{\frac{1}{2m}\ln\frac{1}{\delta}}$ qui dépend de la complexité de Rademacher $\mathtt{rad}(\mathbb{H})$. Alors, nous pouvons obtenir le corollaire suivant qui est une borne dépendant de $\Phi_{\mathtt{u}}(\delta)$.

Corollaire 9.5.2 (Borne basée sur la complexité de Rademacher)**.** Soit $\ell : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y})\to[0,1]$ une fonction perte. En appliquant le Théorème 9.3.1 avec la fonction paramétrique $\mu()$ définie pour tout $(h,\mathbb{S}) \in \mathbb{H}\times(\mathbb{X}\times\mathbb{Y})^m$ par

$$\mu(h,\mathbb{S}) = -\sqrt{m}[\mathrm{R}^\ell_{\mathcal{D}}(h) - \widehat{\mathrm{R}}^\ell_{\mathbb{S}}(h)] - \sqrt{m}\left[\mathtt{rad}(\mathbb{H}) + \sqrt{\tfrac{1}{2m}\ln\tfrac{2}{\delta}}\right] - \ln\pi(h),$$

pour tout $\delta\in]0,1]$, on obtient la borne :

$$\mathop{\mathbb{P}}_{\mathbb{S}\sim\mathcal{D}^m}\left[\sup_{h\in\mathbb{H}}\left[\mathrm{R}^\ell_{\mathcal{D}}(h) - \widehat{\mathrm{R}}^\ell_{\mathbb{S}}(h)\right] \;\le\; \mathtt{rad}(\mathbb{H}) + \sqrt{\frac{1}{2m}\ln\frac{4}{\delta}} + \frac{\ln\frac{128}{\delta^3}+2}{\sqrt{m}}\right] \ge 1-\delta.$$

La borne du Corollaire 9.5.2 est supérieure à la borne du Théorème 2.3.3 (Chapitre 2). C'est un comportement attendu puisque nous utilisons la borne dans la fonction $\mu()$. Nous pouvons cependant noter que plus le nombre d'exemples m est grand, plus notre borne sera proche de la borne classique de Mohri et al. (2012). Obtenir de nouvelles bornes basées sur la convergence uniforme (sans s'appuyer sur des bornes déjà connues) en définissant une fonction paramétrique spécifique $\mu()$ est une tâche non triviale, ce qui en fait une piste de recherche passionnante à explorer.

Borne en généralisation dépendante de l'algorithme. De manière similaire, nous pouvons obtenir la borne en généralisation suivante.

Corollaire 9.5.3 (Borne dépendante de l'algorithme)**.** Soit $\ell : \mathbb{H}\times(\mathbb{X}\times\mathbb{Y})\to[0,1]$ une fonction perte et $\phi:[0,1]^2\to\mathbb{R}$ l'écart en généralisation. On suppose qu'il existe une fonction $\Phi_{\mathtt{a}} :]0,1]\to\mathbb{R}$ vérifiant la Définition 2.3.4. En appliquant le Théorème 9.3.1 avec la fonction paramétrique $\mu()$ définie pour tout $(h,\mathbb{S}) \in \mathbb{H}\times(\mathbb{X}\times\mathbb{Y})^m$ par

$$\mu(h,\mathbb{S}) = -\phi(\mathrm{R}^\ell_{\mathcal{D}}(h), \widehat{\mathrm{R}}^\ell_{\mathbb{S}}(h)) - \Phi_{\mathtt{a}}(\tfrac{\delta}{2}) - \ln\pi(h),$$

pour tout $\delta\in]0,1]$, on obtient la borne :

$$\mathop{\mathbb{P}}_{\substack{\mathbb{S}\sim\mathcal{D}^m\\ h'\sim\pi}}\left[\begin{array}{l}\phi\Big(\mathrm{R}^\ell_{\mathcal{D}}(h_{\mathbb{S}}), \widehat{\mathrm{R}}^\ell_{\mathbb{S}}(h_{\mathbb{S}})\Big) \le \\ \quad \Phi_{\mathtt{a}}(\frac{\delta}{2}) + \ln\left(\dfrac{16}{\delta^2}\mathop{\mathbb{E}}_{\mathbb{S}'\sim\mathcal{D}^m}\mathop{\mathbb{E}}_{f\sim\pi}\exp\left[\phi\Big(\mathrm{R}^\ell_{\mathcal{D}}(f), \widehat{\mathrm{R}}^\ell_{\mathbb{S}'}(f)\Big) - \phi\Big(\mathrm{R}^\ell_{\mathcal{D}}(h'), \widehat{\mathrm{R}}^\ell_{\mathbb{S}}(h')\Big)\right]\right)\end{array}\right] \ge 1-\delta.$$

Ainsi, notre cadre est également suffisamment général pour pouvoir retrouver des bornes qui dépendent de l'algorithme d'apprentissage. Plus précisément, l'écart en généralisation $\phi\Big(\mathrm{R}^\ell_{\mathcal{D}}(h_{\mathbb{S}}), \widehat{\mathrm{R}}^\ell_{\mathbb{S}}(h_{\mathbb{S}})\Big)$ associé à l'hypothèse $h_{\mathbb{S}}$ est majoré par une constante. Comme pour les Corollaires 9.5.1 et 9.5.2, l'inconvénient du Corollaire 9.5.3 réside dans le fait qu'il est

nécessaire de s'appuyer sur une borne déjà connue pour obtenir notre résultat. Ainsi, des recherches supplémentaires doivent être menées pour établir de nouvelles bornes entièrement dépendantes de l'algorithme en définissant une fonction paramétrique spécifique $\mu()$.

9.6 Conclusion

Contrairement aux cadres classiques de la théorie statistique de l'apprentissage, où une mesure de complexité est imposée par le cadre lui-même, nous proposons une borne en généralisation générique et novatrice permettant à l'utilisateur de choisir une fonction paramétrique agissant comme mesure de complexité. Cette mesure intègre une fonction qui dépend à la fois des données et du modèle. Un des intérêts est que cette mesure peut être conçue pour favoriser des propriétés souhaitées sur les hypothèses. En particulier, nous avons montré empiriquement que lorsque cette fonction est apprise de manière à représenter l'écart en généralisation, nos bornes sont précises, même sans recours à des *priors* dépendants des données. À notre connaissance, notre cadre est l'un des rares suffisamment généraux pour offrir des garanties théoriques pour des mesures de complexité apprises et pour celles utilisées en pratique (par exemple, basées sur certaines normes de poids).

Enfin, nous pensons que ce travail ouvre de nouvelles perspectives de recherche visant à rapprocher la théorie statistique de l'apprentissage et la pratique. En effet, notre cadre pourrait fournir des informations précieuses sur la généralisation des modèles profonds en intégrant de nouvelles mesures de complexité, telles que *(i)* l'apprentissage d'un modèle interprétable basé sur des caractéristiques comme la configuration d'un réseau de neurones, ou *(ii)* de nouvelles fonctions paramétriques conçues à la main, simples, mais prédictives de la généralisation.

Cinquième partie

Bilan global

Conclusion 10

Les travaux présentés dans ce manuscrit regroupent mes contributions en théorie PAC-Bayésienne, fil rouge de mes recherches depuis plus de 10 ans. Cette histoire a débuté lors de ma deuxième année de thèse avec ma rencontre avec François Laviolette et la publication de mon premier article sur la théorie PAC-Bayésienne : *PAC-Bayesian Generalization Bound on Confusion Matrix for Multi-Class Classification* (Morvant et al., 2012b), en collaboration avec Sokol Koço et Liva Ralaivola. Ce papier, à l'approche purement théorique, propose une borne PAC-Bayésienne sans lien direct avec un algorithme particulier. C'est ce résultat qui m'a conduit à me familiariser avec la théorie PAC-Bayésienne (à l'époque peu explorée en comparaison avec les avancées actuelles), pour ensuite arriver au développement des nouvelles méthodes et algorithmes présentés dans ce manuscrit, avec comme point culminant le développement d'algorithmes auto-certifiés, directement orientés vers l'optimisation des garanties théoriques. J'ai donc choisi de ne pas inclure dans ce manuscrit mes travaux qui ne relèvent pas directement de cette thématique (Bellet et al., 2014 ; Marchand et al., 2014 ; Goyal et al., 2018 ; Gautheron et al., 2019 ; Patracone et al., 2024), ou qui n'ont pas donné lieu à une publication.

Les travaux présentés (ou non) dans ce manuscrit constituent sans aucun doute les bases et les fondements des orientations futures que je poursuivrai. Je considère ce manuscrit comme une photographie à l'instant t de mon parcours scientifique. Il illustre le chemin parcouru au fil des années, tout en reflétant mes réflexions, mes collaborations et mes projets. L'objectif de cette partie n'est donc pas de fournir une conclusion globale au manuscrit, mais plutôt de partager quelques réflexions personnelles sur mon parcours et sur la direction que je prends.

10.1 Sur l'importance du partage des savoirs

Je commence par ce qui est très probablement le plus important pour moi, à savoir que la recherche et l'enseignement suivent un même fil conducteur : le partage des connaissances. Avec le recul, c'est précisément ce qui constitue le moteur de mon avancement. Ce que j'estime être ma véritable réussite aujourd'hui ne réside pas réellement dans les résultats scientifiques présentés dans ce manuscrit, mais plutôt dans le partage de savoirs qui a permis d'atteindre de tels résultats. Sans enseignement à la base, il n'y aurait pas d'avancée scientifique (et réciproquement). Nous nous souvenons tous et toutes d'un ou plusieurs enseignants qui ont marqué notre parcours scolaire, en posant les fondements de ce que nous sommes devenus. C'est pourquoi mon engagement dans l'enseignement en Licence, en particulier avec ma responsabilité en L2, est au cœur de mes activités.

Même s'il est difficile de suivre le devenir des étudiants, j'essaie de leur transmettre, à mon tour, ce qui m'a été transmis, apportant ainsi ma pierre à l'édifice de leur construction personnelle et professionnelle. Il est d'ailleurs extrêmement gratifiant de retrouver en thèse, qu'ils soient encadrés par moi ou non, des étudiants que j'ai eu la chance d'accompagner depuis leur L1 ou leur L2. L'encadrement de stagiaires de master et de doctorants s'inscrit

ainsi naturellement dans la continuité de mon implication en enseignement à la Faculté des Sciences et Techniques de l'UJM, offrant une autre facette du partage des connaissances.

Quel que soit le doctorant, je perçois la thèse comme un échange d'expériences et de compétences, à la fois scientifique et humain, enrichissant pour les deux parties. J'envisage le rôle d'encadrant de thèse comme une relation de Maître à élève, où il s'agit de guider, d'orienter et de mettre le doctorant sur la voie, tout en lui laissant l'espace nécessaire pour se forger sa propre expérience, apprendre de ses erreurs et devenir autonome. Une fois "sur les rails", le doctorant pourra à son tour transmettre ce qu'il a appris.

C'est avec cette vision que j'ai encadré mes trois premiers doctorants et que j'en encadre actuellement deux. C'est grâce au travail de ces doctorants que nous avons pu obtenir les résultats présentés dans ce manuscrit, sans eux et sans toutes les personnes avec qui j'ai pu collaborer, cela n'aurait pas été possible. Leur contribution est fondamentale, et je leur en suis profondément reconnaissante.

10.2 Sur mon parcours scientifique

J'ai effectué ma thèse d'octobre 2010 à septembre 2013 au sein de l'équipe Qarma du Laboratoire d'Informatique Fondamentale de Marseille (aujourd'hui le LIS) sous la direction d'Amaury Habrard et de Stéphane Ayache. Mes travaux portaient sur l'apprentissage automatique, et j'ai eu la chance de pouvoir explorer à la fois des problématiques pratiques, en développant des algorithmes pour la classification de données multimédia (images et vidéos), et des problématiques théoriques, notamment en lien avec la théorie de l'adaptation de domaine et la théorie PAC-Bayésienne. J'ai poursuivi avec un post-doctorat d'octobre 2013 à septembre 2014 au sein de l'équipe de vision par ordinateur et apprentissage automatique dirigée par Christoph Lampert à l'IST Austria. Comme pour ma thèse, ce post-doctorat se situait à la croisée de la théorie et de la pratique. C'est durant cette période que j'ai pris conscience de mon intérêt profond pour l'apport de la théorie dans l'apprentissage automatique, ce qui a défini la ligne directrice de mes travaux depuis.

En 2014, j'ai rejoint l'Université Jean Monnet de Saint-Étienne en tant que Maître de Conférences. Mes recherches ont d'abord porté sur des problématiques assez proches de ma thèse et de mon post-doctorat. J'ai en effet déposé un projet de thèse de doctorat auprès de la région Auvergne-Rhône-Alpes (dispositif ARC6) sur l'apprentissage multivues et la théorie PAC-Bayésienne. Cette demande a conduit au financement de la thèse d'Anil Goyal (2015-2018) que j'ai co-encadré avec Massih Reza Amini (LIG, Grenoble), dont une partie des résultats est présentée dans le Chapitre 3. En parallèle, j'ai étendu mes travaux sur la théorie de l'adaptation de domaine débutés durant ma thèse et présentés dans le Chapitre 4. Ensuite, j'ai été impliquée dans la thèse de Léo Gautheron (2017-2020) en co-encadrement avec Marc Sebban et Amaury Habrard, thèse durant laquelle je suis montée en compétence sur l'apprentissage à partir de données déséquilibrées et sur l'apprentissage de métriques. Une partie des résultats obtenus durant cette thèse est présentée dans le Chapitre 5. C'est durant cette période que mes recherches se sont plus spécifiquement recentrées sur la théorie PAC-Bayésienne et son potentiel pour améliorer les méthodes d'apprentissage de représentation. Cela inclut des approches en apprentissage de métriques et des travaux en apprentissage profond. Ces recherches m'ont conduit à déposer un projet ANR (projet APRIORI) qui a été accepté et que j'ai ainsi piloté de 2019 à 2023. C'est dans ce contexte que j'ai pu co-encadrer la thèse de Paul Viallard (2019-2022, aujourd'hui chercheur à l'Inria Rennes) avec Amaury Habrard et Pascal Germain (GRAAL, Université Laval, Québec). Malgré les contraintes imposées par la période du COVID, ce projet a été une réussite, posant les

bases pour la dérivation d'algorithmes auto-certifiés. Une partie des résultats obtenus sont présentés dans les Chapitres 6, 7, 8 et 9.

10.3 Sur la suite

Depuis l'obtention des derniers résultats présentés dans ce manuscrit, je m'intéresse à des problématiques cruciales en apprentissage automatique aujourd'hui. En effet, l'analyse théorique des capacités en généralisation des méthodes d'apprentissage sur de nouvelles données devient de plus en plus essentielle pour garantir la robustesse des décisions face aux attaques extérieures, ainsi que l'équité des modèles face aux biais intrinsèques des données ou de la structure des algorithmes. Actuellement, je suis impliquée en tant que coordinatrice locale dans un projet ANR consacré à l'apprentissage équitable à partir de données multi-vues (faisant une sorte de lien entre mes travaux post thèse et mes intérêts actuels). Ce projet finance la thèse de Julien Bastian, débutée en octobre 2024, que je co-encadre avec Christine Largeron et Guillaume Metzler (ERIC, Lyon 2). Par ailleurs, la thèse de Hind Atbir, co-encadrée avec Rémi Eyraud, Farah Cherfaoui et Paul Viallard, a également débuté en 2024 (financée par l'École Doctorale SIS). Ses travaux se concentrent actuellement sur la dérivation de bornes PAC-Bayésiennes pour des fonctions pertes pouvant être utilisées non seulement dans le cadre de l'apprentissage équitable, mais également pour d'autres problématiques telle que l'apprentissage à partir de données déséquilibrées. Nous avons d'ailleurs déjà obtenu des premiers résultats préliminaires prometteurs présentés lors de la conférence en apprentissage automatique francophone (Atbir et al., 2024).

En parallèle, motivée par l'objectif d'améliorer la confiance, la fiabilité et la sécurité des modèles d'apprentissage automatique, j'ai déposé un nouveau projet ANR, en collaboration notamment avec Paul Viallard. Ce projet, nommé APosTeriori (en référence à la théorie PAC-Bayésienne), s'inscrit dans la continuité des avancées obtenues avec le projet APRIORI. Il vise à combler le fossé entre les avancées théoriques et les applications pratiques dans deux cadres spécifiques de l'apprentissage automatique : les bandits manchots et la prédiction conforme. Contrairement aux approches classiques dans ces cadres, où les algorithmes sont conçus *a priori* sans nécessairement tenir compte des garanties théoriques, ce projet a pour objectif de dériver de nouveaux algorithmes directement à partir de ces garanties (*a posteriori*). L'objectif principal est d'établir des bornes théoriques exploitables pour le regret (dans le cadre des bandits manchots) et pour le *coverage* (dans le cadre de la prédiction conforme). Ces bornes serviront de base pour développer des algorithmes auto-certifiés, capables d'apprendre des modèles avec des garanties théoriques précises et adaptées à chaque cas spécifique.

≪ La route ? Là où on va, on n'a pas besoin de route ! ≫

De Robert Zemeckis / Retour vers le futur, Dr. Emmett Brown

Bibliographie

R. Agrawal, T. Campbell, J. Huggins et T. Broderick. Data-dependent compression of random features for large-scale kernel approximation. *International Conference on Artificial Intelligence and Statistics*. (2019) — **cité page 81**.

P. Alquier et B. Guedj. Simpler PAC-Bayesian bounds for hostile data. *Machine Learning*. (2018) — **cité pages 86, 87**.

P. Alquier, J. Ridgway et N. Chopin. On the properties of variational approximations of Gibbs posteriors. *Journal of Machine Learning Research*. (2016) — **cité pages 76, 77, 138**.

A. Ambroladze, E. Parrado-Hernández et J. Shawe-Taylor. Tighter PAC-Bayes Bounds. *Advances in Neural Information Systems*. (2006) — **cité pages 64, 65, 129**.

M. R. Amini, N. Usunier et C. Goutte. Learning from multiple partially observed views-an application to multilingual text categorization. *Advances in Neural Information Processing Systems*. (2009) — **cité pages 46, 52**.

H. Atbir, F. Cherfaoui, G. Metzler, E. Morvant et P. Viallard. Une borne PAC-Bayésienne sur une mesure de risque pour l'apprentissage équitable. *French Conference on Machine Learning (CAp)*. (2024) — **cité page 150**.

P. K. Atrey, M. A. Hossain, A. El Saddik et M. S. Kankanhalli. Multimodal fusion for multimedia analysis: a survey. *Multimedia systems*. (2010) — **cité page 45**.

F. R. Bach. On the Equivalence between Kernel Quadrature Rules and Random Feature Expansions. *Journal of Machine Learning Research*. (2017) — **cité pages 74, 75**.

M. Balcan, A. Blum et N. Srebro. A theory of learning with similarity functions. *Machine Learning*. (2008) — **cité page 78**.

M. Balcan, A. Blum et N. Srebro. Improved Guarantees for Learning via Similarity Functions. *Annual Conference on Learning Theory*. (2008b) — **cité page 78**.

P. Bartlett et S. Mendelson. Rademacher and Gaussian Complexities: Risk Bounds and Structural Results. *Journal of Machine Learning Research*. (2002) — **cité pages 13, 25**.

B. Bauvin, C. Capponi, J. Roy et F. Laviolette. Fast greedy C-bound minimization with guarantees. *Machine Learning*. (2020) — **cité pages 91, 94, 108**.

L. Bégin, P. Germain, F. Laviolette et J. Roy. PAC-Bayesian Bounds based on the Rényi Divergence. *International Conference on Artificial Intelligence and Statistics*. (2016) — **cité pages 40, 87, 124, 126**.

A. Bellet, A. Habrard, E. Morvant et M. Sebban. Learning A Priori Constrained Weighted Majority Votes. *Machine Learning*. (2014) — **cité pages 91, 148**.

S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira et J. Vaughan. A theory of learning from different domains. *Machine Learning*. (2010) — **cité pages 54, 55, 57, 70**.

S. Ben-David, J. Blitzer, K. Crammer et F. Pereira. Analysis of Representations for Domain Adaptation. *Advances in Neural Information Processing Systems*. (2006) — **cité pages 54, 55, 57, 70**.

S. Ben-David, T. Lu, T. Luu et D. Pal. Impossibility Theorems for Domain Adaptation. *International Conference on Artificial Intelligence and Statistics*. (2010b) — **cité pages 54, 56**.

S. Ben-David et R. Urner. On the Hardness of Domain Adaptation and the Utility of Unlabeled Target Samples. *Algorithmic Learning Theory*. (2012) — **cité pages 54, 56, 61, 62**.

S. Ben-David et R. Urner. Domain adaptation—can quantity compensate for quality? *Annals of Mathematics and Artificial Intelligence*. (2014) — **cité pages 54, 56**.

J. Besag. Comments on "Representations of knowledge in complex systems" by U. Grenander and MI Miller. *Journal of the Royal Statistical Society, Series B*. (1994) — **cité page 141**.

B. Biggio, I. Corona, D. Maiorca, B. Nelson, N. Srndic, P. Laskov, G. Giacinto et F. Roli. Evasion Attacks against Machine Learning at Test Time. *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*. (2013) — **cité page 110**.

F. Biggs et B. Guedj. Differentiable PAC-Bayes Objectives with Partially Aggregated Neural Networks. *Entropy*. (2021) — **cité page 123**.

F. Biggs et B. Guedj. On Margins and Derandomisation in PAC-Bayes. *International Conference on Artificial Intelligence and Statistics*. (2022) — **cité pages 123, 125**.

C. Bishop. Pattern recognition and machine learning (5th Edition). *Information science and statistics. Springer*. (2007) — **cité page 34**.

G. Blanchard et F. Fleuret. Occam's Hammer. *Annual Conference on Learning Theory*. (2007) — **cité pages 41, 42, 124, 127, 128, 130, 133, 136**.

J. Blitzer, R. McDonald et F. Pereira. Domain Adaptation with Structural Correspondence Learning. *Conference on Empirical Methods in Natural Language Processing*. (2006) — **cité page 69**.

R. Boik et J. Robinson-Cox. Derivatives of the incomplete beta function. *Journal of Statistical Software*. (1999) — **cité page 108**.

B. Boser, I. Guyon et V. Vapnik. A training algorithm for optimal margin classifiers. *Annual Conference on Learning Theory*. (1992) — **cité page 72**.

O. Bousquet et A. Elisseeff. Stability and Generalization. *Journal of Machine Learning Research*. (2002) — **cité pages 26, 27**.

S. Boyd et L. Vandenberghe. Convex Optimization. *Cambridge University Press*. (2004) — **cité pages 95, 99**.

L. Breiman. Bagging Predictors. *Machine Learning*. (1996) — **cité page 30**.

L. Breiman. Random Forests. *Machine Learning*. (2001) — **cité pages 15, 30, 31, 34**.

L. Bruzzone et M. Marconcini. Domain Adaptation Problems: A DASVM Classification Technique and a Circular Validation Strategy. *IEEE Transaction Pattern Analysis and Machine Intelligence*. (2010) — **cité pages 54, 70**.

N. Carlini et D. Wagner. Towards Evaluating the Robustness of Neural Networks. *IEEE Symposium on Security and Privacy*. (2017) — **cité pages 110, 113**.

O. Catoni. Statistical learning theory and stochastic optimization: Ecole d'Eté de Probabilités de Saint-Flour, XXXI-2001. *Springer Science & Business Media*. (2004) — **cité page 138**.

O. Catoni. PAC-Bayesian Supervised Classification: The Thermodynamics of Statistical Learning. *Inst. of Mathematical Statistic*. (2007) — **cité pages 35, 37, 40-43, 49, 63, 77, 105, 124, 127, 128, 130, 133, 136, 138**.

M. Chen, K. Q. Weinberger et J. Blitzer. Co-Training for Domain Adaptation. *Advances in Neural Information Processing Systems*. (2011) — **cité pages 69, 70**.

M. Chen, Z. E. Xu, K. Q. Weinberger et F. Sha. Marginalized Denoising Autoencoders for Domain Adaptation. *International Conference on Machine Learning*. (2012) — **cité pages 54, 62**.

T.-S. Chiang, C.-R. Hwang et S. J. Sheu. Diffusion for global optimization in $\mathbb{R}^n$. *SIAM Journal on Control and Optimization*. (1987) — **cité page 139**.

S. Chib et E. Greenberg. Understanding the Metropolis-Hastings Algorithm. *The american statistician*. (1995) — **cité page 141**.

K. Choromanski, M. Rowland, T. Sarlós, V. Sindhwani, R. E. Turner et A. Weller. The Geometry of Random Features. *International Conference on Artificial Intelligence and Statistics*. (2018) — **cité page 75**.

J. Cohen, E. Rosenfeld et Z. Kolter. Certified Adversarial Robustness via Randomized Smoothing. *International Conference on Machine Learning*. (2019) — **cité page 113**.

C. Cortes, Y. Mansour et M. Mohri. Learning Bounds for Importance Weighting. *Advances in Neural Information Processing Systems*. (2010) — **cité page 54**.

C. Cortes et M. Mohri. Domain adaptation and sample bias correction theory and algorithm for regression. *Theoretical Computer Science*. (2014) — **cité page 54**.

C. Cortes, M. Mohri et A. M. Medina. Adaptation Algorithm and Theory Based on Generalized Discrepancy. *ACM International Conference on Knowledge Discovery and Data Mining*. (2015) — **cité page 54**.

C. Cortes et V. Vapnik. Support-vector networks. *Machine Learning*. (1995) — **cité pages 30, 72**.

N. Courty, R. Flamary, D. Tuia et A. Rakotomamonjy. Optimal transport for Domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. (2016) — **cité pages 54, 55**.

N. Courty, R. Flamary, D. Tuia et A. Rakotomamonjy. Optimal Transport for Domain Adaptation. *IEEE Transaction Pattern Analysis and Machine Intelligence*. (2017) — **cité pages 54, 55**.

A. S. Dalalyan et A. B. Tsybakov. Sparse regression learning by aggregation and Langevin Monte-Carlo. *Journal of Computer and System Sciences*. (2012) — **cité page 77**.

S. Diamond et S. Boyd. CVXPY: A Python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*. (2016) — **cité page 98**.

T. Dietterich. Ensemble methods in machine learning. *International workshop on multiple classifier systems*. (2000) — **cité page 32**.

Z. Ding et Y. Fu. Deep Domain Generalization With Structured Low-Rank Constraint. *IEEE Transactions on Image Processing*. (2018) — **cité page 54**.

L. Dinh, R. Pascanu, S. Bengio et Y. Bengio. Sharp Minima Can Generalize For Deep Nets. *International Conference on Machine Learning*. (2017) — **cité page 144**.

M. Donsker et S. Varadhan. Asymptotic evaluation of certain Markov process expectations for large time - III. *Communications on pure and applied Mathematics*. (1976) — **cité page 39**.

P. Drineas et M. W. Mahoney. On the Nyström method for approximating a Gram matrix for improved kernel-based learning. *Journal of Machine Learning Research*. (2005) — **cité page 73**.

G. K. Dziugaite, A. Drouin, B. Neal, N. Rajkumar, E. Caballero, L. Wang, I. Mitliagkas et D. M. Roy. In search of robust measures of generalization. *Advances in Neural Information Systems*. (2020) — **cité pages 135, 144**.

G. K. Dziugaite, K. Hsu, W. Gharbieh, G. Arpino et D. Roy. On the role of data in PAC-Bayes. *International Conference on Artificial Intelligence and Statistics*. (2021) — **cité pages 35, 106, 118, 140, 144**.

G. K. Dziugaite et D. Roy. Computing Nonvacuous Generalization Bounds for Deep (Stochastic) Neural Networks with Many More Parameters than Training Data. *Conference on Uncertainty in Artificial Intelligence*. (2017) — **cité pages 129, 132, 144**.

G. K. Dziugaite et D. Roy. Data-dependent PAC-Bayes priors via differential privacy. *Advances in Neural Information Systems*. (2018) — **cité pages 118, 131, 132, 142**.

T. van Erven et P. Harremoës. Rényi Divergence and Kullback-Leibler Divergence. *IEEE Transactions on Information Theory*. (2014) — **cité pages 40, 60, 128**.

F. Farnia, J. Zhang et D. Tse. Generalizable Adversarial Training via Spectral Normalization. *International Conference on Learning Representations*. (2019) — **cité page 113**.

M. Figurnov, S. Mohamed et A. Mnih. Implicit Reparameterization Gradients. *Advances in Neural Information Systems*. (2018) — **cité page 108**.

Y. Freund. Boosting a weak learning algorithm by majority. *Information and computation*. 2. (1995) — **cité page 46**.

Y. Freund. Self Bounding Learning Algorithms. *Annual Conference on Learning Theory*. (1998) — **cité pages 14, 92, 109, 111, 118, 126**.

Y. Freund et R. Schapire. Experiments with a New Boosting Algorithm. *International Conference on Machine Learning*. (1996) — **cité page 30**.

Y. Freund et R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*. 1. (1997) — **cité pages 46, 50**.

J. Friedman. Greedy function approximation: a gradient boosting machine. *Annals of Statistics*. (2001) — **cité pages 73, 78, 79**.

Y. Ganin, E. Ustinova, A. H, P. Germain, H. Larochelle, F. Laviolette, M. Marchand et V. Lempitsky. Domain-Adversarial Training of Neural Networks. *Journal of Machine Learning Research*. (2016) — **cité pages 54, 62**.

L. Gautheron, P. Germain, A. Habrard, G. Metzler, E. Morvant, M. Sebban et V. Zantedeschi. Landmark-based Ensemble Learning with Random Fourier Features and Gradient Boosting. *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*. (2020) — **cité pages 72, 83, 84**.

L. Gautheron, A. Habrard, E. Morvant et M. Sebban. Metric learning from imbalanced data. *IEEE International Conference on Tools with Artificial Intelligence*. (2019) — **cité page 148**.

P. Germain, A. Habrard, F. Laviolette et E. Morvant. A PAC-Bayesian Approach for Domain Adaptation with Specialization to Linear Classifiers. *International Conference on Machine Learning*. (2013) — **cité pages 53, 55, 58, 64**.

P. Germain, A. Habrard, F. Laviolette et E. Morvant. A New PAC-Bayesian Perspective on Domain Adaptation. *International Conference on Machine Learning*. (2016a) — **cité pages 53, 55, 58**.

P. Germain, A. Lacasse, F. Laviolette et M. Marchand. PAC-Bayesian Learning of Linear Classifiers. *International Conference on Machine Learning*. (2009) — **cité pages 35, 36, 38, 40-42, 64, 65, 77, 125, 129, 132**.

P. Germain, F. Bach, A. Lacoste et S. Lacoste-Julien. PAC-Bayesian Theory Meets Bayesian Inference. *Advances in Neural Information Systems*. (2016b) — **cité page 77**.

P. Germain, A. Habrard, F. Laviolette et E. Morvant. PAC-Bayes and domain adaptation. *Neurocomputing*. (2020) — **cité pages 53, 58, 63, 64, 67, 70, 71**.

P. Germain, A. Lacasse, F. Laviolette, M. Marchand et J. Roy. Risk Bounds for the Majority Vote: From a PAC-Bayesian Analysis to a Learning Algorithm. *Journal of Machine Learning Research*. (2015) — **cité pages 31, 34, 46, 94, 98**.

X. Glorot, A. Bordes et Y. Bengio. Domain adaptation for large-scale sentiment classification: A deep learning approach. *International Conference on Machine Learning*. (2011) — **cité page 54**.

I. Goodfellow, Y. Bengio et A. Courville. Deep Learning. *Adaptive computation and machine learning. MIT Press*. (2016) — **cité pages 73, 131**.

I. Goodfellow, J. Shlens et C. Szegedy. Explaining and Harnessing Adversarial Examples. *International Conference on Learning Representations*. (2015) — **cité pages 110, 112**.

A. Goyal, E. Morvant et M.-R. Amini. Multiview learning of weighted majority vote by bregman divergence minimization. *Symposium on Intelligent Data Analysis*. (2018) — **cité page 148**.

A. Goyal, E. Morvant, P. Germain et M.-R. Amini. PAC-Bayesian Analysis for a two-step Hierarchical Multiview Learning Approach. *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*. (2017) — **cité page 45**.

A. Goyal, E. Morvant, P. Germain et M.-R. Amini. Multiview Boosting by Controlling the Diversity and the Accuracy of View-specific Voters. *Neurocomputing*. (2019) — **cité pages 45, 51**.

T. Graepel, R. Herbrich et J. Shawe-Taylor. PAC-Bayesian Compression Bounds on the Prediction Error of Learning Algorithms for Classification. *Machine Learning*. (2005) — **cité page 30**.

M. Grant, S. Boyd et Y. Ye. Disciplined Convex Programming. *Global optimization*. (2006) — **cité page 98**.

A. Habrard, J.-P. Peyrache et M. Sebban. Iterative Self-labeling Domain Adaptation for Linear Structured Image Classification. *International Journal on Artificial Intelligence Tools*. (2013) — **cité page 54**.

D. Hendrycks et T. Dietterich. Benchmarking Neural Network Robustness to Common Corruptions and Perturbations. *International Conference on Learning Representations*. (2019) — **cité pages 113, 114**.

J. Honorio et T. S. Jaakkola. Tight Bounds for the Expected Risk of Linear Classifiers and PAC-Bayes Finite-Sample Guarantees. *International Conference on Artificial Intelligence and Statistics*. (2014) — **cité page 87**.

J. Huang, A. Smola, A. Gretton, K. Borgwardt et B. Schölkopf. Correcting Sample Selection Bias by Unlabeled Data. *Advances in Neural Information Processing Systems*. (2006) — **cité page 54**.

M. Jankowiak et F. Obermeyer. Pathwise Derivatives Beyond the Reparameterization Trick. *International Conference on Machine Learning*. (2018) — **cité page 108**.

Y. Jiang, B. Neyshabur, H. Mobahi, D. Krishnan et S. Bengio. Fantastic Generalization Measures and Where to Find Them. *International Conference on Learning Representations*. (2019) — **cité pages 135, 143, 144**.

Y. Jiang et al. Methods and Analysis of The First Competition in Predicting Generalization of Deep Learning. *NeurIPS 2020 Competition and Demonstration Track*. (2021) — **cité page 135**.

K. Kawaguchi, L. P. Kaelbling et Y. Bengio. Generalization in Deep Learning. *arXiv preprint arXiv:1710.05468*. (2017) — **cité page 30**.

G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye et T.-Y. Liu. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in Neural Information Processing Systems*. (2017) — **cité page 84**.

H. Kervadec, J. Dolz, J. Yuan, C. Desrosiers, E. Granger et I. B. Ayed. Constrained deep networks: Lagrangian optimization via log-barrier extensions. *IEEE European Signal Processing Conference*. (2022) — **cité page 95**.

J. Khim et P. Loh. Adversarial Risk Bounds for Binary Classification via Function Transformation. *CoRR*. (2018) — **cité page 113**.

D. Kingma et J. Ba. Adam: A Method for Stochastic Optimization. *International Conference on Learning Representations*. (2015) — **cité page 96**.

S. Koço et C. Capponi. A boosting approach to multiview classification with cooperation. *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*. (2011) — **cité page 52**.

A. Krizhevsky. Learning Multiple Layers of Features from Tiny Images. Mém. de mast. University of Toronto, (2009) — **cité page 132**.

L. Kuncheva. Combining pattern classifiers: methods and algorithms. *John Wiley & Sons*. (2014) — **cité pages 32, 46, 50**.

A. Kurakin, I. Goodfellow et S. Bengio. Adversarial Machine Learning at Scale. *International Conference on Learning Representations*. (2017) — **cité pages 110, 112**.

A. Lacasse. Bornes PAC-Bayes et algorithmes d'apprentissage. Thèse de doct. Université Laval, Québec, (2010) — **cité pages 38, 101**.

A. Lacasse, F. Laviolette, M. Marchand, P. Germain et N. Usunier. PAC-Bayes Bounds for the Risk of the Majority Vote and the Variance of the Gibbs Classifier. *Advances in Neural Information Systems*. (2006) — **cité pages 31-34, 92, 94, 95, 97, 108**.

A. Lacasse, F. Laviolette, M. Marchand et F. Turgeon-Boutin. Learning with Randomized Majority Votes. *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*. (2010) — **cité pages 100, 109**.

J. Langford. Tutorial on Practical Prediction Theory for Classification. *Journal of Machine Learning Research*. (2005) — **cité pages 41, 65**.

J. Langford et J. Shawe-Taylor. PAC-Bayes & Margins. *Advances in Neural Information Processing Systems*. (2002) — **cité pages 31, 33, 41, 64, 123**.

F. Laviolette, E. Morvant, L. Ralaivola et J. Roy. Risk upper bounds for general ensemble methods with an application to multiclass classification. *Neurocomputing*. (2017) — **cité pages 31, 34, 91, 93, 101, 109**.

Y. LeCun, C. Cortes et C. Burges. THE MNIST DATASET of handwritten digits. (1998). URL : `http://yann.lecun.com/exdb/mnist/` — **cité pages 132, 142**.

Y. Lee, J. Lee, S. J. Hwang, E. Yang et S. Choi. Neural Complexity Measures. *Advances in Neural Information Processing Systems*. (2020) — **cité pages 134, 139, 143**.

G. Letarte, P. Germain, B. Guedj et F. Laviolette. Dichotomize and Generalize: PAC-Bayesian Binary Activated Deep Neural Networks. *Advances in Neural Information Processing Systems*. (2019a) — **cité pages 123, 129**.

G. Letarte, E. Morvant et P. Germain. Pseudo-Bayesian Learning with Kernel Fourier Transform as Prior. *International Conference on Artificial Intelligence and Statistics*. (2019b) — **cité pages 72, 84**.

G. Lever, F. Laviolette et J. Shawe-Taylor. Tighter PAC-Bayes bounds through distribution-dependent priors. *Theoretical Computer Science*. (2013) — **cité pages 76, 86, 89, 118, 142**.

J. Li, K. Lu, Z. Huang, L. Zhu et H. T. Shen. Heterogeneous Domain Adaptation Through Progressive Alignment. *IEEE Transactions on Neural Netw. Learning Syst.* (2019) — **cité page 54**.

J. Li, K. Lu, Z. Huang, L. Zhu et H. T. Shen. Transfer Independently Together: A Generalized Framework for Domain Adaptation. *IEEE Transactions on Cybernetics*. (2019) — **cité page 54**.

X. Li et J. Bilmes. A Bayesian divergence prior for classifier adaptation. *International Conference on Artificial Intelligence and Statistics*. (2007) — **cité page 54**.

S. S. Lorenzen, C. Igel et Y. Seldin. On PAC-Bayesian bounds for random forests. *Machine Learning*. (2019) — **cité page 92**.

A. Madry, A. Makelov, L. Schmidt, D. Tsipras et A. Vladu. Towards Deep Learning Models Resistant to Adversarial Attacks. *International Conference on Learning Representations*. (2018) — **cité pages 110, 113**.

Y. Mansour, M. Mohri et A. Rostamizadeh. Domain Adaptation: Learning Bounds and Algorithms. *Annual Conference on Learning Theory*. (2009a) — **cité pages 54, 55, 57, 58, 70**.

Y. Mansour, M. Mohri et A. Rostamizadeh. Multiple Source Adaptation and the Rényi Divergence. *Conference on Uncertainty in Artificial Intelligence*. (2009b) — **cité pages 54, 55**.

M. Marchand, H. Su, E. Morvant, J. Rousu et J. Shawe-Taylor. Multilabel Structured Output Learning with Random Spanning Trees of Max-Margin Markov Networks. *Advances in Neural Information Processing Systems*. (2014) — **cité page 148**.

A. Masegosa, S. S. Lorenzen, C. Igel et Y. Seldin. Second Order PAC-Bayesian Bounds for the Weighted Majority Vote. *Advances in Neural Information Processing Systems*. (2020) — **cité pages 31, 33, 34, 92, 108, 118**.

A. Maurer. A Note on the PAC Bayesian Theorem. *arXiv preprint cs/0411099*. (2004) — **cité pages 29, 38**.

D. A. McAllester et J. Keshet. Generalization Bounds and Consistency for Latent Structural Probit and Ramp Loss. *Advances in Neural Information Processing System*. (2011) — **cité pages 64, 65**.

D. McAllester. Some PAC-Bayesian Theorems. *Annual Conference on Learning Theory*. (1998) — **cité pages 105, 117**.

D. McAllester. Some PAC-Bayesian Theorems. *Machine Learning*. (1999) — **cité pages 28, 34**.

D. McAllester. PAC-Bayesian Stochastic Model Selection. *Machine Learning*. (2003) — **cité pages 31, 35, 36, 40, 49, 92-94, 96**.

C. McDiarmid. On the method of bounded differences. *Surveys in Combinatorics*. (1989) — **cité pages 25, 27**.

Z. Mhammedi, P. Grünwald et B. Guedj. PAC-Bayes Un-Expected Bernstein Inequality. *Advances in Neural Information Processing Systems*. (2019) — **cité page 106**.

M. Mohri, A. Rostamizadeh et A. Talwalkar. Foundations of Machine Learning. *Adaptive computation and machine learning*. *MIT Press*. (2012) — **cité pages 23-25, 145**.

O. Montasser, S. Hanneke et N. Srebro. VC Classes are Adversarially Robustly Learnable, but Only Improperly. *Annual Conference on Learning Theory*. (2019) — **cité page 113**.

O. Montasser, S. Hanneke et N. Srebro. Reducing Adversarially Robust Learning to Non-Robust PAC Learning. *Advances in Neural Information Processing Systems*. (2020) — **cité page 113**.

E. Morvant. Domain Adaptation of Weighted Majority Votes via Perturbed Variation-Based Self-Labeling. *Pattern Recognition Letters*. (2015) — **cité pages 54, 91**.

E. Morvant, A. Habrard et S. Ayache. Parsimonious Unsupervised and Semi-Supervised Domain Adaptation with Good Similarity Functions. *Knowledge and Information Systems*. (2012) — **cité page 54**.

E. Morvant, A. Habrard et S. Ayache. Majority Vote of Diverse Classifiers for Late Fusion. *Statistical Techniques in Pattern Recognition and Structural and Syntactic Pattern RecognitionR*. (2014) — **cité pages 34, 45, 46, 50, 52, 91, 94**.

E. Morvant, S. Koço et L. Ralaivola. PAC-Bayesian Generalization Bound on Confusion Matrix for Multi-Class Classification. *International Conference on Machine Learning*. (2012b) — **cité page 148**.

V. Nagarajan et Z. Kolter. Deterministic PAC-Bayesian generalization bounds for deep networks via generalizing noise-resilience. *International Conference on Learning Representations*. (2019a) — **cité page 125**.

V. Nagarajan et Z. Kolter. Uniform convergence may be unable to explain generalization in deep learning. *Advances in Neural Information Processing Systems*. (2019b) — **cité pages 113, 123**.

Y. Nesterov. Smooth minimization of non-smooth functions. *Mathematical Programming*. (2005) — **cité page 103**.

B. Neyshabur, S. Bhojanapalli et N. Srebro. A PAC-Bayesian Approach to Spectrally-Normalized Margin Bounds for Neural Networks. *International Conference on Learning Representations*. (2018) — **cité pages 123, 125**.

B. Neyshabur, R. Salakhutdinov et N. Srebro. Path-SGD: Path-Normalized Optimization in Deep Neural Networks. *Advances in Neural Information Processing Systems*. (2015) — **cité page 144**.

D. Oglic et T. Gärtner. Greedy feature construction. *Advances in Neural Information Processing Systems*. (2016) — **cité page 84**.

J. B. Oliva, A. Dubey, A. G. Wilson, B. Póczos, J. Schneider et E. P. Xing. Bayesian nonparametric kernel-learning. *International Conference on Artificial Intelligence and Statistics*. (2016) — **cité pages 75, 86**.

F. Orabona et T. Tommasi. Training Deep Networks without Learning Rates Through Coin Betting. *Advances in Neural Information Processing Systems*. (2017) — **cité page 96**.

S. Pan et Q. Yang. A Survey on Transfer Learning. *IEEE Transactions on Knowledge and Data Engineering*. (2010) — **cité page 54**.

N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik et A. Swami. The Limitations of Deep Learning in Adversarial Settings. *IEEE EuroS&P*. (2016) — **cité page 110**.

E. Parrado-Hernández, A. Ambroladze, J. Shawe-Taylor et S. Sun. PAC-Bayes bounds with data dependent priors. *Journal of Machine Learning Research*. (2012) — **cité page 64**.

E. Parrado-Hernández, A. Ambroladze, J. Shawe-Taylor et S. Sun. PAC-Bayes Bounds with Data Dependent Priors. *Journal of Machine Learning Research*. (2012) — **cité pages 35, 106, 118, 140**.

A. Paszke et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. *Advances in Neural Information Processing Systems*. (2019) — **cité pages 97, 108**.

J. Patracone, P. Viallard, E. Morvant, G. Gasso, A. Habrard et S. Canu. A Theoretically Grounded Extension of Universal Attacks from the Attacker's Viewpoint. *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*. (2024) — **cité pages 110, 148**.

F. Pedregosa et al. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*. (2011) — **cité page 69**.

J. Peng, A. J. Aved, G. Seetharaman et K. Palaniappan. Multiview boosting with information propagation for classification. *IEEE transactions on neural networks and learning systems*. (2017) — **cité page 52**.

A. Pentina et C. H. Lampert. A PAC-Bayesian bound for Lifelong Learning. *International Conference on Machine Learning*. (2014) — **cité page 101**.

M. Pérez-Ortiz, O. Rivasplata, J. Shawe-Taylor et C. Szepesvári. Tighter Risk Certificates for Neural Networks. *Journal of Machine Learning Research*. (2021) — **cité pages 131, 132, 140, 144**.

R. Pinot, L. Meunier, A. Araujo, H. Kashima, F. Yger, C. Gouy-Pailler et J. Atif. Theoretical evidence for adversarial robustness through randomization. *Advances in Neural Information Processing Systems*. (2019) — **cité page 113**.

R. Pinot, L. Meunier, F. Yger, C. Gouy-Pailler, Y. Chevaleyre et J. Atif. On the robustness of randomized classifiers to adversarial examples. *Machine Learning*. (2022) — **cité page 113**.

M. Raginsky, A. Rakhlin et M. Telgarsky. Non-convex learning via Stochastic Gradient Langevin Dynamics: a nonasymptotic analysis. *Annual Conference on Learning Theory*. (2017) — **cité page 139**.

A. Rahimi et B. Recht. Random Features for Large-Scale Kernel Machines. *Advances in Neural Information Processing Systems*. (2007) — **cité pages 73, 74, 81, 88**.

L. Ralaivola, M. Szafranski et G. Stempfel. Chromatic PAC-Bayes Bounds for Non-IID Data: Applications to Ranking and Stationary β-Mixing Processes. *Journal of Machine Learning Research*. (2010) — **cité page 116**.

I. Redko, A. Habrard et M. Sebban. Theoretical analysis of domain adaptation with optimal transport. *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*. (2017) — **cité pages 54, 55**.

I. Redko, E. Morvant, A. Habrard, M. Sebban et Y. Bennani. Domain Adaptation Theory: Available Theoretical Results. *ISTE Press - Elsevier*. (2019) — **cité page 53**.

I. Redko, E. Morvant, A. Habrard, M. Sebban et Y. Bennani. A survey on domain adaptation theory. *arXiv preprint arXiv:2004.11829*. (2020) — **cité page 53**.

D. Reeb, A. Doerr, S. Gerwinn et B. Rakitsch. Learning Gaussian Processes by Minimizing PAC-Bayesian Generalization Bounds. *Advances in Neural Information Processing Systems*. (2018) — **cité pages 39, 97**.

K. Ren, T. Zheng et X. Liu. Adversarial Attacks and Defenses in Deep Learning. *Engineering*. (2020) — **cité page 112**.

O. Rivasplata, I. Kuzborskij, C. Szepesvári et J. Shawe-Taylor. PAC-Bayes Analysis Beyond the Usual Bounds. *Advances in Neural Information Processing System*. (2020) — **cité pages 41, 124, 127, 130, 133, 135, 136**.

J. Roy, F. Laviolette et M. Marchand. From PAC-Bayes Bounds to Quadratic Programs for Majority Votes. *International Conference on Machine Learning*. (2011) — **cité pages 15, 31, 91, 94, 108**.

J. Roy, M. Marchand et F. Laviolette. A Column Generation Bound Minimization Approach with PAC-Bayesian Generalization Guarantees. *International Conference on Artificial Intelligence and Statistics*. (2016) — **cité pages 46, 91, 93**.

A. Rudi et L. Rosasco. Generalization Properties of Learning with Random Features. *Advances in Neural Information Systems*. (2017) — **cité page 75**.

H. Salman, J. Li, I. Razenshteyn, P. Zhang, H. Zhang, S. Bubeck et G. Yang. Provably Robust Deep Learning via Adversarially Trained Smoothed Classifiers. *Advances in Neural Information Processing Systems*. (2019) — **cité page 113**.

A. S. Sebag, L. Heinrich, M. Schoenauer, M. Sebag, L. F. Wu et S. J. Altschuler. Multi-Domain Adversarial Learning. *International Conference on Learning Representations*. (2019) — **cité page 54**.

M. Sebban, H.-M. Suchier et al. Boosting classifiers built from different subsets of features. *Fundamenta Informaticae*. (2009) — **cité page 51**.

M. Seeger. PAC-Bayesian Generalisation Error Bounds for Gaussian Process Classification. *Journal of Machine Learning Research*. (2002) — **cité pages 35, 37, 38, 40, 42, 49, 92, 94, 96, 104, 105, 117**.

J. Shawe-Taylor et N. Cristianini. Kernel Methods for Pattern Analysis. *Cambridge University Press*. (2004) — **cité page 72**.

J. Shawe-Taylor et R. Williamson. A PAC Analysis of a Bayesian Estimator. *Annual Conference on Learning Theory*. (1997) — **cité pages 13, 28, 34**.

H. Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*. (2000) — **cité page 61**.

R. Shu, H. H. Bui, H. Narui et S. Ermon. A DIRT-T Approach to Unsupervised Domain Adaptation. *International Conference on Learning Representations*. (2018) — **cité page 54**.

A. Sinha et J. C. Duchi. Learning Kernels with Random Features. *Advances in Neural Information Systems*. (2016) — **cité pages 75, 81, 86-89**.

A. Smith et G. Roberts. Bayesian computation via the Gibbs sampler and related Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society, Series B (Methodological)*. (1993) — **cité page 141**.

C. G. Snoek, M. Worring et A. W. Smeulders. Early versus late fusion in semantic video analysis. *ACM Multimedia*. (2005) — **cité page 45**.

M. Sugiyama, S. Nakajima, H. Kashima, P. V. Buenau et M. Kawanabe. Direct importance estimation with model selection and its application to covariate shift adaptation. *Advances in Neural Information Processing Systems*. (2008) — **cité page 54**.

M. Sugiyama, S. Nakajima, H. Kashima, P. von Bünau et M. Kawanabe. Direct Importance Estimation with Model Selection and Its Application to Covariate Shift Adaptation. *Advances in Neural Information Processing Systems*. (2007) — **cité page 54**.

S. Sun, L. Mao, D. Ziang et W. Lidan. Multiview machine learning. *Springer*. (2019) — **cité page 45**.

S. Sun, J. Shawe-Taylor et L. Mao. PAC-Bayes analysis of multi-view learning. *Information Fusion*. (2017) — **cité page 46**.

C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow et R. Fergus. Intriguing properties of neural networks. *International Conference on Learning Representations*. (2014) — **cité pages 110, 112**.

N. Thiemann, C. Igel, O. Wintenberger et Y. Seldin. A Strongly Quasiconvex PAC-Bayesian Bound. *ALT*. (2017) — **cité pages 106, 128**.

R. Urner, S. Shalev-Shwartz et S. Ben-David. Access to Unlabeled Data can Speed up Prediction Time. *International Conference on Machine Learning*. (2011) — **cité page 62**.

A. van der Vaart et J. Wellner. Weak convergence and empirical processes. *Springer series in statistics. Springer*. (1996) — **cité page 27**.

L. Valiant. A Theory of the Learnable. *Communications of the ACM*. (1984) — **cité pages 13, 22**.

V. Vapnik. Statistical Learning Theory. *A Wiley-Interscience publication. Wiley*. (1998) — **cité page 25**.

V. Vapnik et A. Chervonenkis. On the Uniform Convergence of Relative Frequencies of Events to Their Probabilities. *Doklady Akademii Nauk USSR*. (1968) — **cité pages 13, 20, 22, 24**.

V. Vapnik et A. Chervonenkis. On the Uniform Convergence of Relative Frequencies of Events to Their Probabilities. *Theory of Probability and its Applications*. (1971) — **cité pages 20, 22, 24**.

P. Viallard, R. Emonet, A. Habrard, E. Morvant et V. Zantedeschi. Leveraging PAC-Bayes Theory and Gibbs Distributions for Generalization Bounds with Complexity Measures. *International Conference on Artificial Intelligence and Statistics*. (2024a) — **cité page 134**.

P. Viallard, P. Germain, A. Habrard et E. Morvant. Self-Bounding Majority Vote Learning Algorithms by the Direct Minimization of a Tight PAC-Bayesian C-Bound. *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*. (2021a) — **cité pages 91, 92, 99, 108**.

P. Viallard, P. Germain, A. Habrard et E. Morvant. A general framework for the practical disintegration of PAC-Bayesian bounds. *Machine Learning*. 2. (2024) — **cité pages 123, 144**.

P. Viallard, E. G. Vidot, A. Habrard et E. Morvant. A PAC-Bayes Analysis of Adversarial Robustness. *Advances in Neural Information Processing Systems*. (2021b) — **cité pages 110, 120**.

M. Welling et Y. W. Teh. Bayesian Learning via Stochastic Gradient Langevin Dynamics. *International Conference on Machine Learning*. (2011) — **cité page 138**.

C. K. I. Williams et M. Seeger. Using the Nyström Method to Speed Up Kernel Machines. *Advances in Neural Information Systems*. (2001) — **cité page 73**.

D. Wu, B. Wang, D. Precup et B. Boulet. Boosting based multiple kernel learning and transfer regression for electricity load forecasting. *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*. (2017) — **cité page 84**.

H. Xiao, K. Rasul et R. Vollgraf. Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms. *arXiv preprint arXiv:1708.07747*. (2017) — **cité pages 132, 142**.

H. Xu et S. Mannor. Robustness and Generalization. *Annual Conference on Learning Theory*. (2010) — **cité pages 26, 27**.

H. Xu et S. Mannor. Robustness and Generalization. *Machine Learning*. (2012) — **cité page 27**.

T. Yang, Y.-f. Li, M. Mahdavi, R. Jin et Z.-H. Zhou. Nyström Method vs Random Fourier Features: A Theoretical and Empirical Comparison. *Advances in Neural Information Systems*. (2012) — **cité page 73**.

Z. Yang, A. G. Wilson, A. J. Smola et L. Song. A la Carte - Learning Fast Kernels. *International Conference on Artificial Intelligence and Statistics*. (2015) — **cité pages 75, 86**.

D. Yin, K. Ramchandran et P. Bartlett. Rademacher Complexity for Adversarially Robust Generalization. *International Conference on Machine Learning*. (2019) — **cité page 113**.

F. X. Yu, A. T. Suresh, K. M. Choromanski, D. N. Holtmann-Rice et S. Kumar. Orthogonal Random Features. *Advances in Neural Information Systems*. (2016) — **cité page 75**.

V. Zantedeschi, R. Emonet et M. Sebban. Fast and Provably Effective Multi-view Classification with Landmark-based SVM. *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*. (2018) — **cité page 78**.

V. Zantedeschi, M. Nicolae et A. Rawat. Efficient Defenses Against Adversarial Attacks. *ACM Workshop on Artificial Intelligence and Security*. (2017) — **cité pages 110, 113, 114**.

V. Zantedeschi, P. Viallard, E. Morvant, R. Emonet, A. Habrard, P. Germain et B. Guedj. Learning stochastic majority votes by minimizing a PAC-Bayes generalization bound. *Advances in Neural Information Processing Systems*. (2021) — **cité pages 91, 92, 99, 106, 109**.

C. Zhang, L. Zhang et J. Ye. Generalization Bounds for Domain Adaptation. *Advances in Neural Information Processing Systems*. (2012) — **cité pages 54, 56**.

E. Zhong, W. Fan, Q. Yang, O. Verscheure et J. Ren. Cross Validation Framework to Choose amongst Models and Datasets for Transfer Learning. *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*. (2010) — **cité page 70**.

W. Zhou, V. Veitch, M. Austern, R. Adams et P. Orbanz. Non-vacuous Generalization Bounds at the ImageNet Scale: a PAC-Bayesian Compression Approach. *International Conference on Learning Representations*. (2019) — **cité pages 129, 132**.