



HAL
open science

Une nouvelle fonction d'agrégation intervalliste et son utilisation en tant que modèle d'apprentissage automatique pour la régression

Yassine Hmidy

► To cite this version:

Yassine Hmidy. Une nouvelle fonction d'agrégation intervalliste et son utilisation en tant que modèle d'apprentissage automatique pour la régression. Intelligence artificielle [cs.AI]. Université de Montpellier, 2023. Français. NNT : 2023UMONS039 . tel-04650562

HAL Id: tel-04650562

<https://theses.hal.science/tel-04650562v1>

Submitted on 16 Jul 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE POUR OBTENIR LE GRADE DE DOCTEUR DE L'UNIVERSITÉ DE MONTPELLIER

En Intelligence artificielle

École doctorale I2S

Unité de recherche LIRMM

Une nouvelle fonction d'agrégation intervalliste et son utilisation
comme modèle d'apprentissage automatique pour la régression

Présentée par Yassine HMIDY
le [12 décembre 2023]

Sous la direction de Olivier STRAUSS
et Agnès RICO

Devant le jury composé de

[William PUECH, Professeur des Universités]
[Christophe MARSALA, Professeur des Universités]
[Gilles MAURIS, Maître de conférence]
[Agnès RICO, Maître de conférence]

[Président du jury]
[Rapporteur]
[Rapporteur]
[Membre du jury]



UNIVERSITÉ
DE MONTPELLIER

Remerciements

Je tiens avant tout à exprimer ma profonde gratitude à mon directeur de thèse, le Professeur Olivier Strauss. La richesse de son expertise et sa perspicacité ont été des guides précieux tout au long de ce parcours. Sa patience et son soutien constant ont été des piliers sur lesquels je me suis appuyé à chaque étape. Olivier, tes enseignements ont été bien au-delà des simples orientations académiques : tu m’as appris à aborder les défis avec une résilience inébranlable et à toujours chercher l’excellence dans chacun de mes efforts.

Mes remerciements s’adressent également à ma co-encadrante, Agnès Rico. Son regard critique, sa finesse d’esprit et sa disponibilité ont enrichi ma recherche et m’ont permis d’avancer de manière constructive. Agnès, ta capacité à m’éclairer dans les moments de doute a été inestimable.

Le laboratoire LIRMM, et plus particulièrement l’équipe ICAR, ont représenté pour moi bien davantage qu’un simple espace professionnel. Ces lieux ont accueilli mes élans d’enthousiasme, mes phases de doute, mes éclairs de découverte et mes moments d’innovation. Une gratitude profonde est due à Gérard Subsol. Sa remarquable aptitude à rassembler et à galvaniser l’équipe a sculpté un environnement fertile en innovation et en esprit collaboratif, éléments clés de la réussite de mes travaux. Merci Gérard pour ta confiance indéfectible, ton soutien inébranlable et ta constante disponibilité malgré un emploi du temps chargé. Mes remerciements s’étendent également à Marc Chaumont qui a repris le flambeau avec efficacité, insufflant une nouvelle énergie à l’équipe.

À Tristan Cossin, mon co-bureau et fidèle camarade de route : nos discussions, qu’elles soient académiques ou simplement légères, ont égayé mes journées. Ces moments de complicité et d’entraide ont rendu ce voyage doctoral bien plus agréable. Ta présence a été pour moi une bouffée d’air frais dans l’atmosphère parfois tendue de la recherche.

Ma gratitude s’étend également à chaque membre de l’équipe ICAR et du laboratoire LIRMM. Chacun d’entre vous, à sa manière, a contribué à enrichir cette expérience. Vos conseils, votre soutien et votre bienveillance ont créé un environnement dans lequel j’ai pu m’épanouir.

Enfin, et non des moindres, je tiens à remercier mes parents. Votre amour inconditionnel, votre foi en moi et votre soutien indéfectible ont été le socle sur lequel j’ai construit ma vie. Cette thèse est autant le fruit de mes efforts que le reflet de votre éducation. Merci pour tout.

Résumé

Nous vivons dans un monde où les causes et les effets guident notre compréhension des phénomènes qui nous entourent. Une relation de cause à effet peut être considérée comme un système dont les entrées sont une cause et les sorties un effet. À l'ère de la profusion des données, les nombreux exemples entrées/sorties permettent de caractériser de nombreux systèmes. Pour un système d'intérêt, l'objectif est d'anticiper son comportement. Pour cela, on établit un modèle qui est une version simplifiée du système. Le but est qu'il mime cette relation entrée/sortie. Cependant, certains systèmes peuvent avoir des comportements "incohérents", produisant des sorties différentes même avec des entrées identiques. Une tentative de modélisation de tels systèmes pourrait se faire via un modèle qui proposerait en sortie un intervalle, dans le but que cet intervalle recouvre l'ensemble des sorties possibles pour une même entrée. Cette thèse propose une nouvelle fonction d'agrégation, appelée "agrégation macsum", qui prend en entrée un vecteur de valeurs précises et produit en sortie un intervalle de valeurs qui représente un ensemble convexe de sorties d'agrégation linéaire. Une particularité de ce modèle est qu'il utilise une intégrale de Choquet discrète par rapport à une fonction d'ensemble non-monotone, ce qui est peu courant. Une autre particularité est qu'il permet de représenter un ensemble convexe de modèles linéaires avec seulement un paramètre. Il allie donc simplicité d'utilisation et représentativité de systèmes incohérents. Après avoir introduit ce nouveau modèle et ses propriétés, on l'utilise comme modèle d'apprentissage en montrant sa simplicité d'utilisation. On parvient également à établir une extension de ce modèle à des entrées intervallistes, élargissant ainsi la gamme de systèmes qu'il a le potentiel de représenter.

Mots clés : Fonctions d'agrégation, Agrégation Macsum, Apprentissage automatique, Données imprécises, Intégrale de Choquet discrète

Abstract

We live in a world where causes and effects shape our understanding of the phenomena around us. A cause-and-effect relationship can be seen as a system where inputs are causes and outputs are effects. In this era of data abundance, numerous input-output examples help characterize many systems. For a system of interest, the goal is to anticipate its behavior. To achieve this, we establish a model that is a simplified version of the system. The aim is for it to mimic this input-output relationship. However, some systems may exhibit "inconsistent" behaviors, producing different outputs even with identical inputs. An attempt to model such systems could involve a model that outputs an interval, aiming for this interval to encompass all possible outputs for a given input. This thesis introduces a new aggregation function, termed "macsum aggregation," which takes as input a vector of precise values and outputs an interval of values that represents a convex set of linear aggregation outputs. One unique feature of this model is that it employs a discrete Choquet integral with respect to a non-monotonic set function, which is quite unusual. Another distinctiveness is that it allows representing a convex set of linear models using only one parameter. Thus, it combines ease of use with the representation of inconsistent systems. After introducing this new model and its properties, it is utilized as a learning model, showcasing its simplicity. We also manage to extend this model to interval-valued inputs, broadening the range of systems it potentially represents.

Keywords : Aggregation functions, Macsum aggregation, Machine learning, Imprecise data, Discrete Choquet integral

Table des matières

Remerciements	i
Résumé	ii
Abstract	iii
Table des matières	iii
Liste des figures	vi
1 Introduction générale	1
2 Fondements théoriques	5
1 Introduction	6
2 Modélisation d'un système	6
2.1 Types de systèmes basés sur la dimension des entrées-sorties	7
2.2 Types de systèmes basés sur l'étiquetage	7
2.3 La cohérence de la relation entrée-sortie	8
2.4 Précision des entrées et des sorties	8
2.5 Le modèle	9
2.5.1 Modèle paramétrique	9
2.5.2 Modèle non-paramétrique	10
3 Apprentissage supervisé d'un modèle paramétrique	11
4 Les fonctions d'agrégation comme modèle d'apprentissage	12
4.1 Préliminaires	12
4.2 L'intégration	13
4.2.1 Préliminaires	13
4.2.2 Définition de l'intégrale de Choquet discrète	15
4.3 L'intégrale de Choquet comme modèle d'apprentissage	16
4.3.1 La somme pondérée	16
4.3.2 Limites de la somme pondérée comme modèle d'apprentissage	17
4.3.3 L'intégrale de Choquet par rapport à une fonction d'ensemble non-additive	17
4.3.4 Limites de l'intégrale de Choquet comme modèle d'apprentissage	18
5 Agrégation avec sortie intervalliste	18
5.1 Calcul d'un intervalle autour d'une sortie précise d'agrégation	19
5.1.1 Intervalle de confiance	19
5.1.2 Intervalle de prédiction	19

5.1.3	Intervalle de crédibilité	20
5.2	Extension d'une agrégation aux entrées intervallistes	21
5.3	Modèles intrinsèquement intervallistes	21
5.3.1	Paramètre imprécis	22
5.3.2	Agrégation maxitive	22
6	Conclusion	24
3	Opérateur macsum et agrégation macsum	25
1	Introduction	26
2	Généralisation de la domination maxitive	26
2.1	L'opérateur macsum	27
2.2	Concavité de l'opérateur macsum	29
2.3	Domination de l'opérateur macsum sur les opérateurs additifs	30
3	Représentation d'un ensemble convexe d'agrégations linéaires	32
4	Calcul de l'agrégation macsum	34
4.1	Formules	34
4.2	Algorithmes	36
5	Conclusion	37
4	Un nouveau modèle d'apprentissage intervalliste	38
1	Introduction	39
2	Aspects préliminaires	40
2.1	Non vacuité du coeur macsumitif	40
2.2	Formulation de l'agrégation macsum facilitant la dérivation	41
2.3	Dérivation de l'agrégation macsum	46
3	Apprentissage de l'agrégation macsum	49
3.1	Descente de gradient	49
3.2	Fonction de coût pour une sortie intervallistes	49
3.3	Expérimentation	51
4	Conclusion	57
5	Extension et granularité de l'agrégation macsum	58
1	Introduction	59
2	Extension de l'agrégation macsum aux entrées intervallistes	59
2.1	Préliminaires	59
2.2	Extension disjonctive aux entrées intervallistes	60
2.3	Extension conjonctive aux entrées intervallistes	64
2.4	Exemple : Prévission des températures dans une région à topographie variée	67
3	Granularité du coeur macsumitif d'un paramètre	68
4	Conclusion	73
	Conclusion et perspectives	74
	Bibliographie	76

Liste des figures

4.1	Image acquise par un capteur non invariant par décalage.	52
4.2	Quatre des 2000 images utilisées pour cette expérience.	52
4.3	Sous-échantillonnage de la 3ème image de la Figure (4.2)	52
4.4	PSF utilisée pour sous-échantillonner les images : centrale (a) et bordure (b). . .	53
4.5	Images illustrées dans la Figure (4.2) après sous-échantillonnage.	53
4.6	Après l'apprentissage, le noyau $\hat{\psi}$ (a) et le noyau $\hat{\varphi}$ (b)	55
4.7	Convergence de l'apprentissage additif (a) et de l'apprentissage macsum (b). . . .	55
4.8	Convergence de l'apprentissage macsum à travers des distances étendues.	55
4.9	Corrélation entre l'écart moyen et la distance moyenne au centre	56
4.10	Image de l'écart (a) et de la distance au centre (b) pour la 2 ^{ème} image de la Figure (4.5)	57

Chapitre 1

Introduction générale

Objectifs

L'introduction de ce manuscrit vise à dépeindre le paysage contemporain marqué par un foisonnement des données issues de diverses sources et le défi inhérent de leur modélisation pour diverses exploitations. De manière spécifique, l'accent est mis sur l'étude des systèmes "multi-entrée mono-sortie", dont la complexité de leurs mécanismes internes, parfois mal élucidés, rend la prédiction délicate. En ce sens, le modèle, qui offre une version simplifiée du système réel, est privilégié pour en anticiper le comportement, sa fiabilité étant mesurée par sa capacité à reproduire le comportement du système original. L'apprentissage automatique, notamment supervisé, est sollicité pour peaufiner ces modèles. Un autre point nodal de cette introduction est la présentation de la fonction d'agrégation maxitive, dont la particularité est de fournir un intervalle de valeurs et non une valeur unique. L'ambition de cette thèse étant de la généraliser pour modéliser des systèmes incohérents. Ainsi, ce prélude a pour dessein de jeter les bases de la thèse, qui ambitionne de définir et d'explorer une agrégation intervalliste et son utilisation en tant que modèle d'apprentissage automatique.

Nous vivons dans une époque caractérisée par une profusion de données. Cette époque est souvent qualifiée d'ère du numérique. Cette abondance d'informations provient d'une multitude de sources : des appareils connectés que nous utilisons quotidiennement, des plateformes de médias sociaux où nous partageons nos moments, des entreprises qui recueillent des informations pour améliorer leurs services, ou encore des capteurs déployés dans presque toutes les industries. L'analyse et l'utilisation de ces données à des fins scientifiques, commerciales ou autres passent par une modélisation de processus décisionnel ou agrégatif. Nous appelons "système" une entité qui génère des données à partir d'autres données. Les systèmes peuvent générer des données de n'importe quelle dimension. Dans cette thèse, nous nous intéressons aux systèmes dits "multi-entrée mono-sortie" qui génèrent une valeur réelle à partir d'un vecteur de plusieurs valeurs réelles. Certains systèmes reposent sur des mécanismes sous-jacents qui ne sont pas toujours entièrement connus. Cela peut être dû à des perturbations lors des mesures ou à des variables cachées qui influencent la sortie pour certaines valeurs d'entrée. Par exemple, lors de la mesure de la température d'un moteur, des erreurs peuvent survenir à cause des interférences électromagnétiques. De même, dans un système écologique, l'impact réel des prédateurs sur une population donnée peut être influencé par des variables cachées comme les changements climatiques ou la disponibilité de la nourriture. Il est alors difficile de pouvoir prédire correctement quelle sera l'évolution de ce système, ce qui peut être d'une importance majeure, par exemple en écologie. Pour mieux anticiper le comportement de ces systèmes, nous nous appuyons sur ce que l'on appelle un modèle. Un modèle est une représentation simplifiée d'un système, conçue pour permettre des prévisions de la sortie d'un système pour de nouvelles données d'entrées. Un modèle sera considéré comme fiable si son comportement entrée/sortie est proche du système modélisé, ou encore si, pour une même entrée, la sortie du modèle est proche de la sortie réelle. Les systèmes multi-entrées mono-sortie peuvent être modélisés par des fonctions d'agrégations qui sont des fonctions qui à un vecteur de réels associent une valeur réelle. Ces fonctions sont généralement paramétriques, ce qui implique que le calcul de leur résultat fait intervenir un ensemble de paramètres, souvent désignés sous le terme de "poids".

Pour modéliser un système, la première étape consiste à analyser la relation causale entre des données d'entrée introduites dans un système et les sorties qu'il génère. Cette analyse nous aide à mieux comprendre cette relation. Sur la base de cette compréhension, et de ce que nous savons du système, une fonction d'agrégation appropriée est choisie. Cette fonction est censée correspondre, du moins approximativement, à la manière dont le système transforme ses entrées en sorties. Après avoir établi cette fonction initiale, les données sont utilisées pour l'ajuster. Ce que l'on entend par "ajuster", dans le cadre d'un modèle qui serait une fonction d'agrégation paramétrique, est le fait d'estimer des paramètres de telle sorte que, pour chaque même entrée, la sortie du modèle soit le plus proche possible de celle du système. L'apprentissage automatique est une technique d'ajustement dont l'utilisation est en forte croissance depuis ce que l'on nomme la révolution du "big data". On entend par apprentissage l'utilisation d'algorithmes pour permettre à un modèle de s'ajuster à partir d'un ensemble de données.

Une fonction d'agrégation particulièrement utilisée comme modèle d'apprentissage est la somme pondérée, et ce dans de nombreux domaines, car elle permet d'attribuer des poids spé-

cifiques à chaque entrée, garantissant la prise en compte de l'influence de chaque entrée sur le résultat final. De plus, la somme pondérée est facilement dérivable, ce qui est essentiel pour l'utilisation des algorithmes d'optimisation basés sur la descente de gradient.

Dans le monde réel, tous les systèmes ne sont pas toujours cohérents, c'est-à-dire qu'une même entrée peut être associée à différentes sorties. Prenons, par exemple, un système qui associe des caractéristiques individuelles à une opinion. Deux personnes ayant des caractéristiques similaires, comme la même formation académique, le même âge et vivant dans la même ville, pourraient avoir des opinions radicalement différentes sur un sujet. Ces différences d'opinions pourraient être influencées par des facteurs cachés tels que leur expérience personnelle, des influences sociales subtiles ou des événements marquants de leur vie passée. Ces éléments inattendus ajoutent une couche de complexité à une prédiction, des comportements et des opinions, basée uniquement sur des caractéristiques observables. Un système dit "incohérent" sera donc un système qui à une même entrée peut associer différentes sorties. Une des ambitions de cette thèse est de proposer un modèle pour de tels systèmes. Ce modèle que nous proposons calcule un intervalle de sortie à partir d'entrées précises.

Un intervalle réel est un ensemble continu de nombres réels compris entre deux bornes : une borne inférieure et une borne supérieure. Nous appelons "fonction d'agrégation intervalliste" une fonction qui, au lieu de fournir une valeur unique, donne un intervalle de valeurs. Cette thèse porte sur une nouvelle fonction d'agrégation intervalliste nommée l'agrégation macsum et son utilisation en tant que modèle d'apprentissage. Les bases de cette approche ont été jetées par des travaux antérieurs de Loquin et al. [Loquin, 2008a]. Ils ont proposé une agrégation intervalliste qui, plutôt que de fournir une valeur de sortie unique, génère un intervalle qui représente un ensemble convexe de sommes pondérées. Cependant, cette proposition est accompagnée de contraintes : les poids associés à cette somme doivent être positifs et sommer à 1. Ce qui signifie que, pour utiliser cette fonction en tant que modèle d'apprentissage, il faudrait contraindre les poids à respecter ces deux conditions lors de la phase d'ajustement.

C'est ici que se dessine la perspective de cette thèse. Le but est de généraliser l'agrégation intervalliste proposé par Loquin et al. afin d'avoir un modèle d'apprentissage sans contraintes sur les poids.

La présente thèse se divise en quatre chapitres, structurés de manière à présenter une nouvelle fonction d'agrégation intervalliste, ainsi que son application dans le domaine de l'apprentissage automatique. Le plan de la dissertation est le suivant :

1. **Fondements théoriques** : dans ce chapitre, nous positionnons notre travail par rapport à la littérature existante tout en y présentant également les définitions et propriétés constituant les bases sur lesquelles cette thèse repose.
2. **L'opérateur macsum et l'agrégation macsum** : dans ce chapitre, nous introduisons une nouvelle fonction d'ensemble ainsi qu'une nouvelle fonction d'agrégation intervalliste, nommée "agrégation macsum", qui prend en entrée un vecteur de données et un vecteur de paramètres de même dimension et donne en sortie un intervalle.
3. **Application de l'agrégation macsum à l'apprentissage automatique** : ce chapitre se consacre à l'application de l'agrégation macsum à l'apprentissage automatique.

Une expérience basée sur le traitement d'image est également présentée pour évaluer les avantages et les limitations de l'agrégation macsum dans l'apprentissage automatique.

4. **Extension et granularité de l'agrégation macsum** : dans ce chapitre, nous proposons deux extensions de l'agrégation macsum aux entrées intervallistes, puis nous définissons un indice de cardinalité permettant d'évaluer le potentiel de représentativité d'une agrégation macsum.

Chapitre 2

Fondements théoriques

Objectifs

Ce chapitre s'intéresse à l'agrégation des données, en mettant l'accent sur son application en apprentissage automatique. Il souligne l'importance des fonctions d'agrégation, en tant que modèle d'apprentissage pertinent pour les systèmes multi-entrée mono-sortie. En se centrant sur l'intégrale de Choquet discrète comme généralisation des fonctions d'agrégation linéaire, il met en avant ses propriétés et son application en apprentissage automatique en discutant de ses avantages et inconvénients. Ce chapitre introduit également la notion de modèles intervallistes et met l'accent sur l'agrégation maxitive qui est une agrégation intervalliste, utilisant l'intégrale de Choquet discrète et, qui constitue le point de départ de cette thèse. Finalement, ce chapitre établit les fondements théoriques sur lesquels repose cette thèse et permet de la situer dans la littérature scientifique.

Sommaire

1	Introduction	6
2	Modélisation d'un système	6
2.1	Types de systèmes basés sur la dimension des entrées-sorties	7
2.2	Types de systèmes basés sur l'étiquetage	7
2.3	La cohérence de la relation entrée-sortie	8
2.4	Précision des entrées et des sorties	8
2.5	Le modèle	9
3	Apprentissage supervisé d'un modèle paramétrique	11
4	Les fonctions d'agrégation comme modèle d'apprentissage	12
4.1	Préliminaires	12
4.2	L'intégration	13
4.3	L'intégrale de Choquet comme modèle d'apprentissage	16
5	Agrégation avec sortie intervalliste	18
5.1	Calcul d'un intervalle autour d'une sortie précise d'agrégation	19
5.2	Extension d'une agrégation aux entrées intervallistes	21
5.3	Modèles intrinsèquement intervallistes	21
6	Conclusion	24

1 Introduction

Un grand nombre de phénomènes peuvent être vus comme une relation entre entrée et sortie en raison de la nature intrinsèquement causale de notre univers. Cette relation est ce que l'on qualifiera de "système". Dans un système donné, une entrée (ou une cause) génère une sortie (ou un effet) selon certaines règles ou lois qui peuvent être déterministes ou probabilistes. Cette dynamique d'entrée-sortie peut être observée à de nombreux niveaux, des interactions moléculaires dans une cellule aux mouvements des marchés financiers mondiaux. La modélisation d'un système est un processus qui consiste à créer une représentation simplifiée du système étudié et est conçue pour simuler son comportement. Cette représentation est ce que l'on appelle un "modèle". La modélisation est un élément-clé permettant d'anticiper les sorties produites par ces systèmes en fonction de nouvelles entrées. Elle offre ainsi la possibilité d'ajustements proactifs et d'une meilleure gestion des ressources et des situations. Un exemple probant est celui des modèles météorologiques. Ces modèles utilisent une multitude d'entrées telles que les données satellitaires, les observations terrestres et océaniques, et les mesures atmosphériques pour prédire les conditions météorologiques futures. L'anticipation de phénomènes météorologiques extrêmes, tels que les ouragans ou les tempêtes de neige, est cruciale pour prévenir les dommages, organiser les évacuations en temps utile, et planifier les interventions d'urgence, sauvant ainsi des vies et des ressources.

Plus les sorties d'un modèle sont proches des sorties d'un système pour les mêmes entrées, plus ce modèle est dit "fiable". Face à une vaste diversité de systèmes, le choix du modèle adéquat se révèle primordial. Parmi les modèles, on distinguera ceux qui ont comme sortie une valeur réelle précise de ceux qui donnent un intervalle réel. Ces modèles sont choisis selon certaines propriétés du système étudié. On distingue également les modèles paramétriques, dont le calcul fait intervenir un vecteur de paramètres, des modèles non-paramétriques. Les modèles paramétriques permettent, via un processus itératif, un ajustement méticuleux des paramètres pour épouser au plus près le comportement du système. Cette étape d'ajustement succède à la phase initiale de modélisation et constitue un jalon essentiel pour augmenter la fiabilité du modèle. L'apprentissage automatique se présente comme une des techniques principales permettant de réaliser cet ajustement. Parmi les différentes variantes, l'apprentissage supervisé est utilisé lorsqu'on possède un jeu de données associant des entrées du système à des sorties connues. Cette technique permet donc d'ajuster les paramètres du modèle à partir d'exemples d'entrée/sortie du système. Dans ce chapitre, nous explorons la modélisation des systèmes en mettant l'accent sur les modèles paramétriques intervallistes et l'apprentissage supervisé comme technique d'ajustement des paramètres.

2 Modélisation d'un système

Un système peut être conceptualisé comme une relation causale. Si nous nommons ce système F , la relation causale peut être exprimée sous une forme fonctionnelle comme $y = F(x)$, où x provient d'un ensemble X et y d'un ensemble Y .

Chaque système, selon ses caractéristiques et propriétés intrinsèques, appartient à une catégorie spécifique. Cette classification permet de faciliter le choix du modèle qui correspond à un système.

Dans cette section, notre objectif est d'élaborer une typologie claire et structurée des systèmes. Cette démarche a pour but de situer notre travail.

2.1 Types de systèmes basés sur la dimension des entrées-sorties

L'une des façons de classer les systèmes est en fonction de la dimensionnalité de leurs vecteurs d'entrée et de sortie.

- **Système mono entrée mono sortie** : ce système associe un vecteur d'entrée de dimension 1 à un vecteur de sortie de dimension 1. Exemple : un baromètre qui reçoit la pression atmosphérique et donne une température [Moya, 2022].
- **Système mono entrée multi sortie** : un vecteur d'entrée de dimension 1 est associé à un vecteur de sortie de dimension supérieure à 1. Exemple : l'impact de l'augmentation du prix du pétrole. Cette unique modification entraîne plusieurs sorties : elle provoque une augmentation du coût des carburants tels que l'essence et le diesel, induit une hausse des prix des biens transportés, entraîne une réduction des déplacements en voiture par les consommateurs, cause une réduction des bénéfices pour les entreprises gourmandes en énergie, stimule une augmentation de la demande pour les véhicules électriques ou hybrides, et peut même déclencher une récession économique dans les pays fortement dépendants du pétrole comme source principale d'énergie. Dans cet exemple, une seule cause (l'augmentation des prix du pétrole) entraîne plusieurs effets ou sorties. Chacune de ces sorties est directement ou indirectement influencée par l'entrée, illustrant ainsi une relation causale mono entrée, multi sortie. [Ferderer, 1996].
- **Système multi entrée multi sortie** : un vecteur d'entrée de dimension supérieure à 1 est associé à un vecteur de sortie de dimension supérieure à 1. Exemple : un simulateur de vol qui associe des données telles que la vitesse, l'angle d'attaque et le taux de roulis à des informations sur l'altitude, la trajectoire et la vitesse verticale [Baarspul, 1990].
- **Système multi entrée mono sortie** : un vecteur d'entrée de dimension supérieure à 1 est lié à un vecteur de sortie de dimension 1. Exemple : la relation entre le cours des actions sur le marché financier et des indicateurs comme le PIB, le taux de chômage et les taux d'intérêt [Agrawal, 2013].

2.2 Types de systèmes basés sur l'étiquetage

- **Système étiqueté** : chaque entrée est associée à une sortie qui est connue. Exemple : un système qui à chaque photo d'animal donne le nom de l'animal [Cunningham, 2008].
- **Système sans étiquette** : les entrées sont fournies sans étiquettes claires, rendant indéterminée la nature ou le type de sortie. Exemple : un système qui reçoit des photos d'animaux et regroupe ces photos en fonction de similitudes visuelles sans connaître ou

nommer spécifiquement quel animal est présenté dans chaque photo [Manohar, 2016]. Dans cet exemple, il est difficile de déterminer les classes à l'avance et donc d'associer à chaque entrée une étiquette.

2.3 La cohérence de la relation entrée-sortie

Lorsque l'on analyse ou modélise un système, il est important de comprendre la relation entre ses entrées et ses sorties, en particulier en termes de cohérence. Un système peut être classé comme cohérent ou incohérent selon la nature de cette relation.

- **Systèmes cohérents** : dans ces systèmes, une entrée donnée génère toujours la même sortie, si le système est dans le même état. Par exemple, une calculatrice est un système cohérent, car pour une opération donnée (comme $2+2$), elle fournira toujours le même résultat (4).
- **Systèmes incohérents** : à l'inverse, dans ces systèmes, une même entrée peut donner lieu à différentes sorties selon différents facteurs. Par exemple, un système qui associe aux caractéristiques d'individus leur opinion, deux individus ayant les mêmes caractéristiques peuvent avoir une opinion différente.

Déterminer si un système est cohérent ou non peut être délicat en fonction de la disponibilité des données et de l'étendue des situations représentées par les exemples d'entrée-sortie. En effet, un système peut montrer son incohérence seulement sur certaines entrées bien particulières.

2.4 Précision des entrées et des sorties

Nous appelons une "donnée précise" une valeur réelle, i.e. un élément de l'ensemble des réels \mathbb{R} . En revanche, une "donnée imprécise" est un ensemble de valeurs réelles. Un cas particulier d'une donnée imprécise est un intervalle. Outre la cohérence de la relation entrée-sortie, les systèmes peuvent également être caractérisés en fonction de la précision de leurs entrées et sorties. Selon cette caractéristique, on peut classer les systèmes en quatre catégories principales :

- **Systèmes à entrées et sorties précises** : ces systèmes reçoivent des données d'entrée précises et produisent des résultats précis. Un exemple pourrait être un convertisseur de température de Celsius à Fahrenheit. Pour une entrée donnée en degrés Celsius, il fournit une sortie en degrés Fahrenheit selon la formule $F = (C \times \frac{9}{5}) + 32$. Ainsi, si on fournit 25°C comme entrée, la sortie sera 77°F .
- **Systèmes à entrées précises et sorties intervallistes** : ces systèmes reçoivent des données d'entrée précises, mais la sortie est donnée sous forme d'intervalle. Par exemple, un système qui à partir de caractéristiques précises sur un bien immobilier, tel que sa superficie et son année de construction, donne un intervalle de prix correspondant à sa valeur.
- **Systèmes à entrées intervallistes et sorties précises** : ces systèmes reçoivent des données d'entrée sous forme d'intervalle, mais produisent une sortie précise. Par exemple

un système qui reçoit une estimation de poids (par exemple, entre 50 et 55 kg) et détermine avec précision la quantité de médicaments à administrer [Chen, 2013].

- **Systèmes à entrées et sorties intervallistes** : ces systèmes fonctionnent avec des données d'entrée et de sortie imprécises. Par exemple, un système économique pourrait recevoir des prévisions de croissance sous forme d'intervalle (par exemple, entre 2% et 4%) et produire une estimation de revenu également sous forme d'intervalle [Chudý, 2020].

La précision des entrées et des sorties est donc un autre aspect important à considérer lors de la modélisation.

2.5 Le modèle

Un modèle est une représentation mathématique conçue pour imiter un système. L'objectif principal d'un modèle est de fournir des sorties qui se rapprochent le plus possible des sorties du système pour des entrées données. Un modèle est généralement plus simple ou structuré différemment du système qu'il représente [Kapur, 1988]. La modélisation se divise principalement en deux branches majeures : la régression et la classification. Un modèle de régression est dédié aux systèmes produisant des valeurs continues. Un modèle de classification est employé pour représenter des systèmes aboutissant à des résultats discrets, généralement catégorisés. Dans ces deux branches, il existe des modèles paramétriques et des modèles non-paramétriques. C'est cet aspect-là que nous détaillons dans cette section.

2.5.1 Modèle paramétrique

Dans un modèle paramétrique, la forme de la fonction qui relie les entrées aux sorties est déterminée à l'avance [Hastie, 2001]. Les données sont ensuite utilisées pour estimer les paramètres de cette fonction. Cette approche dépend grandement des hypothèses de base sur la forme de la relation entre les entrées et les sorties [Janssen, 2015].

Exemples de modèles paramétriques :

- **Bayes naïf** : le modèle du Bayes naïf est un modèle de classification paramétrique. Cet algorithme porte l'adjectif "naïf", car il postule l'indépendance complète entre les entrées [Rish, 2001]. En d'autres termes, dans le processus de décision de classification, le Bayes naïf évalue chaque entrée de manière isolée, sans prendre en compte une éventuelle interdépendance entre elles.
- **Réseau de neurones artificiel** : les réseaux de neurones artificiels, inspirés du cerveau humain [Eluyode, 2013], sont considérés comme des modèles paramétriques, car ils ont des paramètres fixes à apprendre. Ils sont capables de mimer des relations non-linéaires complexes. Leur structure est organisée en couches : d'entrée, cachées et de sortie. Chaque neurone effectue une somme pondérée des entrées, applique une fonction non-linéaire et transmet le résultat aux neurones de la couche suivante. Ils sont largement utilisés pour des tâches telles que la classification et la régression [Goodfellow, 2016].

2.5.2 Modèle non-paramétrique

Contrairement aux modèles paramétriques, qui sont caractérisés par un nombre déterminé de paramètres, les modèles non-paramétriques gagnent en complexité à mesure que la quantité de données augmentent [Gershman, 2012]. Plus il y a de données, plus le modèle peut s'adapter aux tendances observées dans les données : une propriété que nous appelons ici "flexibilité".

Exemples de modèles non-paramétriques :

- **Arbres de décision** : un arbre de décision est un type de graphe particulier pouvant être utilisé pour la classification. Dans la terminologie des graphes :
 - Un *nœud* représente une décision ou un test sur une caractéristique.
 - Une *arête* (ou branche) représente le résultat d'un test et connecte deux nœuds.
 - Une *feuille* est un nœud terminal, c'est-à-dire un nœud qui n'a pas de descendants. Dans le contexte d'un arbre de décision, une feuille représente souvent une classification ou une décision finale.

Un arbre de décision est un graphe dit "acyclique orienté". C'est une structure de graphe où les arêtes ont une direction (c'est-à-dire qu'elles vont d'un nœud source à un nœud cible) et où il est impossible de revenir au même nœud en suivant les arêtes dans leur direction, garantissant ainsi qu'il n'y a pas de cycles. Dans le contexte des arbres de décision, cela signifie qu'une décision ou un test ne peut pas conduire à une décision ou un test précédent, assurant une structure d'arbre cohérente. Considérant un ensemble de données, l'arbre de décision divise cet ensemble en sous-ensembles en fonction de tests effectués sur des caractéristiques, conduisant à une structure de nœuds, d'arêtes et de feuilles. Lorsque l'ensemble de données est volumineux l'arbre peut devenir très dense, avec de nombreux nœuds et arêtes, ce qui entraîne une complexité élevée. Pour simplifier l'arbre, une technique appelée "élagage" est utilisée, qui consiste à supprimer certains nœuds et arêtes pour réduire la complexité de l'arbre [Kingsford, 2008].

- **K-plus proches voisins (K-NN)** : l'algorithme K-NN fonctionne en calculant la distance entre un échantillon des données et le reste des données. Les K observations les plus proches sont sélectionnées, et la classe la plus fréquente parmi ces K observations est attribuée au nouvel échantillon [Guo, 2003].

En résumé, les modèles non-paramétriques représentent une catégorie importante en modélisation qui n'impose pas de forme structurelle prédéfinie pour la relation entre les variables d'entrée et de sortie. À l'inverse, les modèles paramétriques sont associés à un ensemble fini et souvent fixe de paramètres. Ces paramètres sont ensuite ajustés pour que le modèle imite au mieux le système modélisé. L'étape de sélection de la fonction paramétrique appropriée est cruciale et connue sous le nom d'étape de modélisation [Janssen, 2015]. Après celle-ci, vient l'étape non moins importante de l'ajustement des paramètres. C'est dans ce contexte que l'apprentissage supervisé se révèle être un outil précieux. Ce champ d'étude offre des techniques efficaces et éprouvées pour ajuster les paramètres des modèles paramétriques, permettant ainsi d'augmenter leur fiabilité.

3 Apprentissage supervisé d'un modèle paramétrique

L'apprentissage supervisé est une méthode qui consiste à utiliser des données pour ajuster les paramètres qui minimisent l'écart entre les prédictions d'un modèle paramétrique et les sorties d'un système [El Naqa, 2015]. Dans cette optique, l'apprentissage supervisé est une branche particulière de l'apprentissage automatique qui se concentre sur l'ajustement des paramètres des modèles des systèmes étiquetés [Hastie, 2001]. Dans ces systèmes, chaque entrée est associée à une sortie clairement définie, offrant un cadre pour entraîner un modèle à prévoir ces étiquettes.

Considérons un système multi-entrée mono-sorties qui produit une valeur réelle $y \in \mathbb{R}$ pour toute entrée $x \in \mathbb{R}^n$. Supposons que le lien causal entre x et y est représenté par une fonction inconnue sous-jacente F . Cette fonction F représente le système que l'on veut modéliser. Supposons maintenant que nous ayons recueilli un ensemble de données de $\mathcal{D} \subseteq \mathbb{R} \times \mathbb{R}^n$ de m couples entrée-sortie du système F . Soit $\psi \in \mathbb{R}^p$ un vecteur de paramètres, à partir de ces m couples entrées-sorties de la fonction $F : \mathbb{R}^n \rightarrow \mathbb{R}$, une fois que l'on a sélectionné une fonction paramétrique $f_\psi : \mathbb{R}^n \rightarrow \mathbb{R}$ pour modéliser F . L'idée est d'ajuster ses paramètres de manière à ce que son comportement soit le plus proche possible de celui de F . C'est ce qu'on appelle "apprendre" F . Ainsi, l'apprentissage supervisé repose sur l'utilisation d'un ensemble d'apprentissage

$$\mathcal{D} = \{(x^1, y^1), (x^2, y^2), \dots, (x^m, y^m)\} \subseteq \mathbb{R} \times \mathbb{R}^n$$

où chaque x^i représente une entrée et y^i sa sortie ou étiquette correspondante, c'est-à-dire que pour un système F nous avons $F(x^i) = y^i \forall i \in \{1, \dots, m\}$.

Dans le contexte de la modélisation, il est important de vérifier l'aptitude du modèle à "généraliser", c'est-à-dire générer, à partir de nouvelles entrées, des sorties proches de celles du système modélisé. Pour ce faire, l'ensemble d'apprentissage utilisé est généralement divisé en trois ensembles distincts : une base d'entraînement, une base de contrôle et une base de validation. La base d'entraînement et la base de contrôle sont utilisées pour ajuster le modèle. Une fois le modèle élaboré, il est ensuite testé sur la base de validation, puis finalement, il est validé comme étant fiable ou non à l'issue de l'étude de ses performances sur cette base. Cela permet d'évaluer sa capacité à produire des résultats fiables sur des données non utilisées lors de son ajustement.

Formellement, les prédictions du modèle $f_\psi(x^i)$ sont faites en fonction des paramètres contenus dans le vecteur ψ , qui sont ajustés de manière à ce que la prédiction s'approche le plus possible de la sortie du système y^i .

$$f_\psi(x^i) \approx y^i.$$

La fiabilité est mesurée par une fonction de coût L . Pour les problèmes de régression, c'est la perte quadratique qui est le plus souvent utilisé :

$$L(f_\psi(x^i), y^i) = (f_\psi(x^i) - y^i)^2.$$

L'objectif est de trouver le vecteur de paramètres $\hat{\psi}$ qui minimise cette fonction de coût sur

l'ensemble d'apprentissage :

$$\hat{\psi} = \arg \min_{\psi} \sum_{i=1}^m L(f_{\psi}(x^i), y^i).$$

Cependant, un problème majeur peut survenir lors de cette étape : le surapprentissage. Ce phénomène se produit lorsque le modèle apprend trop bien la base d'entraînement, au détriment de sa capacité à généraliser sur de nouvelles données. Ce problème peut être détecté en observant une détérioration des performances du modèle sur une base de contrôle. Pour éviter le surapprentissage, il est courant de stopper le processus d'apprentissage avant une minimisation globale si l'on observe une dégradation des performances sur la base de contrôle. Cette technique est connue sous le nom d'arrêt précoce (early stopping).

La sélection d'un modèle approprié est cruciale : un modèle trop simple peut ne pas saisir la complexité des données, tandis qu'un modèle trop complexe risque de surapprendre sur l'ensemble d'apprentissage et d'être moins fiable sur de nouvelles données. Nous avons vu précédemment que différents modèles offrent une vaste palette d'approches adaptées à divers types de systèmes. La section suivante traite des modèles paramétriques spécifiquement adaptés pour les systèmes multi-entrées mono-sortie.

4 Les fonctions d'agrégation comme modèle d'apprentissage

Une fonction d'agrégation est une fonction qui prend en entrée plusieurs valeurs et les combine pour produire une seule valeur agrégée en sortie [Grabisch, 2009]. L'objectif d'une fonction d'agrégation est de fournir une représentation synthétique d'un ensemble de données.

4.1 Préliminaires

Soit \mathcal{A} une fonction d'agrégation. Les propriétés mathématiques élémentaires des fonctions d'agrégations comprennent l'invariance par permutation, la continuité, l'homogénéité et la monotonie [Grabisch, 2009]. Ces propriétés sont définies comme suit :

- **Invariance par permutation** : une fonction d'agrégation \mathcal{A} est invariante par permutation si pour tout $x_1, x_2, \dots, x_n \in \mathbb{R}$ et toute permutation ϕ de l'ensemble $1, 2, \dots, n$, on a $\mathcal{A}(x_1, x_2, \dots, x_n) = \mathcal{A}(x_{\phi(1)}, x_{\phi(2)}, \dots, x_{\phi(n)})$.
- **Continuité** : une fonction d'agrégation \mathcal{A} est dite continue en un point $x = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$ si, pour tout $\epsilon > 0$, il existe un $\delta > 0$ tel que pour tout vecteur $z = (z_1, z_2, \dots, z_n) \in \mathbb{R}^n$ dont la distance à x (mesurée par la norme, notée $\|z - x\|$, qui est une manière de quantifier la distance entre deux vecteurs) est inférieure à δ , la variation de la valeur de \mathcal{A} est alors inférieure à ϵ . Plus formellement, une fonction d'agrégation \mathcal{A} est dite continue en un point $x = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$ si, pour tout $\epsilon > 0$, il existe un $\delta > 0$ tel que pour tout $z = (z_1, z_2, \dots, z_n) \in \mathbb{R}^n$ vérifiant $\|z - x\| < \delta$, on ait $|\mathcal{A}(z) - \mathcal{A}(x)| < \epsilon$. Si \mathcal{A} est continue en tout point de \mathbb{R}^n , alors on dit que \mathcal{A} est continue sur \mathbb{R}^n .
- **Homogénéité** : une fonction d'agrégation \mathcal{A} est homogène de degré r si, pour tout $x_1, x_2, \dots, x_n \in \mathbb{R}$ et tout $c > 0$, on a $\mathcal{A}(cx_1, cx_2, \dots, cx_n) = c^r \mathcal{A}(x_1, x_2, \dots, x_n)$.

- **Monotonie** : une fonction d'agrégation \mathcal{A} est monotone si, pour tout $x_1, x_2, \dots, x_n \in \mathbb{R}$ avec $x_1 \leq x_2 \leq \dots \leq x_n$, alors $\mathcal{A}(x_1, x_2, \dots, x_n) \leq \mathcal{A}(x_{\phi(1)}, x_{\phi(2)}, \dots, x_{\phi(n)})$ pour toute permutation ϕ de l'ensemble $1, 2, \dots, n$.

Dans le contexte de la modélisation et de l'apprentissage automatique, les fonctions d'agrégation peuvent servir de modèles pour des systèmes multi-entrées mono-sortie [Beliakov, 2007]. Les fonctions d'agrégation avec une sortie précise sont des fonctions qui, à partir de plusieurs entrées, produisent une unique valeur réelle comme sortie. Ces fonctions sont particulièrement adaptées pour modéliser des systèmes cohérents. Contrairement aux fonctions d'agrégation avec sortie imprécise qui peuvent produire divers types de sorties (comme un intervalle qui est une plage de valeurs réelles).

4.2 L'intégration

L'intégration est un outil de calcul fondamental des mathématiques. Cette notion d'intégration repose sur celle de "fonction d'ensemble" qui est une fonction qui à un sous-ensemble d'un ensemble de références associe une valeur réelle. Lorsqu'une fonction d'ensemble s'applique à un ensemble dénombrable, on la qualifie de discrète.

L'intégrale de Choquet a été introduite comme une généralisation de l'intégrale de Lebesgue qui fut une avancée significative en facilitant l'intégration par rapport à des fonctions d'ensembles spécifiques dénommées "mesure" [Denneberg, 1994].

Dans cette section, nous détaillons les propriétés d'une fonction d'agrégation avec sortie précise nommée intégrale de Choquet discrète qui est un pilier du modèle paramétrique que nous introduisons dans cette thèse.

4.2.1 Préliminaires

Dans le vaste domaine des mathématiques, quantifier est primordial. C'est dans cette quête de quantification que la fonction d'ensemble trouve ses racines. Tout au long de cette thèse, $\Omega = \{1, 2, \dots, n\}$ désignera un ensemble fini d'entiers naturels appelé ensemble de référence. Une fonction d'ensemble $\mu : 2^\Omega \rightarrow \mathbb{R}$ donne une valeur réelle à chaque sous-ensemble de Ω .

Definition 4.1. Une fonction d'ensemble μ sur Ω est dite **additive** si elle respecte l'axiome d'additivité de Kolmogorov [Barone, 1978], i.e. pour tous sous-ensembles A et B de Ω :

$$\mu(A \cup B) + \mu(A \cap B) = \mu(A) + \mu(B).$$

Definition 4.2. Soit μ une fonction d'ensemble sur Ω , un ensemble $A \subseteq \Omega$ est dite **μ -décomposable**, s'il existe deux sous-ensembles $B, C \subseteq A$ tel que $B \cup C = A$ et $\mu(A) = \mu(B) + \mu(C)$.

Definition 4.3. Une fonction d'ensemble μ sur Ω est dite **k -additive** si, tout ensemble $A \subseteq \Omega$ tel que A possède strictement plus de k éléments est μ -décomposable et s'il existe au moins un sous-ensemble A avec exactement k éléments qui ne l'est pas.

Definition 4.4. Une fonction d'ensemble μ sur Ω est dite **normalisée** sur Ω si elle satisfait les conditions suivantes :

$$\mu(\emptyset) = 0, \quad \mu(\Omega) = 1.$$

Definition 4.5. Une fonction d'ensemble μ sur Ω est dite **monotone** si, pour tous $A, B \subseteq \Omega$ tels que $A \subseteq B$, elle satisfait la condition suivante :

$$\mu(A) \leq \mu(B).$$

Definition 4.6. Une fonction d'ensemble est dite **maxitive** si, pour tous sous-ensembles $A, B \subseteq \Omega$, elle respecte la condition suivante :

$$\mu(A \cup B) = \max(\mu(A), \mu(B)).$$

Definition 4.7. Une fonction d'ensemble μ sur Ω est dite **concave** (ou sous-modulaire) si, pour tous $A, B \subseteq \Omega$:

$$\mu(A \cup B) + \mu(A \cap B) \leq \mu(A) + \mu(B).$$

Definition 4.8. Une fonction d'ensemble μ sur Ω est dite **convexe** (ou super-modulaire) si, pour tout $A, B \subseteq \Omega$:

$$\mu(A \cup B) + \mu(A \cap B) \geq \mu(A) + \mu(B).$$

Definition 4.9. Une fonction d'ensemble μ sur Ω est appelée **mesure** si elle est additive et qu'elle respecte les propriétés suivantes :

- *Non-négativité* : $\forall A \subseteq \Omega, \mu(A) \geq 0$.
- *Nullité de l'ensemble vide* : $\mu(\emptyset) = 0$.

Une mesure normalisée est appelée mesure de probabilité.

La théorie de l'intégration de Lebesgue s'appuie sur le concept de mesure d'un ensemble, qui généralise la notion de longueur d'un intervalle réel.

Definition 4.10. Une **capacité** est une fonction d'ensemble μ sur Ω qui est à la fois normalisée et monotone.

Il convient de noter que toutes les capacités ne sont pas des mesures, et vice versa. Les principales différences entre les deux sont la contrainte d'additivité pour les mesures et la contrainte de normalisation pour les capacités.

L'intégrale de Choquet généralise l'intégrale de Lebesgue en permettant l'intégration par rapport à des capacités.

Il est à noter que pour toute fonction d'ensemble μ , il existe une fonction complémentaire μ^c définie par $\mu^c(A) = \mu(\Omega) - \mu(A^c)$. Si μ est concave, alors μ^c est convexe et vice versa.

Definition 4.11. Soit μ et ν deux fonctions d'ensemble sur Ω . D'après Schmeidler [Schmeidler, 1986], la fonction μ domine la fonction ν si, pour tout sous-ensemble A de Ω , la condition suivante est satisfaite :

$$\mu(A) \geq \nu(A).$$

Proposition 4.1. Soit μ une fonction d'ensemble concave définie sur Ω , alors il existe au moins une fonction d'ensemble additive P telle que, pour tout sous-ensemble A de Ω , μ domine P et P domine la complémentaire de μ :

$$\mu(A) \geq P(A) \geq \mu^c(A).$$

Cette proposition est prouvée dans [Grabisch, 2016]. De plus dans [Waegenaere, 1997] il est prouvé qu'une fonction d'ensemble concave domine un ensemble convexe de fonction d'ensemble additive.

4.2.2 Définition de l'intégrale de Choquet discrète

L'intégrale de Choquet est un concept mathématique qui occupe une place centrale dans la théorie de l'intégration. Introduite dans la première moitié du XXe siècle par Gustave Choquet, elle s'est révélée être une généralisation de l'intégrale de Lebesgue permettant d'aborder les problèmes où les interactions entre les sources d'information ne sont pas simplement additives.

Bien que ses racines soient ancrées dans le domaine de la physique [Choquet, 1954], l'intégrale de Choquet a trouvé des applications dans de nombreux domaines, notamment la prise de décision multicritère, la théorie des jeux, l'économie et l'ingénierie [Grabisch, 2016]. En offrant un cadre pour intégrer par rapport à des fonctions d'ensemble non additives, elle permet de modéliser une plus large gamme de systèmes que des approches plus traditionnelles.

L'intégrale de Choquet est initialement définie par rapport à une capacité [Denneberg, 1994].

Soit μ une capacité sur Ω . Soit $x = (x_1, \dots, x_n)$ un vecteur de \mathbb{R}^n . La littérature rapporte généralement deux expressions pour calculer l'intégrale de Choquet discrète de x par rapport à la capacité μ [Grabisch, 2000] :

$$\check{C}_\mu(x) = \sum_{k=1}^n x_{[k]} \cdot (\mu(A_{[k]}) - \mu(A_{[k+1]})), \text{ avec } A_{[n+1]} = \emptyset, \quad (2.1)$$

$$\check{C}_\mu(x) = \sum_{k=1}^n (x_{[k]} - x_{[k-1]}) \cdot \mu(A_{[k]}), \text{ avec } x_{[0]} = 0. \quad (2.2)$$

où $[.]$ est une permutation qui trie les éléments de x dans un ordre croissant :

$$x_{[1]} \leq x_{[2]} \leq \dots \leq x_{[n]} \text{ avec } A_{[k]} = \{[k], \dots, [n]\}.$$

L'intégrale de Choquet vis à vis de μ^c peut être facilement calculée en utilisant la relation $\check{C}_{\mu^c}(x) = -\check{C}_\mu(-x)$. Cette relation est prouvée dans [Murofushi, 1994].

Proposition 4.2. *Soit μ et ν deux fonctions d'ensemble sur Ω , soit x un vecteur de \mathbb{R}^n . Si pour tout sous-ensemble $A \subseteq \Omega$, $\mu(A) \geq \nu(A)$ alors $\check{C}_\mu(x) \geq \check{C}_\nu(x)$.*

Cette proposition est prouvée dans [Murofushi, 1994].

Les capacités sont monotones. Cependant, comme le montre le travail de Murofushi et al. [Murofushi, 1994], la monotonie n'est pas essentielle.

Plusieurs travaux, dont [Narukawa, 2006 ; Grabisch, 2000 ; Waegenare, 1997 ; Grabisch, 2015], ont proposé d'étendre l'intégrale de Choquet aux fonctions d'ensemble non-monotones. Il a été démontré dans [Waegenare, 1997], que les expressions (2.1) et (2.2) peuvent être utilisées pour calculer l'intégrale de Choquet par rapport à une fonction d'ensemble non-monotone.

4.3 L'intégrale de Choquet comme modèle d'apprentissage

Dans cette section, nous commençons par le cas particulier de l'intégrale de Choquet par rapport à une fonction d'ensemble additive, puis nous poursuivons vers le cas général.

4.3.1 La somme pondérée

Soit P une fonction d'ensemble additive sur Ω , c'est-à-dire que pour tout sous-ensemble A de Ω , la valeur de P sur A est donnée par :

$$P(A) = \sum_{i \in A} P(\{i\}) \quad (2.3)$$

Par conséquent, la valeur de P est entièrement déterminée par ses valeurs sur les singletons de Ω . Ces n valeurs peuvent être représentées par un vecteur ψ de \mathbb{R}^n tel que $\forall i \in \Omega, \psi_i = P(\{i\})$. Ceci illustre la propriété fondamentale des fonctions d'ensemble additives : bien qu'elles soient définies sur l'ensemble des parties de Ω , elles sont entièrement déterminées par leurs valeurs sur les singletons de Ω . On a donc la relation suivante pour $x \in \mathbb{R}^n$:

$$\check{C}_P(x) = \sum_{i=1}^n \psi_i \cdot x_i \quad (2.4)$$

La somme pondérée est une fonction d'agrégation qui est un modèle paramétrique très utilisé. Cette approche est particulièrement utile lorsque toutes les valeurs n'ont pas la même signification ou le même impact dans le contexte considéré [Theil, 1957].

Son utilisation peut notamment gagner en pertinence en ajoutant des contraintes sur les poids. La positivité et la normalisation des poids empêchent que des poids excessivement élevés biaisent indûment la sortie de la somme pondérée, assurant ainsi la comparabilité des importances relatives que revêt chacune des entrées.

La somme pondérée est continue, homogène de degré 1 et monotone mais elle n'est pas invariante par permutation.

L'utilisation de la somme pondérée en tant que modèle d'apprentissage est une méthode nommée "régression linéaire". C'est l'une des techniques d'apprentissage les plus basiques et largement utilisés en apprentissage supervisé [Hastie, 2001].

4.3.2 Limites de la somme pondérée comme modèle d'apprentissage

La somme pondérée est fréquemment adoptée comme modèle d'apprentissage supervisé pour sa simplicité et son faible coût algorithmique. Ces caractéristiques en font une option de prédiction pour des applications nécessitant une rapidité d'exécution.

Cependant, malgré ses avantages, la somme pondérée révèle des limites intrinsèques pour modéliser des situations où les mesures accordées aux sous-ensembles ne sont pas intégralement déterminées par la mesure de chaque élément individuel.

Prenons l'exemple d'un modèle permettant d'évaluer la qualité d'une connexion se basant sur deux observations issues de deux capteurs : la force du signal cellulaire et la congestion du réseau. Chaque observation, prise individuellement, pourrait suggérer un certain niveau de qualité. Toutefois, leur combinaison - un signal cellulaire faible dans une zone à forte congestion - pourrait représenter une dégradation de la qualité bien plus importante que la somme des dégradations individuelles. La somme pondérée ne serait pas adéquate pour capturer cette amplification de la dégradation.

4.3.3 L'intégrale de Choquet par rapport à une fonction d'ensemble non-additive

L'intégrale de Choquet par rapport à une fonction d'ensemble non-additive, en tant que modèle d'apprentissage, offre des avantages distincts par rapport à la somme pondérée classique.

Tout d'abord, elle permet de modéliser un plus grand nombre de systèmes.

Enfin, et c'est peut-être là où l'intégrale de Choquet se démarque le plus, elle offre des mesures pour quantifier non seulement l'importance des variables prédictives individuelles, mais aussi l'interaction entre les groupes de variables. En d'autres termes, elle prend en compte l'influence relative de chaque variable et la manière dont la combinaison de différentes variables peut avoir un impact plus grand ou moindre que la somme de leurs contributions individuelles. Dans [34], les auteurs ont introduit le concept de "choquistic regression", une généralisation de la régression logistique qui est une méthode de classification basée sur la régression linéaire à laquelle on ajoute une fonction de probabilité. Ils utilisent l'intégrale de Choquet par rapport à une capacité pour remplacer la somme pondérée ce qui leur permet d'obtenir un modèle plus précis i.e. ; les sorties du modèle sont plus proches de celle du système modélisé qu'en utilisant la somme pondérée.

Dans [Havens, 2019], les auteurs ont approfondi l'utilisation de l'intégrale de Choquet en tant que modèle d'apprentissage, proposant un modèle basé sur l'intégrale de Choquet par rapport à une fonction d'ensemble non-monotone.

Dans une perspective différente, l'intégrale de Choquet a également été utilisée dans la modélisation des problèmes de préférence [Tehrani, 2010 ; Herin, 2022]. L'apprentissage des préférences fait référence à une catégorie de techniques d'apprentissage automatique axées sur la

prédiction des préférences d'un utilisateur ou d'un groupe d'utilisateurs. Plutôt que de prédire une valeur ou une catégorie spécifique comme dans les tâches de régression ou de classification, l'apprentissage des préférences vise à prédire un ordre de préférence parmi un ensemble d'objets ou d'options.

4.3.4 Limites de l'intégrale de Choquet comme modèle d'apprentissage

L'intégrale de Choquet, malgré ses avantages, présente certaines limitations intrinsèques lorsqu'elle est utilisée comme modèle d'apprentissage. L'une des principales préoccupations concerne le coût de calcul associé à cette approche.

Complexité exponentielle : représenter une capacité permettant d'agréger n données nécessite 2^n paramètres. Cela signifie que pour un ensemble de taille modérément grande, le nombre de paramètres devient rapidement énorme, rendant la tâche d'apprentissage plus complexe.

K-additivité : pour surmonter la complexité associée à un grand nombre de paramètres, une solution consiste à restreindre la capacité afin qu'elle soit k -additive [64]. Une conséquence de la k -additivité est qu'une capacité k -additive peut être entièrement déterminée par ses valeurs sur les sous-ensembles d'au plus k éléments. Seuls les sous-ensembles jusqu'à k éléments ont une influence non nulle sur la mesure. Ceci a pour effet de réduire considérablement le nombre de paramètres à estimer, le faisant passer de 2^n à $\sum_{i=1}^k \binom{n}{i}$ ¹ ce qui réduit le coût de calcul.

Optimisation sous contrainte de k -additivité : bien que la k -additivité réduise le nombre de paramètres, elle introduit un autre défi : l'apprentissage de ces paramètres nécessite des algorithmes d'optimisation sous contrainte de k -additivité. La mise en place de ces algorithmes, qui doivent garantir que seules les interactions jusqu'à k éléments sont prises en compte, est coûteuse en temps et ressources de calcul.

En résumé, la possibilité de modéliser une variété de systèmes offerte par l'intégrale de Choquet s'accompagne d'un prix : un important coût de calcul. La reconnaissance de ces contraintes est importante lors de l'évaluation de son utilisation en tant que modèle d'apprentissage.

De surcroît, même si l'intégrale de Choquet discrète est une généralisation de la somme pondérée et offre ainsi la possibilité de modéliser une vaste variété de systèmes, elle ne peut, de par sa sortie précise, être un modèle adéquat pour des systèmes incohérents ou des systèmes aux sorties et/ou entrées intervallistes. Dans la section suivante, nous examinons des modèles plus appropriés pour ces types de systèmes.

5 Agrégation avec sortie intervalliste

Une agrégation produisant une sortie intervalliste pourrait être envisagée comme une approche pertinente pour modéliser des systèmes avec des entrées et/ou sorties intervallistes. Elle pourrait également offrir une perspective intéressante pour modéliser des systèmes incohérents en suggérant un intervalle de valeurs englobant les sorties possibles associées à une entrée donnée. Il existe trois approches principales pour obtenir des sorties intervallistes. Une d'entre elles

1. $\binom{n}{i}$ représente le nombre de sous-ensemble à i éléments d'un ensemble à n éléments

consiste à établir un intervalle autour d'une sortie précise, une deuxième à considérer des paramètres qui sont des intervalles et une troisième à étendre une agrégation à des entrées qui sont des intervalles.

5.1 Calcul d'un intervalle autour d'une sortie précise d'agrégation

La quantification de l'incertitude joue un rôle essentiel dans la prise de décisions basées sur les prédictions des modèles en apprentissage automatique. Dans ce contexte, par "incertitude" on entend l'absence de connaissance concernant la correspondance entre la prédiction du modèle et la sortie attendue. Cette section se penche sur les méthodes permettant de calculer un intervalle rendant compte de l'incertitude d'un modèle à sortie précise.

5.1.1 Intervalle de confiance

Un intervalle de confiance est une plage de valeurs à l'intérieur de laquelle on peut raisonnablement supposer que se trouve la vraie valeur d'un paramètre inconnu, avec un certain degré de confiance. Cet intervalle est généralement calculé à partir d'un échantillon de données, et sa largeur dépend de la taille de l'échantillon, de la variabilité des données et du niveau de confiance choisi [Hastie, 2001]. Dans ce contexte, la variabilité des données se réfère à la manière dont les valeurs individuelles des données diffèrent les unes des autres. Une variabilité élevée indique que les valeurs sont très différentes les unes des autres, tandis qu'une faible variabilité indique que les valeurs sont proches les unes des autres.

L'intervalle de confiance est une technique très utilisée en apprentissage automatique avec différents types de modèles [Ranathunga, 2022; Campagner, 2022; Jiang, 2008]. L'intervalle de confiance permet également de comparer des modèles. Dans le processus de sélection de modèles, un intervalle de confiance plus étroit est souvent révélateur d'une plus grande stabilité des performances du modèle [Raschka, 2018]. Cette information est cruciale, car elle permet de déterminer si la différence de performance entre les modèles est statistiquement significative ou non. La comparaison des intervalles de confiance associés aux différents modèles offre ainsi une base solide pour choisir le modèle le plus adéquat [Mendez, 2019].

5.1.2 Intervalle de prédiction

Les intervalles de prédiction en apprentissage automatique offrent un éclairage sur l'incertitude associée aux prédictions d'un modèle [Patel, 1989].

À la différence des intervalles de confiance, qui concernent la fiabilité des paramètres estimés du modèle, les intervalles de prédiction s'intéressent à l'incertitude de la sortie du modèle. Dans le cas d'une régression linéaire, par exemple, un intervalle de prédiction peut indiquer la plage de valeurs au sein de laquelle on s'attend à ce que la valeur réelle d'une nouvelle observation se situe [Kuleshov, 2018]. Cet outil est particulièrement utile dans divers domaines d'application pratique. Dans le secteur financier, par exemple, les intervalles de prédiction sont essentiels pour évaluer les risques d'investissement. Ils permettent aux investisseurs de mieux comprendre la variabilité attendue des rendements d'un actif donné, en fournissant une estimation détaillée

de la gamme de valeurs que cet actif peut potentiellement atteindre [Mykland, 2003]. Ainsi, les intervalles de prédiction aident à prendre des décisions plus éclairées.

5.1.3 Intervalle de crédibilité

Une distribution de probabilité décrit la manière dont la probabilité est répartie sur un ensemble d'issues possibles. Elle fournit une mesure mathématique qui assigne à chaque issue possible une probabilité précise, garantissant que ces probabilités satisfont certaines propriétés fondamentales. Le théorème de Bayes est une pierre angulaire des probabilités et des statistiques. Il décrit la manière dont les probabilités associées à différents événements doivent être à la lumière de nouvelles données mises à jour. En essence, il fournit un cadre pour comprendre comment la probabilité d'un événement évolue lorsqu'une nouvelle observation est réalisée.

La distribution postérieure est la distribution de probabilité des paramètres d'un modèle après avoir considéré les données observées. Cette distribution est le fruit de la combinaison des connaissances initiales des paramètres, représentée par la distribution a priori, et des informations nouvelles fournies par les données [Winkler, 1967].

La distribution a priori est la connaissance initiale concernant les paramètres avant que toutes les données ne soient observées. Elle peut être basée sur des connaissances antérieures, ou simplement être une expression de la subjectivité du modéleur [Winkler, 1967].

Enfin, la vraisemblance est l'outil qui évalue la probabilité d'obtenir les données observées en fonction des différentes valeurs des paramètres. Plus la vraisemblance est élevée pour une valeur donnée des paramètres, plus cette valeur des paramètres est jugée vraisemblable au regard des données.

Le modèle statistique bayésien est une approche en statistiques où l'on utilise le théorème de Bayes pour mettre à jour la probabilité des paramètres à mesure que de nouvelles données sont observées. Cela se fait en combinant la connaissance antérieure sur les paramètres (distribution a priori) avec la probabilité d'observer les données actuelles pour ces paramètres (vraisemblance). Le résultat est une distribution postérieure qui reflète la combinaison de la connaissance antérieure et des nouvelles données [Van de Schoot, 2021].

Un intervalle de crédibilité est utilisé dans les modèles statistiques bayésiens pour donner une estimation de la plage de valeurs plausibles pour un paramètre ou une variable, en tenant compte à la fois des données observées et des connaissances antérieures ou des hypothèses subjectives [Van de Schoot, 2021].

Dans l'apprentissage automatique, l'intervalle de crédibilité est utilisé pour estimer l'incertitude associée à une prédiction ou à une estimation de paramètre. Il s'agit de l'intervalle dans lequel un paramètre non observé (ou une prédiction future) doit se situer, compte tenu de la probabilité postérieure, pour un certain seuil de probabilité a priori.

Contrairement à l'intervalle de confiance, l'intervalle de crédibilité bayésien prend en compte les connaissances a priori sur les paramètres, ce qui le rend particulièrement utile lorsqu'on dispose d'informations précédentes sur le problème [Charles, 2016 ; Wang, 2016 ; Hong, 2022].

5.2 Extension d'une agrégation aux entrées intervallistes

Les entrées intervallistes sont des données qui sont exprimées sous forme d'intervalle plutôt que sous forme de valeurs uniques. Cela peut être induit, dans de nombreuses situations réelles, par la difficulté à obtenir une mesure exacte en raison des incertitudes inhérentes aux instruments de mesure ou aux fluctuations naturelles des quantités mesurées. Par exemple, les instruments de mesure de la température peuvent avoir une certaine marge d'erreur, ou la température peut varier de manière naturelle à différents endroits et à différents moments. Ces intervalles peuvent être utiles pour refléter l'incertitude dans les données : si nous ne connaissons pas la température exacte, mais savons qu'elle est comprise entre 20 et 25 degrés, cela peut être représenté par l'intervalle $[20, 25]$.

L'extension d'une agrégation à ces types d'entrées intervallistes est un processus non trivial. Cela nécessite de définir comment les intervalles doivent être combinés. Il existe plusieurs méthodes pour faire cela, et la méthode appropriée peut dépendre du contexte [Hickey, 2001 ; Garczarczyk, 2000 ; Nikulchev, 2023 ; Pekaslan, 2021]. Toutefois, l'approche la plus employée est l'utilisation de l'arithmétique des intervalles. L'arithmétique des intervalles est une branche de l'arithmétique qui traite des intervalles en lieu et place des valeurs réelles. Les opérations arithmétiques de base (addition, soustraction, multiplication, division) peuvent être effectuées sur des intervalles [Jaulin, 2001].

— **Addition :**

Soit $I = [a, b]$ et $J = [c, d]$, alors $I + J = [a + c, b + d]$.

— **Soustraction :**

$I - J = [a - d, b - c]$.

— **Multiplication :**

La multiplication est plus complexe, car elle dépend des signes de a , b , c et d . En général, il faut considérer toutes les combinaisons possibles de produits ac , ad , bc et bd et prendre le minimum et le maximum des quatre valeurs comme bornes de l'intervalle résultant.

— **Division :**

La division est similaire à la multiplication. Si l'intervalle diviseur contient la valeur 0, les bornes de l'intervalle résultant sont rejetées à l'infini.

Cette extension des opérations arithmétiques élémentaires aux intervalles rend possible l'extension de la plupart des formes d'agrégation à des entrées intervallistes, car pour la plupart, leur calcul repose sur ces opérations basiques [Ahn, 2007].

5.3 Modèles intrinsèquement intervallistes

Les modèles intrinsèquement intervallistes représentent une catégorie de modèles qui diffèrent fondamentalement de ceux traditionnellement utilisés en modélisation. Ces modèles sont caractérisés par leur aptitude à produire une sortie sous forme d'intervalle dans tous les cas, même lorsque les entrées sont des valeurs précises. Cette particularité distingue les modèles intrinsèquement intervallistes des autres types de modèles intervallistes. Dans un modèle traditionnel, la sortie peut être un intervalle uniquement lorsque les entrées le sont, ou bien, même

si la sortie est un nombre précis, on peut calculer un intervalle de confiance, de crédibilité ou de prédiction pour évaluer l'incertitude associée à cette prédiction. En revanche, dans les modèles intrinsèquement intervallistes, la sortie est toujours un intervalle, indépendamment de la précision des entrées. Cela peut offrir une représentation de divers phénomènes, en particulier lorsque les données disponibles sont issues d'un système incohérent.

5.3.1 Paramètre imprécis

Les modèles à paramètres d'intervalle permettent aux paramètres d'être représentés par un intervalle de valeurs possibles plutôt que par une valeur précise [Jin-Tsong, 2003]. Dans [Hladík, 2012] les auteurs introduisent une méthode novatrice de régression linéaire par intervalles. Dans cette approche, les paramètres appris ne sont pas des valeurs ponctuelles, mais des intervalles. Plutôt que d'utiliser simplement des valeurs uniques, l'algorithme d'apprentissage considère à la fois le centre et le rayon de chaque intervalle. Cette technique offre la possibilité de modéliser des systèmes dont les sorties sont des intervalles via les méthodes d'apprentissage.

5.3.2 Agrégation maxitive

L'agrégation maxitive, introduite par Loquin [Loquin, 2008a], se présente comme un modèle potentiel pour les systèmes incohérents. Elle prend un vecteur de valeurs réelles en entrée pour produire un intervalle de valeurs réelles en sortie, avec comme paramètre un vecteur de valeurs précises. Ces paramètres différencient l'agrégation maxitive de celles mentionnées dans la section 5.3.1, où les paramètres n'étaient pas des valeurs précises. Par conséquent, l'agrégation maxitive est un modèle intrinsèquement intervalliste, doté de paramètres de même nature que ceux des agrégations à sortie précise et ainsi conserve leur facilité d'utilisation.

Les paramètres utilisés dans l'approche maxitive répondent à des contraintes spécifiques et sont qualifiés par Loquin de noyaux.

Definition 5.1 (Noyau sommatif). Le noyau sommatif, noté κ , est défini sur l'ensemble Ω et prend ses valeurs dans $[0, 1]$. Le noyau sommatif est représenté soit par une séquence $(\kappa_i)_{i=1, \dots, n}$ d'éléments de $[0, 1]$, soit comme un vecteur dans $[0, 1]^n$. Pour un noyau sommatif, la somme de tous ses éléments est égale à 1.

$$\sum_{i \in \Omega} \kappa_i = 1. \tag{2.5}$$

L'utilisation du noyau sommatif en tant que paramètre d'une somme pondérée, assure que l'importance relative de chaque entrée est proportionnelle à la valeur du poids qui lui est associé. Cette méthodologie permet une analyse et une interprétation claire des contributions relatives de chaque entrée à l'agrégation totale.

Definition 5.2 (Mesure additive associée à un noyau sommatif). Soit un noyau sommatif κ . La

mesure additive P_κ associée à κ est définie pour tout ensemble $A \subseteq \Omega$ par :

$$P_\kappa(A) = \sum_{i \in A} \kappa_i.$$

Une mesure additive associée à un noyau sommatif est une mesure de probabilité.

Definition 5.3 (Noyau maxitif). Le noyau maxitif, noté π , est défini sur Ω et prend ses valeurs dans $[0, 1]$. Il vérifie la propriété de maxitivité, c'est-à-dire que pour tout sous-ensemble de Ω , le supremum des valeurs du noyau sur ce sous-ensemble est toujours égal à 1.

$$\sup_{i \in \Omega} \pi_i = 1, \quad (2.6)$$

Definition 5.4 (Mesure de possibilité associée à un noyau maxitif). Soit un noyau maxitif π , la mesure de possibilité, notée Π_π , est définie pour tout ensemble $A \subseteq \Omega$ par :

$$\Pi_\pi(A) = \sup_{i \in A} \pi_i.$$

Definition 5.5 (Mesure de nécessité associée à un noyau maxitif). Soit une mesure de possibilité Π_π associée à un noyau maxitif π . La mesure de nécessité est la complémentaire de la mesure de possibilité définie pour tout ensemble $A \subseteq \Omega$ par :

$$\Pi_\pi^c(A) = \Pi_\pi(\Omega) - \Pi_\pi(A^c),$$

où A^c est le complémentaire de A dans Ω .

Definition 5.6 (Agrégation maxitive). L'agrégation maxitive pour un vecteur d'entrée $x \in \mathbb{R}^n$ est donnée par :

$$\mathcal{A}_{\Pi_\pi}(x) = [\mathbb{C}_{\Pi_\pi^c}(x), \mathbb{C}_{\Pi_\pi}(x)] \quad (2.7)$$

La mesure de possibilité étant concave sa complémentaire, qui est la mesure de nécessité, est convexe.

Étant concave, une mesure de possibilité permet d'encoder un ensemble de mesures additives.

Definition 5.7 (Domination d'un noyau sommatif par un noyau maxitif). Soit $\pi \in [0, 1]^n$ un noyau maxitif et $\kappa \in [0, 1]^n$ un noyau sommatif, on dit que π dommine κ si la mesure de possibilité associé à π domine la mesure additive associé à κ .

Definition 5.8 (Cœur maxitif). Soit $\pi \in [0, 1]^n$ un noyau maxitif. Le cœur maxitif de π , noté $\mathcal{M}(\pi)$, est défini par :

$$\mathcal{M}(\pi) = \{\kappa \in [0, 1]^n \mid \forall A \subseteq \Omega, \Pi_\pi^c(A) \leq P_\kappa(A) \leq \Pi_\pi(A)\}. \quad (2.8)$$

Le cœur maxitif d'un noyau maxitif désigne l'ensemble des noyaux sommatifs qu'il domine. Il est important de noter que le cœur maxitif d'un noyau maxitif est toujours non vide [Loquin,

2008a]. Par ailleurs, étant donné qu'une intégrale de Choquet par rapport à une mesure additive s'interprète comme une somme pondérée, l'agrégation maxitive peut être envisagée comme un ensemble convexe de sommes pondérées associées à des noyaux sommatifs.

De ce fait, avec cette agrégation il est possible de modéliser des systèmes incohérents tout en conservant une complexité linéaire, ce qui est un avantage significatif. Toutefois, un obstacle majeur demeure, la contrainte sur la positivité des poids et leurs normalisation. Cette contrainte limite sa capacité à modéliser des systèmes dont les modèles linéaires font intervenir des paramètres autres que des noyaux sommatifs.

6 Conclusion

À la croisée des enjeux de la modélisation des systèmes, la question du choix des modèles gagne en intérêt. Les fonctions d'agrégation paramétriques, sont des modèles bien adaptés pour la modélisation des systèmes multi-entrées mono-sortie. L'apprentissage supervisé émerge comme une technique d'ajustement des paramètres du modèle dans le cas de la modélisation des systèmes étiquetés. Ces systèmes étiquetés peuvent avoir des sorties précises comme imprécises. De plus, ils peuvent être cohérents ou incohérents. Face à cette diversité de système l'attrait des modèles paramétriques intervallistes est manifeste.

Dans la lignée des travaux de Loquin et al., l'objectif de cette thèse est de généraliser l'agrégation maxitive à des paramètres quelconques en évitant la complexité des modèles d'agrégation intervallistes traditionnels. Ainsi, l'ambition est de mettre au point une méthode d'agrégation intervalliste qui présente les avantages de l'agrégation maxitive sans la contrainte de la normalisation et de la positivité des paramètres.

Cette recherche aspire à offrir un modèle capable de rendre compte de l'incohérence des systèmes, tout en restant simple et pratique pour l'apprentissage automatique.

Chapitre 3

Opérateur macsum et agrégation macsum

Objectifs

L'objectif de ce chapitre est d'introduire un nouvel outil en matière d'agrégation de données : une nouvelle fonction d'agrégation intervalliste. Cette fonction nommée agrégation macsum est construite à partir d'un vecteur d'entrées et d'un vecteur de paramètres de dimension identique. Sa particularité réside dans le fait que sa sortie n'est pas une valeur unique, mais un intervalle. Le chapitre se déroule en plusieurs sections. Il débute par la présentation d'une nouvelle fonction d'ensemble concave, établissant le fondement de cette nouvelle agrégation intervalliste. On démontre ensuite que l'agrégation macsum permet de représenter un ensemble convexe d'agrégations linéaires. Enfin, une méthode simple pour la calculer est exposée. Le but de ce chapitre est de proposer un modèle qui combine la simplicité d'utilisation des modèles linéaires classiques avec la représentativité des modèles paramétriques intervallistes.

Sommaire

1	Introduction	26
2	Généralisation de la domination maxitive	26
2.1	L'opérateur macsum	27
2.2	Concavité de l'opérateur macsum	29
2.3	Domination de l'opérateur macsum sur les opérateurs additifs	30
3	Représentation d'un ensemble convexe d'agrégations linéaires	32
4	Calcul de l'agrégation macsum	34
4.1	Formules	34
4.2	Algorithmes	36
5	Conclusion	37

1 Introduction

Dans ce chapitre, nous introduisons une nouvelle fonction d'agrégation intervalliste [Strauss, 2022a]. La sortie de cette fonction se calcule à partir d'un vecteur d'entrées et d'un vecteur de paramètres de même dimension, rappelant ainsi la structure des méthodes d'agrégation paramétriques traditionnelles comme la somme pondérée. Cependant, une différence significative réside dans sa sortie qui consiste en un intervalle englobant un ensemble convexe de résultats obtenus par des sommes pondérées. L'atout majeur de cette approche est d'allier à la fois la fiabilité d'un modèle paramétrique intervalliste et la simplicité d'utilisation de la somme pondérée. Plus concrètement, imaginons que l'on essaye d'estimer la température d'une journée printanière. Si on donne un intervalle, disons entre 20°C et 25°C, il y a de fortes chances que la température réelle se trouve à l'intérieur de cet intervalle, comparée à une prédiction précise, par exemple 22°C. En donnant un intervalle plutôt qu'une valeur précise, l'estimation couvre une plage de valeurs, augmentant ainsi les chances d'inclure la température réelle et donc la fiabilité du modèle.

Néanmoins, cette méthode basée sur des intervalles sacrifie la précision en fournissant une plage de valeurs possibles plutôt qu'une estimation précise. Un bon modèle intervalliste doit offrir un équilibre entre assurer une couverture plus large (fiabilité) et conserver une certaine simplicité d'utilisation, tout en reconnaissant l'inévitable compromis avec la précision. De surcroît, il existe des cas de systèmes pour lesquels un modèle à sortie précise est incapable d'être un représentant adéquat. C'est le cas des systèmes à sorties imprécises et des systèmes incohérents, dont la nature intrinsèquement imprécise nécessite un modèle à sortie imprécise. Dans les systèmes incohérents, une entrée spécifique peut conduire à différentes sorties. Par conséquent, si l'on veut modéliser un système incohérent par un modèle ayant une sortie précise, ce dernier doit nécessairement associer à une même entrée un ensemble de valeurs différentes. Dans ce cas un modèle à sortie intervalliste est plus approprié. Ce chapitre est organisé comme suit : nous commençons par définir une nouvelle fonction d'ensemble concave en tant que généralisation signée de la mesure de possibilité. Ensuite, nous montrons comment cette fonction d'ensemble peut être utilisée pour représenter un ensemble convexe de fonctions d'ensemble additive, conduisant à une agrégation intervalliste. Enfin un algorithme simple pour calculer cette agrégation intervalliste est présenté.

Notre objectif est d'ouvrir la voie à des modèles intervalliste ayant une simplicité d'utilisation comparable à celle des modèles linéaires classiques.

2 Généralisation de la domination maxitive

L'agrégation maxitive proposée par Loquin [Loquin, 2008a] permet de produire un intervalle représentant un ensemble convexe de sommes pondérées avec un seul paramètre. Cependant, ce paramètre est un noyau maxitif, c'est-à-dire que ses composantes doivent appartenir à $[0, 1]$. Une conséquence de cela est que les poids des sommes pondérées dominées par l'agrégation maxitive sont des noyaux sommatifs. Cette technique ne permet donc pas de modéliser un système dont les sorties ne correspondent pas à des sommes pondérées des entrées par un noyau

sommatif. Par ailleurs, son utilisation en tant que modèle d'apprentissage supervisé nécessite l'ajout de contrainte lors de la phase d'ajustement des paramètres. En effet supposons que la relation entrée/sortie puisse correspondre au modèle maxitif. En apprentissage supervisé, les paramètres sont automatiquement modifiés via un processus itératif afin de rapprocher les sorties du modèle de celles du système modélisé. Au cours de ce processus, les modifications apportées aux paramètres peuvent les amener à prendre n'importe quelle valeur réelle et donc rien ne garantit qu'après ces modifications, le vecteur de paramètres soit un noyau maxitif. Pour le garantir, l'optimisation d'une agrégation maxitive nécessite d'imposer une contrainte lors de l'ajustement des paramètres, ce qui engendre une hausse du coût de calcul. Par conséquent, l'agrégation maxitive est non seulement limitée dans sa capacité à représenter les systèmes réels, mais elle alourdit également le coût de calcul du processus d'apprentissage. Notre objectif est de lever les restrictions sur les paramètres de l'agrégation maxitive afin de développer un modèle d'apprentissage intervalliste représentant un ensemble convexe de sommes pondérées de poids quelconque, avec un seul vecteur de paramètres. Dans cette section, nous étendons les travaux de Loquin et al. [Loquin, 2008a] à des vecteurs de paramètres pouvant prendre des valeurs réelles quelconques. Si la fonction d'ensemble maxitive utilisée par Loquin est associée à un noyau maxitif, nous souhaitons ici définir des fonctions d'ensemble associées à des paramètres de \mathbb{R}^n . On rappelle que $\Omega = \{1, \dots, n\} \subseteq \mathbb{N}$ est notre ensemble de référence.

Definition 2.1. Soit $\varphi \in \mathbb{R}^n$ un vecteur de paramètres, on appelle **opérateur** une fonction d'ensemble concave $\mu_\varphi : 2^\Omega \rightarrow \mathbb{R}$ tel que $\mu_\varphi(\emptyset) = 0$ et qui à tout sous-ensemble $A \subseteq \Omega$ associe une valeur réelle obtenue par un calcul systématique impliquant uniquement les valeurs d'un paramètre $\varphi \in \mathbb{R}^n$.

Definition 2.2. Nous appelons **gain d'un opérateur**, la valeur qu'il prend sur son ensemble de référence. Ainsi le gain de l'opérateur μ_φ est $\mu_\varphi(\Omega)$.

Nous pouvons associer à μ_φ un opérateur complémentaire noté μ_φ^c tel que $\forall A \subseteq \Omega$, $\mu_\varphi^c(A) = \mu_\varphi(\Omega) - \mu_\varphi(A^c)$, où A^c représente l'ensemble complémentaire de A dans Ω [Grabisch, 2000].

La fonction d'ensemble additive introduite dans le chapitre 2 est un exemple pertinent d'un opérateur : une fonction d'ensemble P_φ définie par $\forall A \subseteq \Omega$, $P_\varphi(A) = \sum_{k \in A} \varphi_k$ est définie pour tout paramètre $\varphi \in \mathbb{R}^n$.

Remarque. Les fonctions d'ensembles additives sont par définition à la fois concave et convexe. [Borwein, 2006]

2.1 L'opérateur macsum

Nous proposons ici d'étendre les travaux de Loquin et al. [Strauss, 2019] par un opérateur dont l'objet est de représenter un ensemble convexe de fonctions d'ensemble additives. L'extension que nous proposons sous le nom d'*opérateur macsum* est un opérateur noté ν_φ . Il est défini par :

$$\forall A \subseteq \Omega, \nu_\varphi(A) = \max_{i \in A} \varphi_i^+ + \min_{i \in \Omega} \varphi_i^- - \min_{i \in A^c} \varphi_i^-, \quad (3.1)$$

où A^c est l'ensemble complémentaire de A dans Ω , $\varphi^+ = \max(0, \varphi)$ et $\varphi^- = \min(0, \varphi)$.

En conséquence : $\nu_\varphi(\Omega) = \max_{i \in \Omega} \varphi_i^+ + \min_{i \in \Omega} \varphi_i^- = \bar{\alpha} + \underline{\alpha}$, où $\bar{\alpha} = \max_{i \in \Omega} \varphi_i^+$ et $\underline{\alpha} = \min_{i \in \Omega} \varphi_i^-$.

La fonction d'ensemble complémentaire de ν_φ , notée ν_φ^c , est donnée par :

$$\begin{aligned} \forall A \subseteq \Omega, \quad \nu_\varphi^c(A) &= \nu_\varphi(\Omega) - \nu_\varphi(A^c), \\ &= \bar{\alpha} + \underline{\alpha} - \left(\max_{i \in A^c} \varphi_i^+ - \min_{i \in A} \varphi_i^- + \underline{\alpha} \right), \\ &= \min_{i \in A} \varphi_i^- + \bar{\alpha} - \max_{i \in A^c} \varphi_i^+ = \min_{i \in A} \varphi_i^- + \min_{i \in A^c} (\bar{\alpha} - \varphi_i^+). \end{aligned} \tag{3.2}$$

Ainsi $\nu_\varphi(\Omega) = \nu_\varphi^c(\Omega) = \underline{\alpha} + \bar{\alpha}$. De plus, $\nu_\varphi(\emptyset) = \max_{i \in \emptyset} \varphi_i^+ - \min_{i \in \Omega} \varphi_i^- + \min_{i \in \Omega} \varphi_i = 0$ et $\nu_\varphi^c(\emptyset) = \bar{\alpha} - \max_{i \in \Omega} \varphi_i^+ + \min_{i \in \emptyset} \varphi_i^- = 0$.

Remarque. Le terme *macsum* est utilisé car ν_φ peut être exprimé comme la somme de deux opérateurs maxitifs : $\nu_\varphi(A) = \max_{i \in A} \varphi_i^+ + \underline{\alpha} - \min_{i \in A^c} \varphi_i^- = \max_{i \in A} \varphi_i^+ + \max_{i \in A^c} (\underline{\alpha} - \varphi_i^-)$.

Exemple de calcul : Supposons que $\Omega = \{1, 2, 3, 4\}$ et que $\varphi = (-2, 3, -1, 4)$.

1. Calcul des valeurs de φ_i^+ et φ_i^- :

$$\begin{aligned} \varphi_1^+ &= \max(0, -2) = 0 \\ \varphi_2^+ &= \max(0, 3) = 3 \\ \varphi_3^+ &= \max(0, -1) = 0 \\ \varphi_4^+ &= \max(0, 4) = 4 \\ \varphi_1^- &= \min(0, -2) = -2 \\ \varphi_2^- &= \min(0, 3) = 0 \\ \varphi_3^- &= \min(0, -1) = -1 \\ \varphi_4^- &= \min(0, 4) = 0 \end{aligned}$$

2. Calcul des valeurs requises :

$$\begin{aligned} \max_{i \in A} \varphi_i^+ &= \max(\varphi_1^+, \varphi_2^+, \varphi_3^+, \varphi_4^+) \\ &= \max(0, 3, 0, 4) \\ &= 4 \\ \min_{i \in \Omega} \varphi_i^- &= \min(\varphi_1^-, \varphi_2^-, \varphi_3^-, \varphi_4^-) \\ &= \min(-2, 0, -1, 0) \\ &= -2 \\ \min_{i \in A^c} \varphi_i^- &= \min(\varphi_1^-, \varphi_2^-, \varphi_3^-, \varphi_4^-) \\ &= \min(-2, 0, -1, 0) \\ &= -2 \end{aligned}$$

3. Calcul de $\nu_\varphi(A)$:

$$\begin{aligned}\nu_\varphi(A) &= \max_{i \in A} \varphi_i^+ + \min_{i \in \Omega} \varphi_i^- - \min_{i \in A^c} \varphi_i^- \\ &= 4 + (-2) - (-2) \\ &= 4 + (-2) + 2 \\ &= 4\end{aligned}$$

Ainsi, dans cet exemple, $\nu_\varphi(A) = 4$ lorsque A est un sous-ensemble de Ω défini comme $\{2, 4\}$.

2.2 Concavité de l'opérateur macsum

Dans cette section, nous démontrons que l'opérateur macsum est effectivement concave et domine ainsi un ensemble convexe d'opérateurs additifs [Schmeidler, 1986].

Proposition 2.1. $\forall A, B \subseteq \Omega$, $\max_{i \in A \cup B} \varphi_i + \max_{i \in A \cap B} \varphi_i \leq \max_{i \in A} \varphi_i + \max_{i \in B} \varphi_i$.

Preuve. Soit $\tau = \max_{i \in A \cup B} \varphi_i + \max_{i \in A \cap B} \varphi_i - \max_{i \in A} \varphi_i - \max_{i \in B} \varphi_i$. Montrons que $\tau \leq 0$. Posons $a = \max_{i \in A} \varphi_i$, $b = \max_{i \in B} \varphi_i$ et $c = \max_{i \in A \cap B} \varphi_i$. Par construction, on a $\max_{i \in A \cup B} \varphi_i = \max(a, b)$ et $c \leq a, b$. Sans perte de généralité, supposons que $a \leq b$, donc $\max_{i \in A \cup B} \varphi_i = b$. Ainsi, $c \leq a \leq b$, ce qui implique $\tau = b + c - a - b = c - a \leq 0$. \square

Proposition 2.2. $\forall A, B \subseteq \Omega$, $\min_{i \in A \cup B} \varphi_i + \min_{i \in A \cap B} \varphi_i \geq \min_{i \in A} \varphi_i + \min_{i \in B} \varphi_i$.

Preuve. Soit $\tau = \min_{i \in A \cup B} \varphi_i + \min_{i \in A \cap B} \varphi_i - \min_{i \in A} \varphi_i - \min_{i \in B} \varphi_i$. Montrons que $\tau \geq 0$. Posons $a = \min_{i \in A} \varphi_i$, $b = \min_{i \in B} \varphi_i$ et $c = \min_{i \in A \cap B} \varphi_i$. Par construction, on a $\min_{i \in A \cup B} \varphi_i = \min(a, b)$ et $a, b \leq c$. Sans perte de généralité, supposons que $a \leq b$, donc $\min_{i \in A \cup B} \varphi_i = a$. Ainsi, $a \leq b \leq c$, alors $\tau = a + c - a - b = c - b \geq 0$. \square

En utilisant ces deux propositions, nous pouvons désormais prouver la concavité de l'opérateur macsum.

Proposition 2.3. ν_φ est concave et donc ν_φ^c est convexe.

Preuve. La démonstration est triviale en considérant les propositions 2.1 et 2.2. Soient $A, B \subseteq \Omega$. Soit $\tau = \nu_\varphi(A \cup B) + \nu_\varphi(A \cap B) - \nu_\varphi(A) + \nu_\varphi(B)$.

$$\tau = \tau_1 - \tau_2, \text{ avec } \tau_1 = \max_{i \in A \cup B} \varphi_i^+ + \max_{i \in A \cap B} \varphi_i^+ - \max_{i \in A} \varphi_i^+ - \max_{i \in B} \varphi_i^+,$$

$$\text{et } \tau_2 = \min_{i \in (A \cup B)^c} \varphi_i^- + \min_{i \in (A \cap B)^c} \varphi_i^- - \min_{i \in A^c} \varphi_i^- - \min_{i \in B^c} \varphi_i^-.$$

$$\tau_2 = \min_{i \in A^c \cap B^c} \varphi_i^- + \min_{i \in A^c \cup B^c} \varphi_i^- - \min_{i \in A^c} \varphi_i^- - \min_{i \in B^c} \varphi_i^-.$$

En utilisant la proposition 2.1, on a $\tau_1 \leq 0$ et en utilisant la proposition 2.2, on a $\tau_2 \geq 0$. Ainsi, $\tau = \tau_1 - \tau_2 \leq 0$ et donc ν_φ est concave. La démonstration que ν_φ^c est convexe peut être faite de la même manière. \square

Ainsi, l'opérateur macsum est une fonction d'ensemble concave et donc elle domine un ensemble convexe d'opérateur additif.

2.3 Domination de l'opérateur macsum sur les opérateurs additifs

Dans cette section, nous abordons un concept clé : la domination de l'opérateur macsum sur les opérateurs additifs. Pour ce faire, nous commençons par introduire la notion de cœur macsumitif, notée $\mathcal{M}(\varphi)$, associée à un paramètre $\varphi \in \mathbb{R}^n$. Cette notion généralise celle de cœur maxitif introduite par Loquin et al. [Loquin, 2008a].

Definition 2.3. Le cœur macsumitif $\mathcal{M}(\varphi)$ d'un paramètre $\varphi \in \mathbb{R}^n$ est l'ensemble :

$$\mathcal{M}(\varphi) = \left\{ \psi \in \mathbb{R}^n / \forall A \subseteq \Omega, \nu_\varphi^c(A) \leq P_\psi(A) \leq \nu_\varphi(A) \right\}. \quad (3.3)$$

Cette définition coïncide avec celle de [Loquin, 2008b] lorsque $\varphi \in [0, 1]^n$.

Proposition 2.4. Soit $\varphi \in \mathbb{R}^n$, $\forall \psi \in \mathbb{R}^n$ tel que $\psi \in \mathcal{M}(\varphi)$, le gain de l'opérateur additif P_ψ est égal à celui de l'opérateur ν_φ .

Preuve. Soit $\varphi \in \mathbb{R}^n$, $\forall \psi \in \mathbb{R}^n$ tel que $\psi \in \mathcal{M}(\varphi)$ on a que $\forall A \subseteq \Omega, \nu_\varphi^c(A) \leq P_\psi(A) \leq \nu_\varphi(A)$. Par conséquent, $\nu_\varphi^c(\Omega) \leq P_\psi(\Omega) \leq \nu_\varphi(\Omega)$. Comme $\nu_\varphi^c(\Omega) = \nu_\varphi(\Omega) - \nu_\varphi(\emptyset) = \nu_\varphi(\Omega)$, nécessairement $P_\psi(\Omega) = \nu_\varphi(\Omega)$. \square

Definition 2.4. Si $\psi, \varphi \in \mathbb{R}^n$ et que $\psi \in \mathcal{M}(\varphi)$, nous disons que φ **domine** ψ si l'opérateur macsum basé sur φ domine l'opérateur additif basé sur ψ .

Nous rappelons ici brièvement quelques lemmes utiles [Loquin, 2008b] :

Lemme 2.5. (*domination faible*) Soit $\kappa \in [0, 1]^n$ un noyau sommatif de Ω , alors le noyau maxitif $\check{\kappa} \in [0, 1]^n$, défini par $\forall i \in \Omega, \check{\kappa}_i = \sum_{j \in \Omega} \min(\kappa_i, \kappa_j)$, domine κ , c'est-à-dire $\kappa \in \mathcal{M}(\check{\kappa})$.

Lemme 2.6. (*domination forte*) Soit $\kappa \in [0, 1]^n$ un noyau sommatif de Ω , alors le noyau maxitif $\hat{\kappa} \in [0, 1]^n$, défini par $\forall i \in \Omega, \hat{\kappa}_i = \sum_{j \in A_i} \kappa_j$, où $A_i = \{j \in \Omega / \kappa_j \leq \kappa_i\}$, domine κ , c'est-à-dire $\kappa \in \mathcal{M}(\hat{\kappa})$.

Remarque. $\check{\kappa}$ domine faiblement κ par rapport à $\hat{\kappa}$ car $\mathcal{M}(\hat{\kappa}) \subseteq \mathcal{M}(\check{\kappa})$ [Loquin, 2008a].

Ce que nous proposons ici est d'utiliser ces résultats pour également généraliser les notions de domination faible et forte.

Proposition 2.7. (*domination faible généralisée*) Soit $\psi \in \mathbb{R}^n$ un paramètre, alors le paramètre $\check{\psi} \in \mathbb{R}^n$ défini par $\forall i \in \Omega, \check{\psi}_i = \sum_{j \in \Omega} \min(\psi_i^+, \psi_j^+) + \sum_{j \in \Omega} \max(\psi_i^-, \psi_j^-)$, où $\psi^+ = \max(0, \psi)$ et $\psi^- = \min(0, \psi)$, domine ψ , c'est-à-dire $\psi \in \mathcal{M}(\check{\psi})$.

Preuve. Définissons $\alpha^+ = \sum_{i \in \Omega} \psi_i^+$ et $\alpha^- = \sum_{i \in \Omega} \psi_i^-$, puis les noyaux κ^+ et κ^- définis par : $\forall i \in \Omega, \kappa_i^+ = \psi_i^+ / \alpha^+$ et $\kappa_i^- = -\psi_i^- / \alpha^-$ sont sommatifs par construction (c'est-à-dire positifs et normalisés). Ainsi, selon le Lemme 2.5, nous pouvons définir deux noyaux maxitifs $\check{\kappa}^+$ et $\check{\kappa}^-$ par :

$$\forall i \in \Omega, \check{\kappa}_i^+ = \sum_{j \in \Omega} \min(\kappa_i^+, \kappa_j^+) \text{ et } \check{\kappa}_i^- = \sum_{j \in \Omega} \min(-\kappa_i^-, -\kappa_j^-).$$

Définissons deux noyaux $\check{\psi}^+$ et $\check{\psi}^-$, tel que :

$$\forall i \in \Omega, \check{\varphi}_i^+ = \alpha^+ \cdot \check{\pi}_i^+ = \sum_{j \in \Omega} \min(\psi_i^+, \psi_j^+) \text{ et } \check{\varphi}_i^- = \alpha^- \cdot \check{\pi}_i^- = - \sum_{j \in \Omega} \min(-\psi_i^-, -\psi_j^-).$$

Par construction, $\forall i \in \Omega, \check{\varphi}_i^+ \geq 0$ et $\check{\varphi}_i^- \leq 0$. Définissons maintenant $\check{\varphi} = \check{\varphi}^+ - \check{\varphi}^-$.

Selon le Lemme 2.5, nous avons :

$$\forall A \subseteq \Omega, \Pi_{\check{\pi}^+}(A) = \max_{i \in A} \check{\pi}_i^+ \geq P_{\kappa^+}(A) = \sum_{i \in A} \kappa_i^+$$

et

$$\forall A \subseteq \Omega, \Pi_{\check{\pi}^-}(A) = \max_{i \in A} \check{\pi}_i^- \geq P_{-\kappa^-}(A) = - \sum_{i \in A} \kappa_i^-$$

Considérons les deux fonctions d'ensemble $\mu_{\check{\varphi}^+}$ et $\mu_{\check{\varphi}^-}$, définies par :

$$\forall A \subseteq \Omega, \mu_{\check{\varphi}^+}(A) = \max_{i \in A} \check{\varphi}_i^+ \text{ et } \mu_{\check{\varphi}^-}(A) = \min_{i \in A} \check{\varphi}_i^-.$$

nous avons,

$$\forall A \subseteq \Omega, \mu_{\check{\varphi}^+}(A) = \max_{i \in A} \check{\varphi}_i^+ = \alpha^+ \max_{i \in A} \check{\pi}_i^+ \geq \alpha^+ \cdot P_{\kappa^+}(A) = \sum_{i \in A} \alpha^+ \cdot \kappa_i^+ = P_{\psi^+}(A)$$

et

$$\forall A \subseteq \Omega, \mu_{\check{\varphi}^-}(A) = \min_{i \in A} \check{\varphi}_i^- = \alpha^- \max_{i \in A} \check{\pi}_i^- \leq \alpha^- \cdot P_{-\kappa^-}(A) = \sum_{i \in A} \alpha^- \cdot \kappa_i^- = P_{\psi^-}(A)$$

Remarquons que $\min_{i \in \Omega} \check{\varphi}_i^- = \sum_{i \in \Omega} \psi_i^-$. De plus, nous avons également :

$$\forall A \subseteq \Omega, \mu_{\check{\varphi}^-}^c(A) = \min_{i \in \Omega} \check{\varphi}_i^- - \min_{i \in A^c} \check{\varphi}_i^- \geq \sum_{i \in \Omega} \psi_i^- - \sum_{i \in A^c} \psi_i^- = \sum_{i \in A} \psi_i^- = P_{\psi_i^-}(A)$$

Ainsi, $\forall A \subseteq \Omega, \nu_{\check{\varphi}}(A) = \max_{i \in A} \check{\varphi}_i^+ + \min_{i \in \Omega} \check{\varphi}_i^- - \min_{i \in A^c} \check{\varphi}_i^- = \mu_{\check{\varphi}^+}(A) + \mu_{\check{\varphi}^-}^c(A)$, et donc $\nu_{\check{\varphi}}(A) \geq P_{\psi^+}(A) + P_{\psi^-}(A) = P_{\psi}(A)$. \square

Nous avons ainsi démontré la domination faible généralisée; explorons maintenant le cas de la domination forte généralisée.

Proposition 2.8. (*domination forte généralisée*) Soit $\psi \in \mathbb{R}^n$ un paramètre, alors le paramètre $\check{\varphi} \in \mathbb{R}^n$ défini par $\forall i \in \Omega, \check{\varphi}_i = \sum_{j \in A_i^+} \psi_j^+ + \sum_{j \in A_i^-} \psi_j^-$, ou $\psi^+ = \max(0, \psi)$, $\psi^- = \min(0, \psi)$, $A_i^+ = \{j \in \Omega / \psi_j^+ \leq \psi_i^+\}$ et $A_i^- = \{j \in \Omega / \psi_j^- \geq \psi_i^-\}$ domine ψ i.e. $\psi \in \mathcal{M}(\check{\varphi})$.

La preuve de la Proposition 2.8 suit le même schéma que la preuve de la Proposition 2.7.

Preuve. Comme précédemment, nous définissons les noyaux sommatifs κ^+ et κ^- par : $\forall i \in \Omega, \kappa_i^+ = \psi_i^+ / \alpha^+$ et $\kappa_i^- = -\psi_i^- / \alpha^-$, avec $\alpha^+ = \sum_{i \in \Omega} \psi_i^+$ et $\alpha^- = \sum_{i \in \Omega} \psi_i^-$. Comme proposé dans le Lemme 2.6, nous pouvons définir deux noyaux maxitifs $\check{\pi}^+$ et $\check{\pi}^-$ par :

$$\begin{aligned} \forall i \in \Omega, \check{\pi}_i^+ &= \sum_{j \in A_i^+} \kappa_j^+, \check{\pi}_i^- = \sum_{j \in A_i^-} \kappa_j^-, \\ \text{avec } A_i^+ &= \left\{ j \in \Omega / \kappa_j^+ \leq \kappa_i^+ \right\} = \left\{ j \in \Omega / \psi_j^+ \leq \psi_i^+ \right\} \\ \text{et } A_i^- &= \left\{ j \in \Omega / \kappa_j^- \leq \kappa_i^- \right\} = \left\{ j \in \Omega / \psi_j^- \geq \psi_i^- \right\}. \end{aligned}$$

D'après la proposition 2.5 on a :

$$\begin{aligned} \forall A \subseteq \Omega, \Pi_{\hat{\pi}^+}(A) &= \max_{i \in A} \hat{\pi}_i^+ \geq P_{\kappa^+}(A) = \sum_{i \in A} \kappa_i^+ \\ \text{et } \Pi_{\hat{\pi}^-}(A) &= \max_{i \in A} \hat{\pi}_i^- \geq P_{-\kappa^-}(A) = \sum_{i \in A} -\kappa_i^- \\ \text{On a aussi } \Pi_{\hat{\pi}^-}^c(A) &= 1 - \max_{i \in A^c} \hat{\pi}_i^- \leq P_{-\kappa^-}(A). \end{aligned}$$

Soit $\hat{\varphi}^+ = \alpha^+ \cdot \pi^+ = \sum_{j \in A_i^+} \psi_j^+$ et $\hat{\varphi}^- = \alpha^- \cdot \pi^- = \sum_{j \in A_i^-} \psi_j^-$.

On définit $\hat{\varphi} = \hat{\varphi}^+ - \hat{\varphi}^-$. On a :

$$\begin{aligned} \forall A \subseteq \Omega, \nu_{\hat{\varphi}}(A) &= \mu_{\hat{\varphi}^+}(A) + \mu_{\hat{\varphi}^-}^c(A), \\ \text{avec } \mu_{\hat{\varphi}^+}(A) &= \max_{i \in A} \hat{\varphi}_i^+ = \alpha^+ \cdot \Pi_{\hat{\pi}^+}(A) \\ \text{et } \mu_{\hat{\varphi}^-}^c(A) &= \min_{i \in A} \hat{\varphi}_i^- = \alpha^- \cdot \Pi_{\hat{\pi}^-}(A). \end{aligned}$$

Ainsi,

$$\begin{aligned} \forall A \subseteq \Omega, \nu_{\hat{\varphi}}(A) &= \alpha^+ \cdot \Pi_{\hat{\pi}^+}(A) + \alpha^- \cdot \Pi_{\hat{\pi}^-}^c(A) \geq \alpha^+ \cdot P_{\kappa^+}(A) + \alpha^- \cdot P_{\kappa^-}(A) \\ &= P_{\psi^+}(A) + P_{\psi^-}(A) = P_{\psi}(A). \end{aligned}$$

□

Nous avons exploré dans cette section la notion de domination de l'opérateur macsum sur les opérateurs additifs, cela constitue un pas supplémentaire vers notre objectif : bâtir une fonction d'agrégation qui, avec un seul paramètre, représente un ensemble convexe de sommes pondérées par des poids prenant n'importe quelle valeur réelle.

3 Représentation d'un ensemble convexe d'agrégations linéaires

Un intervalle réel est un ensemble de nombres réels compris entre une borne inférieure et une borne supérieure. Notons \mathbb{IR} l'ensemble de tous les intervalles réels.

Maintenant que nous avons démontré que l'opérateur macsum est concave et qu'il domine un ensemble de fonction d'ensemble additive selon le paramètre qui lui est associé, nous proposons d'introduire notre modèle intervalliste. Ce nouveau modèle est une fonction d'agrégation, dont le calcul est basé sur l'intégrale de Choquet discrète, qui va de \mathbb{R}^n dans \mathbb{IR} . Il s'agit d'une généralisation de l'agrégation maxitive introduite par Loquin [Loquin, 2008a].

Definition 3.1. Soit $\varphi \in \mathbb{R}^n$, l'agrégation macsum, noté $\mathcal{A}_{\nu_\varphi}$, est définie pour tout $x \in \mathbb{R}^n$ par :

$$\mathcal{A}_{\nu_\varphi}(x) = \left[\check{\mathcal{C}}_{\nu_\varphi^c}(x), \check{\mathcal{C}}_{\nu_\varphi}(x) \right]. \quad (3.4)$$

Nous rappelons que notre objectif est de maintenir la simplicité d'utilisation des modèles linéaires, en nous limitant à un seul paramètre de même dimension que les entrées, tout en étant en mesure de représenter un ensemble convexe de modèles linéaires. Nous savons déjà que l'agrégation macsum domine un ensemble d'agrégations linéaires de par la concavité de l'opérateur macsum [Denneberg, 1994]. Dans cette section nous montrons que cet ensemble est convexe.

Remarque. Un opérateur additif P_ψ , étant à la fois concave et convexe, est égal à sa complémentaire P_ψ^c . Par analogie avec l'agrégation macsum nous notons désormais l'agrégation linéaire \mathcal{A}_{P_ψ} .

En effet, $\forall x \in \mathbb{R}^n$, $\mathcal{A}_{P_\psi}(x) = [\check{\mathcal{C}}_{P_\psi^c}(x), \check{\mathcal{C}}_{P_\psi}(x)] = [\check{\mathcal{C}}_{P_\psi}(x), \check{\mathcal{C}}_{P_\psi}(x)] = \check{\mathcal{C}}_{P_\psi}(x) = \sum_{i \in \Omega} \varphi_i \cdot x_i$.

En s'appuyant sur le théorème 3.1 [Schmeidler, 1986], qui assure la domination d'un ensemble d'intégrale de Choquet par rapport à des fonctions d'ensembles additives par une intégrale de Choquet par rapport à une fonction d'ensemble concave, nous démontrons que la sortie sous forme d'intervalle issue d'une agrégation macsum représente un ensemble convexe de sorties d'agrégations linéaires.

Theorem 3.1 (Théorème de Schmeidler-Denneberg [Schmeidler, 1986] Proposition 3). *Soit μ une fonction d'ensemble concave et $\mathcal{M}(\mu)$ l'ensemble des fonctions d'ensemble additives dominées par μ . Nous avons que $\forall x \in \mathbb{R}^n$:*

$$\check{\mathcal{C}}_\mu(x) = \sup_{P \in \mathcal{M}(\mu)} \check{\mathcal{C}}_P(x)$$

$$\check{\mathcal{C}}_{\mu^c}(x) = \inf_{P \in \mathcal{M}(\mu)} \check{\mathcal{C}}_P(x).$$

Proposition 3.2. *Soit $\varphi \in \mathbb{R}^n$, $\forall \psi \in \mathcal{M}(\varphi)$, $\forall x \in \mathbb{R}^n$, $\mathcal{A}_{P_\psi}(x) \in \mathcal{A}_{\nu_\varphi}(x)$. De plus, $\forall y \in \mathcal{A}_{\nu_\varphi}(x)$, $\exists \psi$ tel que $y = \mathcal{A}_{P_\psi}(x)$.*

Preuve. Le fait que pour tout $\psi \in \mathcal{M}(\varphi)$, pour tout $x \in \mathbb{R}^n$, $\mathcal{A}_{P_\psi}(x) \in \mathcal{A}_{\nu_\varphi}(x)$ est simplement une reformulation du théorème de Schmeidler-Denneberg [Schmeidler, 1986] dans ce cas particulier. En effet, ce théorème a été démontré pour toute fonction d'ensemble concave. Nous l'utilisons ici uniquement pour les opérateurs. Montrons maintenant que pour tout $y \in \mathcal{A}_{\nu_\varphi}(x)$, il existe un $\psi \in \mathbb{R}^n$ tel que $y = \mathcal{A}_{P_\psi}(x)$. On a :

$\check{\mathcal{C}}_{\nu_\varphi}(x) = \sum_{k=1}^n x_{[k]} \cdot (\nu_\varphi(A_{[k]}) - \nu_\varphi(A_{[k+1]}))$ et $\check{\mathcal{C}}_{\nu_\varphi^c}(x) = \sum_{k=1}^n x_{[k]} \cdot (\nu_\varphi^c(A_{[k]}) - \nu_\varphi^c(A_{[k+1]}))$, où $[\cdot]$ est une permutation qui trie les éléments de x dans un ordre croissant :

$$x_{[1]} \leq x_{[2]} \leq \dots \leq x_{[n]} \text{ et } A_{[k]} = \{[k], \dots, [n]\}.$$

Comme $\mathcal{A}_{\nu_\varphi}(x) = [\check{\mathcal{C}}_{\nu_\varphi^c}(x), \check{\mathcal{C}}_{\nu_\varphi}(x)]$ est un ensemble convexe, et $y \in \mathcal{A}_{\nu_\varphi}(x)$ on a donc :

$$y = \lambda \check{\mathcal{C}}_{\nu_\varphi^c}(x) + (1 - \lambda) \check{\mathcal{C}}_{\nu_\varphi}(x) \text{ avec } \lambda \in [0, 1]$$

$$\text{et } y = \lambda \sum_{k=1}^n x_{[k]} \cdot (\nu_\varphi^c(A_{[k]}) - \nu_\varphi^c(A_{[k+1]})) + (1 - \lambda) \sum_{k=1}^n x_{[k]} \cdot (\nu_\varphi(A_{[k]}) - \nu_\varphi(A_{[k+1]})),$$

ce qui nous donne $y = \sum_{k=1}^n x_{[k]} \cdot (\lambda((\nu_\varphi^c(A_{[k]}) - \nu_\varphi^c(A_{[k+1]})) + (1 - \lambda)((\nu_\varphi(A_{[k]}) - \nu_\varphi(A_{[k+1]}))))$

ainsi, $\psi = \{\psi_{[k]}\}_{k \in \Omega}$ est un paramètre tel que :

$$\psi_{[k]} = \lambda((\nu_\varphi^c(A_{[k]}) - \nu_\varphi^c(A_{[k+1]})) + (1 - \lambda)((\nu_\varphi(A_{[k]}) - \nu_\varphi(A_{[k+1]})))$$

et donc $y = \sum_{k=1}^n \psi_{[k]} \cdot x_{[k]} = \mathcal{A}_{P_\psi}(x)$ car la somme est commutative. \square

Nous avons établi que les bornes de l'agrégation macsum sont atteintes.

4 Calcul de l'agrégation macsum

Cette section a pour objectif de proposer un algorithme simple pour calculer $[\underline{y}, \bar{y}] = \mathcal{A}_{\nu_\varphi}(x)$.

4.1 Formules

Proposition 4.1.

$$\begin{aligned}\bar{y} &= \check{\mathcal{C}}_{\nu_\varphi}(x) = \sum_{k=1}^n \left(x_{[k]} (\alpha_k - \alpha_{k-1}) \right) + \sum_{k=1}^n \left(x_{[k]} (\beta_k - \beta_{k-1}) \right), \\ \underline{y} &= \check{\mathcal{C}}_{\nu_\varphi^c}(x) = \sum_{k=1}^n \left(x_{[k]} (\alpha_k - \alpha_{k-1}) \right) + \sum_{k=1}^n \left(x_{[k]} (\beta_k - \beta_{k-1}) \right),\end{aligned}$$

$$\begin{aligned}\text{avec } \alpha_k &= \max_{i=n-k+1}^n \varphi_{[i]}, \quad \beta_k = \min_{i=1}^k \varphi_{[i]}. \quad \text{Par convention} \\ \beta_0 &= \min_{i=1}^0 \varphi_{[i]} = 0 = \max_{i=n+1}^n \varphi_{[i]} = \alpha_0.\end{aligned}$$

où $[\cdot]$ est la permutation qui trie les éléments de x par ordre croissant et $[\cdot]$ est la permutation qui trie les éléments de x par ordre décroissant.

Preuve. Considérons d'abord \bar{y} .

$$\bar{y} = \check{\mathcal{C}}_{\nu_\varphi}(x) = \sum_{k=1}^n (x_{[k]} - x_{[k-1]}) \cdot \nu_\varphi(A_{[k]}) \text{ avec } x_{[0]} = 0$$

$$\begin{aligned}\text{et } \nu_\varphi(A_{[k]}) &= \max_{i \in A_{[k]}} \varphi_i^+ - \min_{i \in A_{[k]}^c} \varphi_i^- + \min_{i \in \Omega} \varphi_i^-, \\ \text{où } A_{[k]} &= \{[k], \dots, [n]\} \text{ et } A_{[k]}^c = \{[1], \dots, [k-1]\}.\end{aligned}$$

$$\text{Ainsi } \nu_\varphi(A_{[k]}) = \max_{i=k}^n \varphi_{[i]}^+ - \min_{i=1}^{k-1} \varphi_{[i]}^- + \min_{i \in \Omega} \varphi_i^-.$$

Par conséquent, \bar{y} peut être réécrit comme :

$$\bar{y} = \check{\mathcal{C}}_{\nu_\varphi}(x) = \sum_{k=1}^n (x_{[k]} - x_{[k-1]}) \left(\max_{i=k}^n \varphi_{[i]}^+ - \min_{i=1}^{k-1} \varphi_{[i]}^- + \min_{i \in \Omega} \varphi_i^- \right).$$

Décomposons cette somme :

$$\begin{aligned}\bar{y} &= \check{\mathcal{C}}_{\nu_\varphi}(x) = \sum_{k=1}^n (x_{[k]} - x_{[k-1]}) \max_{i=k}^n \varphi_{[i]}^+ + \sum_{k=1}^n (x_{[k]} - x_{[k-1]}) \min_{i \in \Omega} \varphi_i^- \\ &\quad - \sum_{k=1}^n (x_{[k]} - x_{[k-1]}) \min_{i=1}^{k-1} \varphi_{[i]}^-, \\ &= \dot{\bar{y}} + \ddot{\bar{y}}, \\ \text{avec } \dot{\bar{y}} &= \sum_{k=1}^n (x_{[k]} - x_{[k-1]}) \max_{i=k}^n \varphi_{[i]}^+ \\ \text{et } \ddot{\bar{y}} &= \sum_{k=1}^n (x_{[k]} - x_{[k-1]}) \min_{i \in \Omega} \varphi_i^- - \sum_{k=1}^n (x_{[k]} - x_{[k-1]}) \min_{i=1}^{k-1} \varphi_{[i]}^-\end{aligned}$$

Notons que $x_{[k]} = x_{[n-k+1]}$. En effectuant le changement d'indice : $\bar{k} = n - k + 1 \rightarrow k = n - \bar{k} + 1$, nous obtenons :

$$\dot{\bar{y}} = \sum_{\bar{k}=1}^n (x_{[\bar{k}]} - x_{[\bar{k}+1]}) \max_{i=n-\bar{k}+1}^n \varphi_{[i]}^+.$$

Maintenant, posons $\alpha_k = \max_{i=n-k+1}^n \varphi_{[i]}^+$, alors :

$$\dot{\bar{y}} = \sum_{k=1}^n (x_{[k]} - x_{[k+1]}) \alpha_k = \sum_{k=1}^n (x_{[k]}) (\alpha_k - \alpha_{k-1}).$$

Notons que : $\sum_{k=1}^n (x_{[k]} - x_{[k+1]}) = x_{[N]}$. Ainsi, $\ddot{\bar{y}} = x_{[N]} \cdot \min_{i \in \Omega} \varphi_i^- - \sum_{k=1}^n (x_{[k]} - x_{[k+1]}) \cdot \min_{i=1}^{k-1} \varphi_{[i]}^-$.

Réorganisons maintenant l'expression de $\ddot{\bar{y}}$.

$$\begin{aligned} \ddot{\bar{y}} &= x_{[n]} \min_{i=1}^n \varphi_{[i]}^- - x_{[n-1]} \min_{i=1}^{n-1} \varphi_{[i]}^- + \dots + x_{[1]} \min_{i=1}^1 \varphi_{[i]}^- \\ &= \sum_{k=1}^n x_{[k]} \left(\min_{i=1}^k \varphi_{[i]}^- - \min_{i=1}^{k-1} \varphi_{[i]}^- \right). \end{aligned}$$

Posons maintenant $\beta_k = \min_{i=1}^k \varphi_{[i]}^-$, alors $\ddot{\bar{y}} = \sum_{k=1}^n x_{[k]} (\beta_k - \beta_{k-1})$. Par conséquent,

$$\bar{y} = \dot{\bar{y}} + \ddot{\bar{y}} = \sum_{k=1}^n x_{[k]} (\alpha_k - \alpha_{k-1}) + \sum_{k=1}^n x_{[k]} (\beta_k - \beta_{k-1}).$$

\underline{y} peut être calculé de la même manière en se rappelant simplement que $y = \check{\mathbb{C}}_{\nu_\varphi^c}(x) = -\check{\mathbb{C}}_{\nu_\varphi}(-x)$. En effet, soit $[\cdot]$ la permutation qui trie les éléments de $-x$ par ordre décroissant et $[\cdot]$ la permutation qui trie les éléments de $-x$ par ordre croissant.

$$-\check{\mathbb{C}}_{\nu_\varphi}(-x) = - \sum_{k=1}^n (-x)_{[k]} (\alpha_k - \alpha_{k-1}) - \sum_{k=1}^n (-x)_{[k]} (\beta_k - \beta_{k-1}).$$

Comme la permutation qui trie les éléments de $-x$ par ordre croissant est celle qui trie les éléments de x par ordre décroissant, que la permutation qui trie les éléments de $-x$ par ordre décroissant est celle qui trie les éléments de x par ordre croissant, et que l'on a $\forall k \in \Omega$, $(-x)_{[k]} = -x_{[k]}$ et $(-x)_{[k]} = -x_{[k]}$, nous avons que :

$$\begin{aligned} \underline{y} &= -\check{\mathbb{C}}_{\nu_\varphi}(-x) = - \sum_{k=1}^n -x_{[k]} (\alpha_k - \alpha_{k-1}) - \sum_{k=1}^n -x_{[k]} (\beta_k - \beta_{k-1}), \\ &= \sum_{k=1}^n x_{[k]} (\alpha_k - \alpha_{k-1}) + \sum_{k=1}^n x_{[k]} (\beta_k - \beta_{k-1}). \end{aligned}$$

Ainsi, les algorithmes sont les mêmes, nous devons simplement échanger la permutation qui trie les éléments de x en ordre décroissant par la permutation qui trie les éléments de x en ordre croissant, et vice versa.

□

Les valeurs de α_k et β_k peuvent être calculées de manière récursive.

Proposition 4.2. *Soient $\alpha, \beta \in \mathbb{R}^n$ définis par $\forall k \in \Omega$, $\alpha_k = \max_{i=n-k+1}^n \varphi_{[i]}^+$ et $\beta_k = \min_{i=1}^k \varphi_{[i]}^-$, avec $\alpha_0 = 0$ et $\beta_0 = 0$, alors $\alpha_k = \max(\alpha_{k-1}, \varphi_{[k]}^+)$ et $\beta_k = \min(\beta_{k-1}, \varphi_{[k]}^-)$.*

La preuve est triviale et découle directement des définitions de α et β .

4.2 Algorithmes

L'algorithme pour calculer \bar{y} est fourni dans l'Algorithme 1, tandis que l'algorithme pour calculer y est fourni dans l'Algorithme 2.

Algorithm 1: Calcul de \bar{y}

Input: $x = \{x_i\}_{i=1,\dots,n}$, $\varphi = \{\varphi_i\}_{i=1,\dots,n}$

Output: \bar{y}

trie (x, φ) par rapport à x dans l'ordre décroissant ;

$\alpha = 0$, $\beta = 0$;

$\bar{y} = 0$;

for $k = 1$ **to** n **do**

$\beta = \alpha$;

$\alpha = \max(\alpha, \varphi_k)$;

$\bar{y} = \bar{y} + (\alpha - \beta) \cdot x_k$;

trie (x, φ) par rapport à x dans l'ordre croissant (i.e. inversé le trie) ;

$\alpha = 0$, $\beta = 0$;

for $k = 1$ **to** n **do**

$\beta = \alpha$;

$\alpha = \min(\alpha, \varphi_k)$;

$\bar{y} = \bar{y} + (\alpha - \beta) \cdot x_k$;

Dans cette section, nous avons présenté un algorithme simple et efficace pour le calcul de l'agrégation macsum. Nous avons commencé par introduire les formules nécessaires sous forme de propositions mathématiques. Ensuite, nous avons prouvé la validité de ces formules, et en nous basant sur celles-ci, nous avons développé deux algorithmes pour calculer les deux bornes de l'intervalle de sortie de l'agrégation macsum.

Algorithm 2: Calcul de \underline{y} **Input:** $x = \{x_i\}_{i=1,\dots,n}$, $\varphi = \{\varphi_i\}_{i=1,\dots,n}$ **Output:** \underline{y} trie (x, φ) par rapport x dans l'ordre croissant ; $\alpha = 0, \beta = 0$; $\underline{y} = 0$;**for** $k = 1$ **to** n **do**

$\beta = \alpha$;
$\alpha = \max(\alpha, \varphi_k)$;
$\underline{y} = \underline{y} + (\alpha - \beta) \cdot x_k$;

trie (x, φ) par rapport à x dans l'ordre décroissant (i.e. inversé le trie) ; $\alpha = 0, \beta = 0$;**for** $k = 1$ **to** n **do**

$\beta = \alpha$;
$\alpha = \min(\alpha, \varphi_k)$;
$\underline{y} = \underline{y} + (\alpha - \beta) \cdot x_k$;

5 Conclusion

Ce chapitre a introduit l'agrégation macsum comme une généralisation de l'agrégation maxitive. Cette agrégation a pour objet le calcul d'un intervalle à partir d'un vecteur de paramètres de valeurs précises et d'un vecteur d'entrées de même dimension. Les bornes supérieures et inférieures de cet intervalle sont obtenues respectivement par le calcul d'une intégrale de Choquet discrètes par rapport à l'opérateur macsum et par rapport au complémentaire de l'opérateur macsum.

L'opérateur macsum, par sa concavité, domine un ensemble convexe d'opérateurs additifs. Comme l'intégrale de Choquet par rapport à un opérateur additif est une agrégation linéaire, l'agrégation macsum représente un ensemble convexe d'agrégations linéaires de même gain. Cela peut lui permettre d'offrir un large éventail de représentations de systèmes, tout en alliant la simplicité des modèles linéaires à la fiabilité induite par des sorties imprécises.

L'agrégation macsum est un nouveau modèle intervalliste pour la modélisation de systèmes à partir de données entrée/sortie. La prochaine étape de notre travail est l'apprentissage de ce modèle.

Chapitre 4

Un nouveau modèle d'apprentissage intervalliste

Objectifs

L'objectif de ce chapitre est d'établir la pertinence de l'utilisation de l'agrégation macsum en tant que modèle d'apprentissage automatique. Pour cela, on démontre les propriétés qui en font un modèle simple d'utilisation mais capable de représenter un ensemble convexe de modèles linéaires. Enfin, pour appuyer nos propos, nous concluons par une expérimentation en apprentissage dans le cadre du traitement d'image. Ce chapitre permet l'introduction d'un nouveau modèle d'apprentissage intervalliste.

Sommaire

1	Introduction	39
2	Aspects préliminaires	40
	2.1 Non vacuité du coeur macsumitif	40
	2.2 Formulation de l'agrégation macsum facilitant la dérivation	41
	2.3 Dérivation de l'agrégation macsum	46
3	Apprentissage de l'agrégation macsum	49
	3.1 Descente de gradient	49
	3.2 Fonction de coût pour une sortie intervallistes	49
	3.3 Expérimentation	51
4	Conclusion	57

1 Introduction

L'apprentissage supervisé vise à entraîner un modèle en utilisant un ensemble de données qui contient des entrées du système à mimer et leurs sorties associées. Le but est d'amener le modèle à généraliser les tendances présentes dans cet ensemble afin qu'il puisse prédire de manière fiable des sorties du système pour des entrées inédites. Pour évaluer la fiabilité du modèle, on compare ses prédictions aux données réelles et on mesure cet écart à travers une fonction de coût. Dans l'optique d'augmenter la fiabilité du modèle en ajustant ses paramètres, l'idée est de minimiser l'écart entre les prédictions du modèle et les sorties du système et donc de minimiser la fonctions de coût. Dans le processus de minimisation de cette fonction, une technique couramment employée est la descente de gradient. Elle nécessite de calculer le gradient de la fonction de coût par rapport aux paramètres du modèle pour déterminer comment les ajuster. Étant donné que la fonction de coût est composée avec le modèle (elle prend en argument à la fois les prédictions du modèle et les étiquettes réelles), la mise à jour des paramètres requiert, pour calculer la dérivée de la fonction de coût par rapport aux paramètres, celle du modèle lui-même.

La dérivée joue ici un rôle crucial. Elle nous indique comment la fonction de coût change lorsque les paramètres du modèle sont modifiés. En d'autres termes, la dérivée reflète les variations de la fonction de coût par rapport à chaque paramètre. Grâce à cette information, la descente de gradient est capable de déterminer dans quelle direction il faut ajuster les paramètres pour approcher au plus près le minimum de la fonction de coût, et ainsi améliorer la fiabilité du modèle. Notre objectif est d'utiliser l'agrégation macsum en tant que modèle d'apprentissage supervisé. Pour ce faire, il faut calculer la dérivée de l'agrégation macsum vis-à-vis de ses paramètres. Nous proposons de réduire la définition de cette dérivée à celle de ses bornes. Nous présentons dans ce chapitre une reformulation du calcul des bornes de l'agrégation macsum qui simplifie le calcul de leur dérivée.

Par ailleurs, l'un des atouts majeurs de ce modèle réside dans le fait qu'il représente un ensemble convexe d'agrégations linéaires à l'aide d'un unique vecteur de paramètres et ce peu importe les modifications apportées à ce paramètre. Cette propriété, démontrée dans ce chapitre, découle directement du fait que le cœur macsumitif d'un paramètre est toujours non vide. Cela signifie que l'opérateur macsum associé à n'importe quel paramètre domine toujours au moins un opérateur additif. C'est une des forces de l'agrégation macsum de permettre l'ajustement de ses paramètres via l'apprentissage sans compromettre sa représentativité. À l'opposé, l'agrégation maxitive, si utilisée en apprentissage, ne pourrait pas conserver ses propriétés sans imposer des contraintes de positivités et de normalisations des paramètres lors de leur ajustement. C'est donc à travers l'utilisation de l'agrégation macsum en tant que modèle d'apprentissage que la pertinence de la généralisation des travaux de Loquin concernant l'agrégation maxitive se manifeste [Hmidy, 2023b]. Cette approche est relativement nouvelle, car l'agrégation macsum est basée sur une intégrale de Choquet et très peu de travaux ont proposé d'apprendre une fonction d'agrégation basée sur cette intégrale.

2 Aspects préliminaires

Dans cette section, nous explorons certains aspects fondamentaux de l'agrégation macsum comme modèle potentiellement représentatif de systèmes incohérents dans le contexte de l'apprentissage automatique. Le premier aspect abordé est la non-vacuité du cœur macsumitif d'un paramètre. Cette caractéristique assure que, peu importe les valeurs du paramètre appris, l'agrégation macsum représente un ensemble d'agrégation linéaire. Ensuite, nous proposons une reformulation de l'agrégation macsum de manière à réduire la complexité du calcul de sa dérivée par rapport à ses paramètres.

2.1 Non vacuité du cœur macsumitif

Parmi les modèles paramétriques, l'agrégation linéaire s'est imposée principalement pour deux de ses qualités intrinsèques. Premièrement, sa simplicité structurelle la rend peu coûteuse en calcul. Deuxièmement, son fort potentiel de représentation offre une manière pertinente de synthétiser les données [Grabisch, 2009].

L'agrégation macsum, à partir d'un vecteur de paramètres et d'un vecteur d'entrées de même dimension, calcule un intervalle dont la borne supérieure et la borne inférieure sont des agrégations non-linéaires. Nous démontrons que pour chaque paramètre appris, l'intervalle de sortie de l'agrégation macsum contient au moins une sortie d'agrégation linéaire, c'est-à-dire que le cœur macsumitif d'un paramètre n'est jamais vide.

Proposition 2.1. $\forall \varphi \in \mathbb{R}^n$, $\mathcal{M}(\varphi)$ n'est pas vide.

Preuve. Pour prouver la Proposition 2.1, il suffit d'être en mesure d'associer à chaque paramètre $\varphi \in \mathbb{R}^n$ qu'il existe au moins un paramètre $\psi \in \mathbb{R}^n$ tel que $\psi \in \mathcal{M}(\varphi)$. Cette preuve repose sur la construction de ψ de telle sorte que φ domine ψ . Supposons que φ est le paramètre qui domine fortement ψ , alors comme montré dans le chapitre 3 : $\varphi_i = \sum_{j \in A_i^+} \psi_j^+ + \sum_{j \in A_i^-} \psi_j^-$, où $\psi^+ = \max(0, \psi)$, $\psi^- = \min(0, \psi)$, $A_i^+ = \{j \in \Omega / \psi_j^+ \leq \varphi_i^+\}$ et $A_i^- = \{j \in \Omega / \psi_j^- \geq \varphi_i^-\}$. Nous pouvons supposer, sans perte de généralité, que les éléments de φ sont triés par ordre croissant : $\varphi_1 \leq \varphi_2 \leq \dots \leq \varphi_n$. Soient $\varphi^+ = \max(0, \varphi)$ et $\varphi^- = \min(0, \varphi)$, c'est-à-dire $\varphi = \varphi^+ + \varphi^-$.

Si φ domine fortement ψ , $\varphi_i^+ = \sum_{j=1}^i \psi_j^+$ et $\varphi_i^- = \sum_{j=i}^n \psi_j^-$.

Sur cette base, nous pouvons inverser cette construction pour construire itérativement les valeurs de ψ^+ et ψ^- :

$$\forall i \in \{2, \dots, n\}, \quad \psi_i^+ = \varphi_i^+ - \sum_{j=1}^{i-1} \psi_j^+, \quad \text{avec } \psi_1^+ = \varphi_1^+ \text{ et}$$

$$\forall i \in \{n-1, \dots, 1\}, \quad \psi_i^- = \varphi_i^- - \sum_{j=i+1}^n \psi_j^-, \quad \text{avec } \psi_n^- = \varphi_n^-.$$

Cela garantit que le paramètre ψ ainsi construit appartient à $\mathcal{M}(\varphi)$ et donc $\mathcal{M}(\varphi) \neq \emptyset$. \square

Nous avons donc prouvé, au cours de cette section, la non-vacuité du cœur de l'opérateur macsum pour tout paramètre, ce qui garantit que l'on peut ajuster les paramètres de l'agrégation macsum sans altérer sa capacité à représenter un ensemble convexe d'agrégation linéaire.

2.2 Formulation de l'agrégation macsum facilitant la dérivation

Dans cette section, nous reformulons l'agrégation macsum de manière à la rendre linéaire par rapport à ses paramètres et ainsi faciliter sa dérivation en vue de l'utilisation de la méthode de descente de gradient. Cette nouvelle formulation de l'agrégation macsum se base sur une proposition établie et démontrée par Dubois et Rico dans le cadre de la théorie des possibilités [Dubois, 2016] :

Proposition 2.2. *Soit π un noyau maxitif. Soit Π_π la mesure de possibilité définie par $\forall A \subseteq \Omega$, $\Pi_\pi(A) = \max_{i \in A} \pi_i$, alors :*

$$\begin{aligned} \check{C}_{\Pi_\pi}(x) &= \sum_{k=1}^n (x_{(k)} - x_{(k-1)}) \cdot \Pi_\pi(A_{(k)}) = \sum_{k=1}^n (x_{(k)} - x_{(k-1)}) \cdot \max_{i=k}^n \pi(i) \\ &= \sum_{k=1}^n (\pi_{[k]} - \pi_{[k+1]}) \max_{i=1}^k x_{[i]}, \text{ et} \\ \check{C}_{\Pi_\pi^c}(x) &= \sum_{k=1}^n (x_{(k)} - x_{(k-1)}) \cdot \Pi_\pi^c(A_{(k)}) = x_{(n)} - \sum_{k=1}^n (x_{(k)} - x_{(k-1)}) \cdot \max_{i=1}^{k-1} \pi(i) \\ &= \sum_{k=1}^n (\pi_{[k]} - \pi_{[k+1]}) \min_{i=1}^k x_{[i]} \end{aligned}$$

où $(.)$ est une permutation qui trie les éléments du vecteur x en ordre croissant :

$$x_{(1)} \leq x_{(2)} \leq \dots \leq x_{(n)}, \quad x_{(0)} = 0,$$

$[.]$ est une permutation qui trie les éléments du noyau π en ordre décroissant :

$$1 = \pi_{[1]} \geq \pi_{[2]} \geq \dots \geq \pi_{[n]} \text{ et } \pi_{[n+1]} = 0.$$

La proposition décrit une forme d'intégrale de Choquet par rapport à une mesure de possibilité. Nous l'employons pour établir une formulation de l'intégrale de Choquet par rapport à l'opérateur macsum qui réduit le coût de calcul de sa dérivation.

Proposition 2.3.

$$\bar{y} = \check{C}_{\nu_\varphi}(x) = \sum_{k=1}^n (\varphi_{[k]}^+ - \varphi_{[k+1]}^+) \cdot \max_{i=1}^k x_{[i]} + \sum_{k=1}^n (\varphi_{[k]}^- - \varphi_{[k+1]}^-) \cdot \min_{i=1}^k x_{[i]}, \quad (4.1)$$

$$\underline{y} = \check{C}_{\nu_\varphi^c}(x) = \sum_{k=1}^n (\varphi_{[k]}^+ - \varphi_{[k+1]}^+) \cdot \min_{i=1}^k x_{[i]} + \sum_{k=1}^n (\varphi_{[k]}^- - \varphi_{[k+1]}^-) \cdot \max_{i=1}^k x_{[i]}, \quad (4.2)$$

où $[.]$ est une permutation qui trie les éléments de φ en ordre décroissant

$$\varphi_{[1]} \geq \varphi_{[2]} \geq \dots \geq \varphi_{[n]}$$

et $[.]$ est une permutation qui trie les éléments de φ en ordre croissant,

$$\varphi_{[1]} \leq \varphi_{[2]} \leq \dots \leq \varphi_{[n]},$$

avec $\varphi_{[n+1]} = 0$ et $\varphi_{[n+1]} = 0$.

Preuve. Nous allons d'abord prouver cette proposition pour \bar{y} . Nous avons trois cas :

- 1) $\exists i, j \in \Omega$ tels que $\varphi_i > 0$ et $\varphi_j < 0$,
- 2) $\forall i \in \Omega, \varphi_i \geq 0$,
- 3) $\forall i \in \Omega, \varphi_i \leq 0$.

Cas 1) $\exists i, j \in \Omega$ tel que $\varphi_i > 0$ et $\varphi_j < 0$.

Alors, $\min_{i \in \Omega} \varphi_i^+ = 0, \max_{i \in \Omega} \varphi_i^- = 0$, donc $\underline{\alpha} < 0$ et $\bar{\alpha} > 0$.

Définissons deux noyaux maxitifs π^+ et π^- par $\forall i \in \Omega, \pi_i^+ = \frac{1}{\bar{\alpha}} \cdot \varphi_i^+$ et $\forall i \in \Omega, \pi_i^- = \frac{1}{\underline{\alpha}} \cdot \varphi_i^-$.

Par construction, $\min_{i \in \Omega} \pi_i^- = \min_{i \in \Omega} \pi_i^+ = 0$ et $\max_{i \in \Omega} \pi_i^- = \max_{i \in \Omega} \pi_i^+ = 1$.

De plus, $[\cdot]$ trie les éléments de π^+ en ordre décroissant ($1 = \pi_{[1]}^+ \geq \pi_{[2]}^+ \geq \dots \geq \pi_{[n]}^+ = 0$) et $[\cdot]$ trie les éléments de π^- en ordre décroissant ($1 = \pi_{[1]}^- \geq \pi_{[2]}^- \geq \dots \geq \pi_{[n]}^- = 0$).

Soit (\cdot) une permutation qui trie les éléments de x en ordre croissant ($x_{(1)} \leq x_{(2)} \leq \dots \leq x_{(n)}$), avec $x_{(0)} = 0$. Réécrivons $\check{C}_{\nu_\varphi}(x)$:

$$\begin{aligned} \bar{y} &= \check{C}_{\nu_\varphi}(x) = \sum_{k=1}^n (x_{(k)} - x_{(k-1)}) \cdot \max_{i=k}^n \varphi_i^+ + \underline{\alpha} \cdot x_{(n)} - \sum_{k=1}^n (x_{(k)} - x_{(k-1)}) \cdot \min_{i=1}^{k-1} \varphi_i^-, \\ &= \bar{\alpha} \cdot \sum_{k=1}^n (x_{(k)} - x_{(k-1)}) \cdot \max_{i=k}^n \pi_i^+ + \underline{\alpha} \cdot x_{(n)} - \underline{\alpha} \cdot \sum_{k=1}^n (x_{(k)} - x_{(k-1)}) \cdot \max_{i=1}^{k-1} \pi_i^-, \\ &= \bar{\alpha} \cdot \sum_{k=1}^n (x_{(k)} - x_{(k-1)}) \cdot \max_{i=k}^n \pi_i^+ + \underline{\alpha} \cdot \left(x_{(n)} - \sum_{k=1}^n (x_{(k)} - x_{(k-1)}) \cdot \max_{i=1}^{k-1} \pi_i^- \right). \end{aligned}$$

En raison de la proposition 2.2, nous avons :

$$\begin{aligned} \sum_{k=1}^n (x_{(k)} - x_{(k-1)}) \cdot \max_{i=k}^n \pi_i^+ &= \sum_{k=1}^n (\pi_{[k]}^+ - \pi_{[k+1]}^+) \cdot \max_{i=1}^k x_{[i]}, \text{ et} \\ x_{(n)} - \sum_{k=1}^n (x_{(k)} - x_{(k-1)}) \cdot \max_{i=1}^{k-1} \pi_i^- &= \sum_{k=1}^n (\pi_{[k]}^- - \pi_{[k+1]}^-) \cdot \min_{i=1}^k x_{[i]}. \end{aligned}$$

Ainsi,

$$\begin{aligned} \bar{y} &= \check{C}_{\nu_\varphi}(x) = \bar{\alpha} \cdot \sum_{k=1}^n (\pi_{[k]}^+ - \pi_{[k+1]}^+) \cdot \max_{i=1}^k x_{[i]} + \underline{\alpha} \cdot \sum_{k=1}^n (\pi_{[k]}^- - \pi_{[k+1]}^-) \cdot \min_{i=1}^k x_{[i]}, \\ &= \sum_{k=1}^n (\bar{\alpha} \cdot \pi_{[k]}^+ - \bar{\alpha} \cdot \pi_{[k+1]}^+) \cdot \max_{i=1}^k x_{[i]} + \sum_{k=1}^n (\underline{\alpha} \cdot \pi_{[k]}^- - \underline{\alpha} \cdot \pi_{[k+1]}^-) \cdot \min_{i=1}^k x_{[i]}, \\ &= \sum_{k=1}^n (\varphi_{[k]}^+ - \varphi_{[k+1]}^+) \cdot \max_{i=1}^k x_{[i]} + \sum_{k=1}^n (\varphi_{[k]}^- - \varphi_{[k+1]}^-) \cdot \min_{i=1}^k x_{[i]}. \end{aligned}$$

Cas 2) $\forall i \in \Omega, \varphi_i \geq 0$.

Alors, $\underline{\alpha} \cdot x_{(n)} - \sum_{k=1}^n (x_{(k)} - x_{(k-1)}) \cdot \min_{i=1}^{k-1} \varphi_i^- = 0$. Comme précédemment, soit π^+ le noyau maxitif tel que : $\forall i \in \Omega, \pi_i^+ = \frac{1}{\bar{\alpha}} \cdot \varphi_i^+$.

Alors,

$$\bar{y} = \check{C}_{\nu_\varphi}(x) = \bar{\alpha} \cdot \sum_{k=1}^n (\pi_{[k]}^+ - \pi_{[k+1]}^+) \cdot \max_{i=1}^k x_{[i]} = \sum_{k=1}^n (\varphi_{[k]}^+ - \varphi_{[k+1]}^+) \cdot \max_{i=1}^k x_{[i]}.$$

Cas 3) $\forall i \in \Omega, \varphi_i \leq 0$.

Ainsi, $\sum_{k=1}^n (x_{(k)} - x_{(k-1)}) \cdot \max_{i=k}^n \varphi_{(i)}^+ = 0$. Comme précédemment, soit π^- le noyau maxitif tel que : $\forall i \in \Omega, \pi_i^- = \frac{1}{\alpha} \cdot \varphi_i^-$. Ainsi, la borne supérieure de l'agrégation macsum peut s'écrire :

$$\bar{y} = \check{C}_{\nu_\varphi}(x) = \alpha \cdot \sum_{k=1}^n (\pi_{[k]}^- - \pi_{[k+1]}^-) \cdot \min_{i=1}^k x_{[i]} = \sum_{k=1}^n (\varphi_{[k]}^- - \varphi_{[k+1]}^-) \cdot \min_{i=1}^k x_{[i]}.$$

Le calcul de \underline{y} est obtenu en considérant l'équation $\underline{y} = \check{C}_{\nu_\varphi^c}(x) = -\check{C}_{\nu_\varphi}(-x)$.

$$\begin{aligned} \underline{y} &= -\check{C}_{\nu_\varphi}(-x) = -\sum_{k=1}^n (\varphi_{[k]}^+ - \varphi_{[k+1]}^+) \cdot \max_{i=1}^k -x_{[i]} - \sum_{k=1}^n (\varphi_{[k]}^- - \varphi_{[k+1]}^-) \cdot \min_{i=1}^k -x_{[i]}, \\ &= \sum_{k=1}^n (\varphi_{[k]}^+ - \varphi_{[k+1]}^+) \cdot \left(-\max_{i=1}^k -x_{[i]} \right) - \sum_{k=1}^n (\varphi_{[k]}^- - \varphi_{[k+1]}^-) \cdot \left(-\min_{i=1}^k -x_{[i]} \right), \\ &= \sum_{k=1}^n (\varphi_{[k]}^+ - \varphi_{[k+1]}^+) \cdot \min_{i=1}^k x_{[i]} - \sum_{k=1}^n (\varphi_{[k]}^- - \varphi_{[k+1]}^-) \cdot \max_{i=1}^k x_{[i]}. \end{aligned}$$

□

Exemple de calcul :

Supposons que nous avons $\Omega = \{1, 2, 3, 4\}$, $\mathbf{x} = (0.5, 0.2, 0.7, 0.9)$ et $\boldsymbol{\varphi} = (2.5, -1.5, 1.0, -2.0)$.

1. Calculons d'abord \bar{y} en utilisant la formule (4.1) :

$$\begin{aligned} \bar{y} &= \sum_{k=1}^N (\varphi_{[k]}^+ - \varphi_{[k+1]}^+) \max_{i=1}^k x_{[i]} + \sum_{k=1}^N (\varphi_{[k]}^- - \varphi_{[k+1]}^-) \min_{i=1}^k x_{[i]} \\ &= (2.5 - (-1.5)) \times 0.5 + (-2.0 - 1.0) \times 0.2 \\ &\quad + ((-1.5) - 1.0) \times 0.5 + (1.0 - (-2.0)) \times 0.9 \\ &\quad + (1.0 - (-2.0)) \times 0.7 + ((-1.5) - 2.5) \times 0.2 \\ &\quad + (-2.0 - 0) \times 0.9 \\ &= 2 \times 0.5 + (-3) \times 0.2 + (-2.5) \times 0.5 + 3 \times 0.9 \\ &\quad + 3 \times 0.7 + (-4) \times 0.2 + (-2.0) \times 0.9 \\ &= 2 + (-0.6) + (-1.25) + 2.7 + 2.1 + (-0.8) + (-1.8) \\ &= 1.15. \end{aligned}$$

2. Calculons maintenant \underline{y} en utilisant la formule (4.2) :

$$\begin{aligned}
 \underline{y} &= \sum_{k=1}^N (\varphi_{[k]}^+ - \varphi_{[k+1]}^+) \min_{i=1}^k x_{[i]} + \sum_{k=1}^N (\varphi_{[k]}^- - \varphi_{[k+1]}^-) \max_{i=1}^k x_{[i]} \\
 &= (2.5 - (-1.5)) \times 0.5 + (-2.0 - 1.0) \times 0.9 \\
 &\quad + ((-1.5) - 1.0) \times 0.2 + (1.0 - (-2.0)) \times 0.7 \\
 &\quad + (1.0 - (-2.0)) \times 0.5 + ((-1.5) - 2.5) \times 0.9 \\
 &\quad + (-2.0 - 0) \times 0.7 \\
 &= (2.5 - (-1.5)) \times 0.5 + (-3) \times 0.9 \\
 &\quad + (-1.5 - 1.0) \times 0.2 + 3 \times 0.7 \\
 &\quad + 3 \times 0.5 + (-4) \times 0.9 \\
 &\quad + (-2.0) \times 0.7 \\
 &= 1.15 + (-2.7) + (-1.5) + 2.1 + 1.5 + (-3.6) + (-1.4) \\
 &= -4.6.
 \end{aligned}$$

Par conséquent, dans cet exemple, l'intervalle $[\underline{y}, \bar{y}]$ est $[-4.6, 1.15]$.

En s'appuyant sur la proposition 2.2, nous avons pu développer une formulation alternative de l'agrégation macsum. Afin de faciliter la dérivation de l'agrégation macsum par rapport à ses paramètres, nous proposons de simplifier les équations (4.2) et (4.1) en les reformulant :

Proposition 2.4.

$$\bar{y} = \check{C}_{\nu_{\varphi}}(x) = \sum_{k=1}^n \varphi_{[k]}^+ \cdot \left(\max_{i=1}^k x_{[i]} - \max_{i=1}^{k-1} x_{[i]} \right) + \sum_{k=1}^n \varphi_{[k]}^- \cdot \left(\min_{i=1}^k x_{[k]} - \min_{i=1}^{k-1} x_{[k-1]} \right), \quad (4.3)$$

$$\underline{y} = \check{C}_{\nu_{\varphi}^c}(x) = \sum_{k=1}^n \varphi_{[k]}^+ \cdot \left(\min_{i=1}^k x_{[i]} - \min_{i=1}^{k-1} x_{[i]} \right) + \sum_{k=1}^n \varphi_{[k]}^- \cdot \left(\max_{i=1}^k x_{[i]} - \max_{i=1}^{k-1} x_{[i]} \right), \quad (4.4)$$

Par convention $\max_{i=1}^0 x_{[i]} = 0 = \min_{i=1}^0 x_{[i]}$.

Preuve. L'Equation (4.1) donne :

$$\check{C}_{\nu_{\varphi}}(x) = \sum_{k=1}^n (\varphi_{[k]}^+ - \varphi_{[k+1]}^+) \cdot \max_{i=1}^k x_{[i]} + \sum_{k=1}^n (\varphi_{[k]}^- - \varphi_{[k+1]}^-) \cdot \min_{i=1}^k x_{[i]}.$$

Réécrivons :

$$\begin{aligned}
 \sum_{k=1}^n \left(\varphi_{[k]}^+ - \varphi_{[k+1]}^+ \right) \cdot \max_{i=1}^k x_{[i]} &= \sum_{k=1}^n \varphi_{[k]}^+ \max_{i=1}^k x_{[i]} - \sum_{k=1}^n \varphi_{[k+1]}^+ \max_{i=1}^k x_{[i]} \\
 &= \sum_{k=1}^n \varphi_{[k]}^+ \max_{i=1}^k x_{[i]} - \sum_{k=2}^{n+1} \varphi_{[k]}^+ \max_{i=1}^{k-1} x_{[i]} \\
 &= \sum_{k=1}^n \varphi_{[k]}^+ \max_{i=1}^k x_{[i]} - \sum_{k=1}^n \varphi_{[k]}^+ \max_{i=1}^{k-1} x_{[i]} \\
 &\quad - \varphi_{[n+1]}^+ \cdot \max_{i=1}^n x_{[i]} + \varphi_{[1]}^+ \cdot \max_{i=1}^0 x_{[i]} \\
 &= \sum_{k=1}^n \varphi_{[k]}^+ \cdot \left(\max_{i=1}^k x_{[i]} - \max_{i=1}^{k-1} x_{[i]} \right),
 \end{aligned}$$

en posant $\varphi_{[n+1]}^+ = 0$, et

$$\begin{aligned}
 \sum_{k=1}^n \left(\varphi_{[k]}^- - \varphi_{[k+1]}^- \right) \cdot \min_{i=1}^k x_{[i]} &= \sum_{k=1}^n \varphi_{[k]}^- \min_{i=1}^k x_{[i]} - \sum_{k=1}^n \varphi_{[k+1]}^- \min_{i=1}^k x_{[i]} \\
 &= \sum_{k=1}^n \varphi_{[k]}^- \min_{i=1}^k x_{[i]} - \sum_{k=2}^{n+1} \varphi_{[k]}^- \min_{i=1}^{k-1} x_{[i]} \\
 &= \sum_{k=1}^n \varphi_{[k]}^- \min_{i=1}^k x_{[i]} - \sum_{k=1}^n \varphi_{[k]}^- \min_{i=1}^{k-1} x_{[i]} \\
 &\quad - \varphi_{[n+1]}^- \cdot \min_{i=1}^n x_{[i]} + \varphi_{[1]}^- \cdot \min_{i=1}^0 x_{[i]} \\
 &= \sum_{k=1}^n \varphi_{[k]}^- \cdot \left(\min_{i=1}^k x_{[k]} - \min_{i=1}^{k-1} x_{[k-1]} \right),
 \end{aligned}$$

en posant $\varphi_{[n+1]}^- = 0$.

Ainsi,

$$\check{C}_{\nu_\varphi}(x) = \sum_{k=1}^n \varphi_{[k]}^+ \cdot \left(\max_{i=1}^k x_{[i]} - \max_{i=1}^{k-1} x_{[i]} \right) + \sum_{k=1}^n \varphi_{[k]}^- \cdot \left(\min_{i=1}^k x_{[i]} - \min_{i=1}^{k-1} x_{[i]} \right).$$

Prouver le calcul de \underline{y} est triviale en considérant l'équation $\check{C}_{\nu_\varphi^c}(x) = -\check{C}_{\nu_\varphi}(-x)$. En effet,

$$\begin{aligned}
 \underline{y} = -\check{C}_{\nu_\varphi}(-x) &= -\sum_{k=1}^n \varphi_{[k]}^+ \cdot \left(\max_{i=1}^k (-x)_{[i]} - \max_{i=1}^{k-1} (-x)_{[i]} \right) - \sum_{k=1}^n \varphi_{[k]}^- \cdot \left(\min_{i=1}^k (-x)_{[i]} - \min_{i=1}^{k-1} (-x)_{[i]} \right), \\
 &= \sum_{k=1}^n \varphi_{[k]}^+ \cdot \left(-\max_{i=1}^k (-x)_{[i]} - \left(-\max_{i=1}^{k-1} (-x)_{[i]} \right) \right) \\
 &\quad + \sum_{k=1}^n \varphi_{[k]}^- \cdot \left(-\min_{i=1}^k (-x)_{[i]} - \left(-\min_{i=1}^{k-1} (-x)_{[i]} \right) \right), \\
 &= \sum_{k=1}^n \varphi_{[k]}^+ \cdot \left(\min_{i=1}^k x_{[i]} - \min_{i=1}^{k-1} x_{[i]} \right) + \sum_{k=1}^n \varphi_{[k]}^- \cdot \left(\max_{i=1}^k x_{[i]} - \max_{i=1}^{k-1} x_{[i]} \right).
 \end{aligned}$$

□

2.3 Dérivation de l'agrégation macsum

Dans cette section, nous présentons la dérivation de l'agrégation macsum par rapport à chaque composant du paramètre φ . Nous utilisons pour cela les formules (4.3) et (4.4).

Proposition 2.5. *Pour tout $k \in \Omega$, soit l l'indice tel que $\lfloor l \rfloor = k$ et u l'indice tel que $\lceil u \rceil = k$, alors :*

$$\frac{\delta \check{\mathcal{C}}_{\nu_\varphi}(x)}{\delta \varphi_k} = \left(\max_{i=1}^l x_{\lfloor i \rfloor} - \max_{i=1}^{l-1} x_{\lfloor i \rfloor} \right) + \left(\min_{i=1}^u x_{\lceil i \rceil} - \min_{i=1}^{u-1} x_{\lceil i \rceil} \right),$$

et

$$\frac{\delta \check{\mathcal{C}}_{\nu_\varphi^c}(x)}{\delta \varphi_k} = \left(\min_{i=1}^l x_{\lfloor i \rfloor} - \min_{i=1}^{l-1} x_{\lfloor i \rfloor} \right) + \left(\max_{i=1}^u x_{\lceil i \rceil} - \max_{i=1}^{u-1} x_{\lceil i \rceil} \right).$$

avec $\min_{i=1}^0 x_{\lfloor i \rfloor} = 0 = \max_{i=1}^0 x_{\lceil i \rceil}$.

Preuve. Tout d'abord, rappelons que $\forall k \in \Omega$, $\varphi_k^+ = \max(\varphi_k, 0)$ et $\varphi_k^- = \min(\varphi_k, 0)$ et donc $\varphi_k = \varphi_k^+ + \varphi_k^-$ et $\delta \varphi_k = \delta \varphi_k^+ + \delta \varphi_k^-$. Par conséquent,

$$\begin{aligned} \frac{\delta \check{\mathcal{C}}_{\nu_\varphi^c}(x)}{\delta \varphi_k} &= \frac{\delta \check{\mathcal{C}}_{\nu_\varphi^c}(x)}{\delta \varphi_k^+} \cdot \frac{\delta \varphi_k^+}{\delta \varphi_k} + \frac{\delta \check{\mathcal{C}}_{\nu_\varphi^c}(x)}{\delta \varphi_k^-} \cdot \frac{\delta \varphi_k^-}{\delta \varphi_k}, \\ \frac{\delta \check{\mathcal{C}}_{\nu_\varphi}(x)}{\delta \varphi_k} &= \frac{\delta \check{\mathcal{C}}_{\nu_\varphi}(x)}{\delta \varphi_k^+} \cdot \frac{\delta \varphi_k^+}{\delta \varphi_k} + \frac{\delta \check{\mathcal{C}}_{\nu_\varphi}(x)}{\delta \varphi_k^-} \cdot \frac{\delta \varphi_k^-}{\delta \varphi_k}. \end{aligned}$$

En considérant

$$\frac{\delta \varphi_k}{\delta \varphi_k^+} = \frac{\delta \varphi_k^+}{\delta \varphi_k^+} + \frac{\delta \varphi_k^-}{\delta \varphi_k^+} = 1$$

ainsi que

$$\frac{\delta \varphi_k}{\delta \varphi_k^-} = \frac{\delta \varphi_k^+}{\delta \varphi_k^-} + \frac{\delta \varphi_k^-}{\delta \varphi_k^-} = 1$$

on obtient donc :

$$\frac{\delta \check{\mathcal{C}}_{\nu_\varphi^c}(x)}{\delta \varphi_k} = \frac{\delta \check{\mathcal{C}}_{\nu_\varphi^c}(x)}{\delta \varphi_k^+} + \frac{\delta \check{\mathcal{C}}_{\nu_\varphi^c}(x)}{\delta \varphi_k^-} \text{ et } \frac{\delta \check{\mathcal{C}}_{\nu_\varphi}(x)}{\delta \varphi_k} = \frac{\delta \check{\mathcal{C}}_{\nu_\varphi}(x)}{\delta \varphi_k^+} + \frac{\delta \check{\mathcal{C}}_{\nu_\varphi}(x)}{\delta \varphi_k^-}.$$

Maintenant, prouver que la Proposition 2.5 est direct en considérant la dérivation des équations (4.3) et (4.4) par rapport à φ_k^+ et φ_k^- . Pour tout $k \in \Omega$, soit l l'indice tel que $\lfloor l \rfloor = k$ et u l'indice tel que $\lceil u \rceil = k$, alors nous avons :

$$\frac{\delta \check{\mathcal{C}}_{\nu_\varphi}(x)}{\delta \varphi_k^+} = \max_{i=1}^l x_{\lfloor i \rfloor} - \max_{i=1}^{l-1} x_{\lfloor i \rfloor}$$

puis

$$\frac{\delta \check{\mathcal{C}}_{\nu_\varphi}(x)}{\delta \varphi_k^-} = \min_{i=1}^u x_{\lceil i \rceil} - \min_{i=1}^{u-1} x_{\lceil i \rceil}.$$

Nous avons également

$$\frac{\delta\check{\mathbb{C}}_{\nu\varphi}(x)}{\delta\varphi_k^+} = \min_{i=1}^l x_{[i]} - \min_{i=1}^{l-1} x_{[i]}$$

puis

$$\frac{\delta\check{\mathbb{C}}_{\nu\varphi^c}(x)}{\delta\varphi_k^-} = \max_{i=1}^u x_{[i]} - \max_{i=1}^{u-1} x_{[i]}.$$

Ce qui nous donne :

$$\frac{\delta\check{\mathbb{C}}_{\nu\varphi}(x)}{\delta\varphi_k} = \left(\min_{i=1}^l x_{[i]} - \min_{i=1}^{l-1} x_{[i]} \right) + \left(\min_{i=1}^u x_{[i]} - \min_{i=1}^{u-1} x_{[i]} \right),$$

et

$$\frac{\delta\check{\mathbb{C}}_{\nu\varphi^c}(x)}{\delta\varphi_k} = \left(\min_{i=1}^l x_{[i]} - \min_{i=1}^{l-1} x_{[i]} \right) + \left(\max_{i=1}^u x_{[i]} - \max_{i=1}^{u-1} x_{[i]} \right).$$

avec $\min_{i=1}^0 x_{[i]} = 0 = \max_{i=1}^0 x_{[i]}$.

□

Cette dérivation est importante pour la mise en œuvre de la minimisation de la fonction de coût. Soit $\underline{y} = \check{\mathbb{C}}_{\nu\varphi}(x)$ et $\bar{y} = \mathbb{C}_{\nu\varphi}(x)$, posons pour tout $k \in \Omega$, $\underline{\delta y}_k = \frac{\partial \check{\mathbb{C}}_{\nu\varphi}(x)}{\partial \varphi_k}$ et $\bar{\delta y}_k = \frac{\partial \mathbb{C}_{\nu\varphi}(x)}{\partial \varphi_k}$. Le calcul des dérivées partielles par rapport aux paramètres de $\check{\mathbb{C}}_{\nu\varphi}(x)$ et de $\mathbb{C}_{\nu\varphi}(x)$ peut être facilement réalisé en utilisant l'Algorithme 3.

Algorithm 3: Calcul de $\underline{\delta y}, \overline{\delta y}$

Input: $x = \{x_i\}_{i=1\dots N}, \varphi = \{\varphi_i\}_{i=1\dots N}$

Output: $\underline{\delta y}, \overline{\delta y}$

soit $\iota = \{1, \dots, N\}$;

trie de (φ, ι) par rapport à φ dans l'ordre croissant ;

$\underline{x}_1 = x_{\iota_1}, \overline{x}_1 = x_{\iota_1}$;

for $k = 2 \dots N$ **do**

$\underline{x}_k = \min(\underline{x}_{k-1}, x_{\iota_k})$;
 $\overline{x}_k = \max(\overline{x}_{k-1}, x_{\iota_k})$;

for $k = 1 \dots N$ **do**

$\underline{\delta y}_{\iota_k} = \underline{x}_k$;
 $\overline{\delta y}_{\iota_k} = \overline{x}_k$;

for $k = 2 \dots N$ **do**

$\underline{\delta y}_{\iota_k} = \underline{\delta y}_{\iota_k} - \underline{x}_{k-1}$;
 $\overline{\delta y}_{\iota_k} = \overline{\delta y}_{\iota_k} - \overline{x}_{k-1}$;

$\underline{\delta y}_{\iota_1} = \underline{\delta y}_{\iota_1} - \overline{x}_N$;

$\overline{\delta y}_{\iota_1} = \overline{\delta y}_{\iota_1} - \underline{x}_N$;

En inversant les éléments de ι ;

$\underline{x}_1 = x_{\iota_1}, \overline{x}_1 = x_{\iota_1}$;

for $k = 2 \dots N$ **do**

$\underline{x}_k = \min(\underline{x}_{k-1}, x_{\iota_k})$;
 $\overline{x}_k = \max(\overline{x}_{k-1}, x_{\iota_k})$;

for $k = 1 \dots N$ **do**

$\underline{\delta y}_{\iota_k} = \underline{\delta y}_{\iota_k} + \underline{x}_k$;
 $\overline{\delta y}_{\iota_k} = \overline{\delta y}_{\iota_k} + \overline{x}_k$;

for $k = 2 \dots N$ **do**

$\underline{\delta y}_{\iota_k} = \underline{\delta y}_{\iota_k} - \underline{x}_{k-1}$;
 $\overline{\delta y}_{\iota_k} = \overline{\delta y}_{\iota_k} - \overline{x}_{k-1}$;

3 Apprentissage de l'agrégation macsum

Dans cette section, nous introduisons la méthode de descente de gradient puis nous présentons une fonction de coût conçue pour quantifier l'écart entre une prédiction imprécise et une étiquette précise.

3.1 Descente de gradient

La descente de gradient est une méthode d'optimisation largement utilisée en apprentissage automatique pour ajuster les paramètres d'un modèle [Goodfellow, 2016]. Elle repose sur le concept du gradient, qui, en mathématiques, représente la direction et la magnitude de la variation la plus rapide d'une fonction. Autrement dit, le gradient pointe vers la direction où la fonction augmente le plus rapidement. Le gradient d'une fonction par rapport à ses paramètres est un vecteur composé des dérivées partielles de cette fonction par rapport à chacun de ses paramètres. Chaque dérivée partielle mesure la sensibilité ou le taux de variation de la fonction lorsqu'on effectue une petite perturbation sur un paramètre spécifique, tout en maintenant les autres paramètres constants.

En contexte d'optimisation d'un modèle d'apprentissage, l'idée principale est de minimiser une fonction de coût en déplaçant progressivement les paramètres du modèle dans la direction opposée au gradient de cette fonction de coût.

Plus formellement, pour une fonction de coût $L(\varphi)$ où $\varphi = (\varphi_1, \varphi_2, \dots, \varphi_p)$ est le vecteur des paramètres de notre modèle, la mise à jour des paramètres à chaque itération de l'algorithme de descente de gradient est donnée par :

$$\varphi^{(t+1)} = \varphi^{(t)} - \eta \nabla L(\varphi^{(t)}),$$

où η est le taux d'apprentissage, un paramètre qui contrôle la taille des pas faits dans la direction opposée du gradient, $\nabla L(\varphi^{(t)})$ est le gradient de la fonction de coût L au point $\varphi^{(t)}$ et t indique l'itération courante de l'algorithme. En d'autres termes, à chaque itération, nous ajustons chaque paramètre φ_i par une quantité proportionnelle à la dérivée partielle de L par rapport à φ_i .

La descente de gradient est un algorithme itératif qui nécessite plusieurs passages sur les données pour converger vers une solution. La vitesse de convergence dépend du taux d'apprentissage et de la forme de la fonction de coût. Un bon taux d'apprentissage est crucial pour garantir une convergence rapide et éviter les oscillations ou la divergence [Hastie, 2001].

Dans la prochaine section, nous introduisons une fonction de coût qui nous permet de quantifier la différence entre la sortie de l'agrégation macsum, qui est un intervalle, et une sortie précise.

3.2 Fonction de coût pour une sortie intervallistes

Dans le contexte de l'apprentissage supervisé, nous avons à notre disposition un ensemble d'exemples d'apprentissage, constitué d'entrées et de sorties correspondantes, appelées étiquettes.

L'objectif de l'apprentissage supervisé est d'ajuster, à partir de ces exemples, les paramètres d'un modèle de manière à ce que ses sorties se rapprochent autant que possible des étiquettes correspondantes.

Pour quantifier la fiabilité du modèle, nous avons besoin d'une fonction qui compare la sortie prédite par le modèle à l'étiquette produite par le système. Cette fonction est généralement une fonction de coût et constitue le critère à minimiser lors de l'apprentissage. Une fonction de coût définit l'objectif à atteindre lors de l'ajustement des paramètres du modèle.

L'utilisation de la descente de gradient, présentée précédemment, permet de minimiser cette fonction de coût pour obtenir les paramètres optimaux d'ajustement du modèle.

Plus formellement, l'objectif de l'apprentissage d'une agrégation macsum est de trouver un paramètre $\hat{\varphi} \in \mathbb{R}^n$ qui rend la valeur $\mathcal{A}_{\nu_{\hat{\varphi}}}(x^j)$ aussi proche que possible de y^j pour tout $j \in \{1, \dots, M\}$, où x^j et y^j représentent respectivement l'entrée et la sortie du j -ème exemple d'un ensemble de données. Comme $\mathcal{A}_{\nu_{\hat{\varphi}}}(x^j)$ est un intervalle, la question à laquelle il faut répondre est *que signifie pour un intervalle réel d'être proche d'une valeur réelle ?* Différentes extensions des distances ont été proposées pour répondre à cette question, que l'on peut trouver dans [Saulnier,]. La plupart de ces solutions conduisent à des distances qui ne sont pas soit pertinentes pour ce problème soit trop non-linéaires pour être utilisées dans un processus d'apprentissage itératif.

Ce que nous proposons ici est une solution simple, évidente et facile à utiliser pour l'apprentissage. L'intervalle de sortie de l'agrégation macsum $[\underline{y}^j, \bar{y}^j] = \mathcal{A}_{\nu_{\hat{\varphi}}}(x^j)$ peut être considéré comme un ensemble convexe de sorties d'agrégations linéaires. Rendre cet ensemble aussi proche que possible de y^j peut être envisagé comme rendre les deux bornes \underline{y}^j et \bar{y}^j aussi proches que possible de y^j . Cela conduit à l'extension de la distance quadratique que nous proposons :

$$\begin{aligned} L(\{y^j, [\underline{y}^j, \bar{y}^j]\}_{j=1\dots M}) &= \sum_{j=1}^M \|y^j - \underline{y}^j\|_2^2 + \sum_{j=1}^M \|y^j - \bar{y}^j\|_2^2, \\ &= \sum_{j=1}^M (y^j \cdot y^j - y^j \cdot \bar{y}^j - y^j \cdot \underline{y}^j + \bar{y}^j \cdot \bar{y}^j + \underline{y}^j \cdot \underline{y}^j). \end{aligned}$$

où $\|\cdot\|_2$ est la norme quadratique et \cdot est le produit scalaire.

L'apprentissage d'une agrégation macsum basée sur cette extension de la distance quadratique peut être écrit comme suit :

$$\hat{\varphi} = \underset{\varphi}{\operatorname{argmin}} L(\{y^j, \mathcal{A}_{\nu_{\varphi}}(x^j)\}_{j=1\dots M}). \quad (4.5)$$

L'approche que nous avons proposée ici est une extension de la distance quadratique, qui minimise la somme des carrés des distances entre chaque valeur de sortie y^j et les deux bornes de l'intervalle $\mathcal{A}_{\nu_{\varphi}}(x^j)$, \underline{y}^j et \bar{y}^j . Cette extension de la distance quadratique nous a permis de formuler notre problème d'apprentissage en problème d'optimisation d'une fonction de coût.

Il convient de noter que, même si cette approche est assez simple et intuitive, elle n'est qu'une

possibilité parmi d'autres pour définir une fonction de coût dans le contexte de l'apprentissage d'agrégations macsum. D'autres extensions de cette distance ou bien d'autres distances pourraient être envisagées, selon les spécificités du problème à résoudre.

3.3 Expérimentation

Dans cette section, notre but est d'utiliser l'agrégation macsum comme modèle d'apprentissage automatique dans un cas d'usage concret.

La plupart des techniques de traitement d'images supposent que la réponse impulsionnelle (PSF) de chaque capteur composant la rétine d'une caméra (également appelé pixel physique) est invariante par translation spatiale. En d'autres termes, chaque élément de la rétine est supposé mesurer, de la même manière, la quantité de lumière qui l'irradie. Cependant, cette invariance par translation est généralement seulement approximative. Par construction, la PSF de chaque pixel peut présenter des fluctuations [Jiang, 2020], c'est-à-dire que la manière dont chaque élément de la rétine intègre l'irradiance peut varier. Cette variation est généralement imperceptible pour l'utilisateur, car l'étalonnage préalable de la rétine permet d'éviter la fluctuation des intensités mesurées. Cependant, cela peut avoir de graves conséquences dans les processus de déconvolution ou dans la reconstruction d'images (super-résolution, tomographie, ...)[Delbracio, 2013; Novak, 2021].

De plus, les lentilles causent souvent des effets de vignettage non intentionnels et indésirables qui se caractérisent par le fait que l'image est plus nette au centre que sur les bords. Un tel phénomène peut être caractérisé par une PSF de pixels dont la spécificité diminue avec la distance au centre de l'image. Cela est très visible en imagerie plénoptique ou omnidirectionnelle ainsi qu'en microscopie optique. La Figure (4.1) présente un exemple d'un tel phénomène.

Cette expérience visait à imiter un effet de vignettage. Nous avons utilisé 2000 images naturelles de dimension 600×600 fournies par le projet CLEF¹ (voir Figure (4.2)). Chacune des 2000 images a été sous-échantillonnée par un facteur de 7, c'est-à-dire qu'à chaque image de dimension 600×600 est associée une image de dimension 85×85 (voir Figure (4.5)). La valeur de chaque pixel de l'image de dimension 85×85 est obtenue en agrégeant un patch de 7×7 pixels de l'image de dimension 600×600 . Cette agrégation est réalisée par convolution avec un noyau de dimension 7×7 qui est censé être la PSF du pixel considéré (voir Figure (4.3)). La spécificité de la PSF associée à chaque pixel de l'image de dimension 85×85 diminue avec la distance du pixel au centre de l'image. La Figure (4.4) illustre les deux valeurs extrêmes du noyau de sous-échantillonnage : Figure (4.4).a la PSF utilisée pour les valeurs centrales, Figure (4.4).b la PSF utilisée pour les valeurs de bord.

Pour être en accord avec la théorie que nous préconisons, chaque PSF est signée (c'est-à-dire qu'elle comporte des valeurs positives et négatives) et intègre à 1.9.

1. <https://www.imageclef.org/>

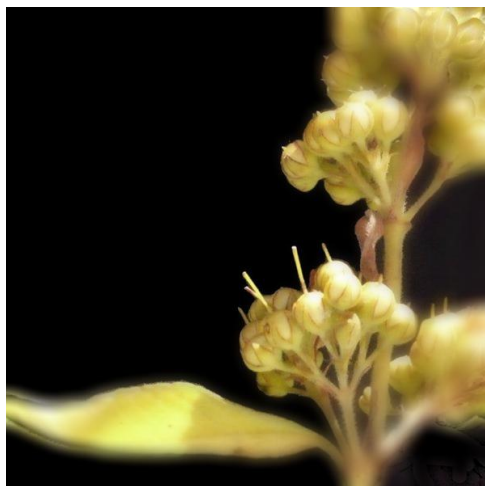


FIGURE 4.1 – Image acquise par un capteur non invariant par décalage.



FIGURE 4.2 – Quatre des 2000 images utilisées pour cette expérience.

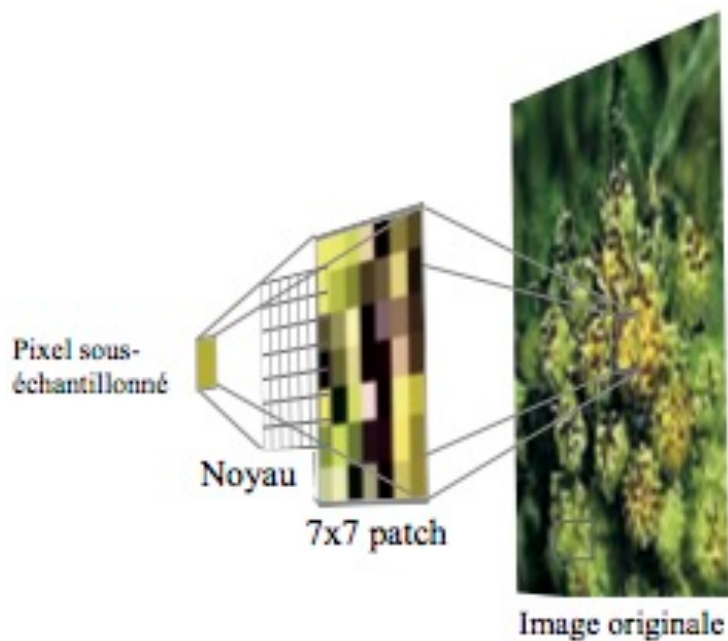


FIGURE 4.3 – Sous-échantillonnage de la 3ème image de la Figure (4.2)

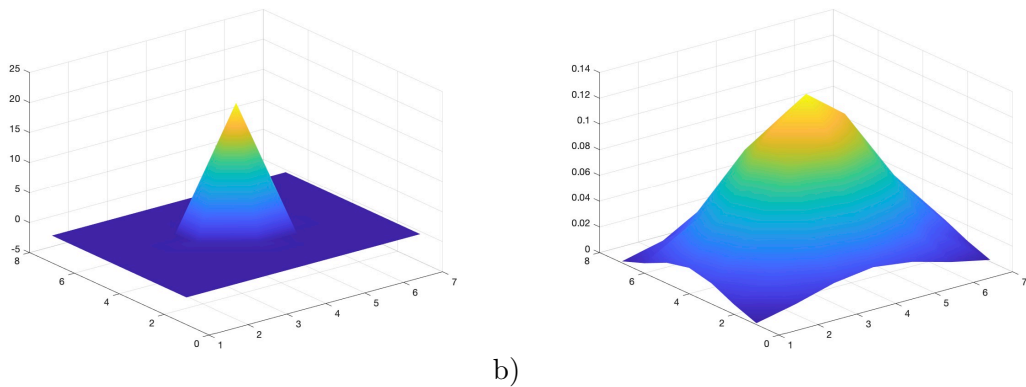


FIGURE 4.4 – PSF utilisée pour sous-échantillonner les images : centrale (a) et bordure (b).



FIGURE 4.5 – Images illustrées dans la Figure (4.2) après sous-échantillonnage.

Dans cette expérience, notre objectif est d'apprendre le noyau de convolution associé au sous-échantillonnage en utilisant à la fois un modèle d'agrégation additive et un modèle d'agrégation macsum. Le sous-échantillonnage avec le modèle additif consiste à calculer la moyenne de chaque bloc de l'image originale, pondérée par un noyau de convolution ψ (que nous appellerons ici sous-échantillonnage précis). Le sous-échantillonnage avec le modèle macsum consiste à calculer l'agrégation à valeurs d'intervalle de chaque bloc de l'image originale, pondérée par un noyau de convolution φ (que nous appellerons ici sous-échantillonnage imprécis).

Nous utilisons 30 images choisies au hasard pour apprendre les noyaux et les 1970 images restantes pour caractériser le résultat. L'idée est d'évaluer comment ce modèle imprécis est capable de prédire la proximité du modèle à la réalité et comment le processus d'apprentissage est capable de trouver un noyau qui, associé à l'opérateur macsum, peut représenter l'ensemble des noyaux qui ont été utilisés pour sous-échantillonner les images originales.

Nous avons arbitrairement effectué 1000 itérations de l'algorithme d'apprentissage pour les deux modèles (additif et macsum). Soit $\hat{\psi}$ le noyau additif et $\hat{\varphi}$ le noyau macsum obtenus par les algorithmes d'apprentissage (voir Figure (4.6)). Soit $\{I_k\}_{k=1\dots 1000}$ l'ensemble des 1000 images originales de dimension 600×600 , $\{F_k\}_{k=1\dots 1000}$ l'ensemble des 1000 images de dimension 85×85 sous-échantillonnées, $\{\hat{F}_k\}_{k=1\dots 1000}$ l'ensemble des images obtenues en sous-échantillonnant les I_k avec $\hat{\psi}$ et $\{[\underline{F}_k, \overline{F}_k]\}_{k=1\dots 1000}$ les images à valeurs d'intervalle obtenues en utilisant le sous-échantillonnage macsum avec le noyau $\hat{\varphi}$.

Pour évaluer à quel point la $k^{\text{ème}}$ image sous-échantillonnée précise prédite \hat{F}_k est proche de la *vraie* image sous-échantillonnée F_k , nous utilisons la distance moyenne L_1 définie par :

$$L_1(\hat{F}_k, F_k) = \frac{1}{7225} \sum_{i=1}^{85} \sum_{j=1}^{85} |\hat{F}_k(i, j) - F_k(i, j)|,$$

Pour évaluer à quel point la $k^{\text{ème}}$ image sous-échantillonnée imprécise prédite $[\underline{F}_k, \overline{F}_k]$ est proche de la *vraie* image sous-échantillonnée F_k , nous utilisons chacune de ces extensions de la distance L_1 suivantes :

- L'extension de Hausdorff est définie par : $L_1^H([\underline{F}_k, \overline{F}_k], F_k) = \sup_{G \in [\underline{F}_k, \overline{F}_k]} L_1(G, F_k)$,
- l'extension naturelle est définie par : $L_1^N([\underline{F}_k, \overline{F}_k], F_k) = \inf_{G \in [\underline{F}_k, \overline{F}_k]} L_1(G, F_k)$,
- la distance moyenne au centre est définie par : $L_1^C([\underline{F}_k, \overline{F}_k], F_k) = L_1(\frac{1}{2}(\underline{F}_k + \overline{F}_k), F_k)$,
- et l'extension proposée par Saulnier dans [Saulnier,] notée $L_1^S([\underline{F}_k, \overline{F}_k], F_k)$.

L'extension de Saulnier peut être exprimée par :

$$L_1^S([\underline{F}_k, \overline{F}_k], F_k) = \frac{1}{7225^2} \sum_{i=1}^{85} \sum_{j=1}^{85} \sum_{\check{i}=1}^{85} \sum_{\check{j}=1}^{85} |\tilde{F}_k(i, j) - F_k(i, j) - \Delta F_k(i, j) + \Delta F_k(\check{i}, \check{j})|,$$

$$\text{avec } \tilde{F}_k(i, j) = \frac{1}{2} \cdot (\overline{F}_k(i, j) + \underline{F}_k(i, j)) \text{ et } \Delta F_k(i, j) = \frac{1}{2} \cdot (\overline{F}_k(i, j) - \underline{F}_k(i, j)).$$

Ces distances et extensions peuvent être utilisées pour caractériser la convergence des algorithmes d'apprentissage. Comme on peut le voir sur la Figure (4.7), l'expérience montre que la convergence de la fonction de coût semble très rapide (moins de 30 itérations pour l'apprentissage additif et macsum).

On peut également considérer la convergence de l'apprentissage macsum à travers les distances étendues – extensions de Hausdorff, Saulnier et naturelle – (voir Figure (4.8)). Dans ce cas, la convergence est généralement obtenue après environ 150 itérations.

La rapidité de convergence des algorithmes dépend naturellement des images utilisées pour l'apprentissage. Au cours de nos différentes expériences, elle n'a jamais dépassé 60 itérations.

Ce qui pourrait être particulièrement intéressant maintenant est d'évaluer la capacité de l'agrégation macsum apprise à prédire sa propre erreur de sortie.

Première question : *Est-ce que le noyau $\hat{\varphi}$ domine tous les noyaux utilisés pour sous-échantillonner l'image ?* Pour répondre à cette question, nous avons utilisé un processus de MonteCarlo qui consiste à sélectionner aléatoirement un noyau ψ , parmi ceux qui ont été utilisés pour le sous-échantillonnage, et un ensemble A et à tester si $P_\psi(A) \in [\nu_\psi^c(A), \nu_\psi(A)]$ ou non. Nous avons réalisé cette procédure 100000 fois et 100% des tests sont positifs. Ce taux est également de 100% si nous considérons le noyau additif appris $\hat{\psi}$ (c'est-à-dire tester si $P_{\hat{\psi}}(A) \in [\nu_{\hat{\psi}}^c(A), \nu_{\hat{\psi}}(A)]$ ou non) lors de la répétition des processus d'apprentissage additif et macsum 300 fois avec différents jeux d'images.

Deuxième question : *Les images sous-échantillonnées F_k sont-elles incluses dans les images imprécises $[\underline{F}_k, \overline{F}_k]$?* Pour répondre à cette question, nous avons compté le nombre de pixels (i, j) tels que $F_k(i, j) \in [\underline{F}_k(i, j), \overline{F}_k(i, j)]$. Nous obtenons un taux de 91% de pixels des images

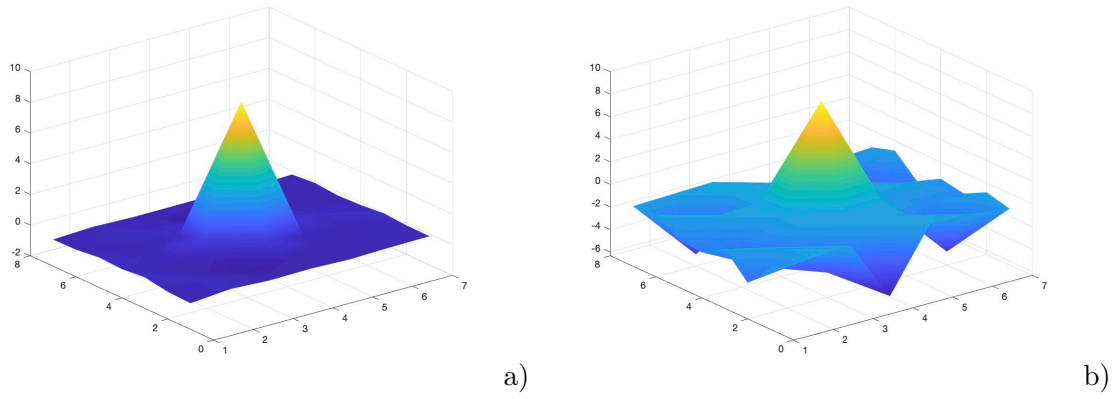


FIGURE 4.6 – Après l'apprentissage, le noyau $\hat{\psi}$ (a) et le noyau $\hat{\varphi}$ (b)

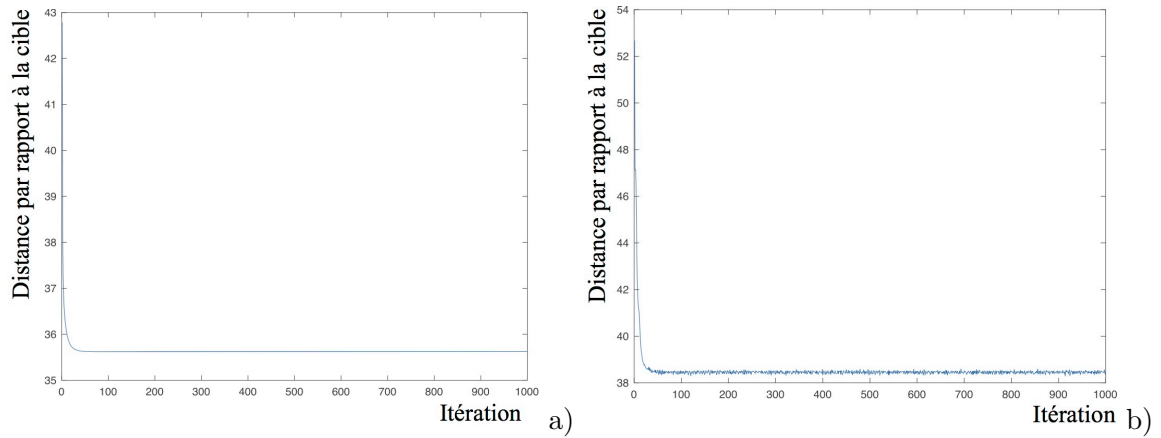


FIGURE 4.7 – Convergence de l'apprentissage additif (a) et de l'apprentissage macsum (b).

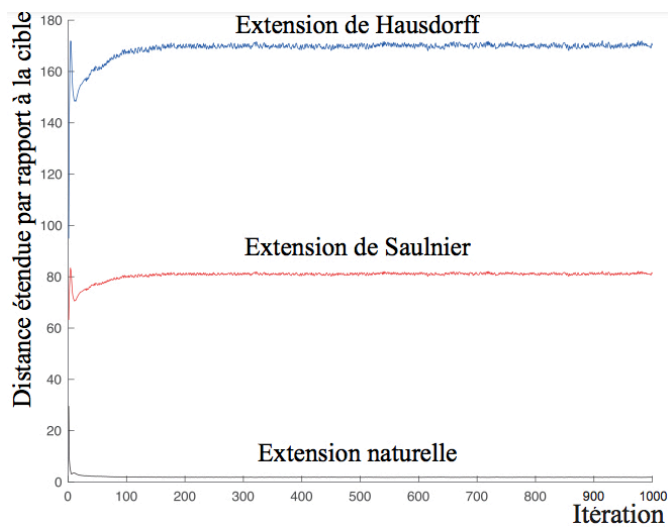


FIGURE 4.8 – Convergence de l'apprentissage macsum à travers des distances étendues.

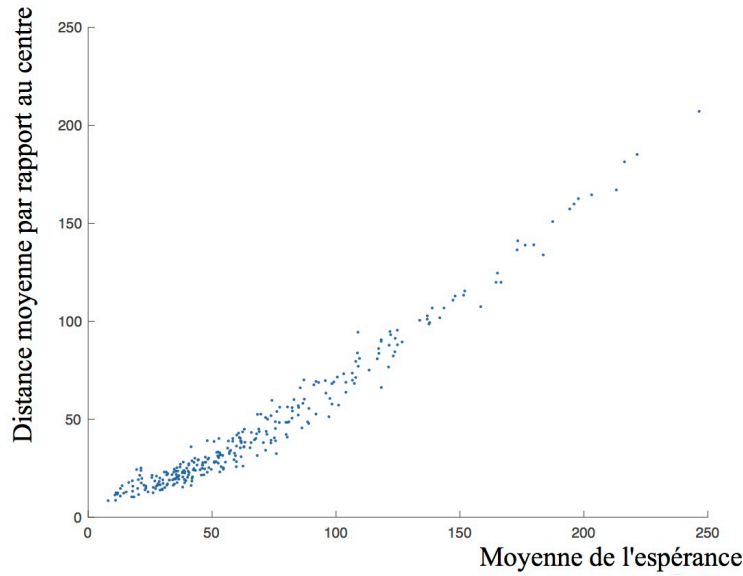


FIGURE 4.9 – Corrélation entre l'écart moyen et la distance moyenne au centre

sous-échantillonnées qui appartiennent aux valeurs imprécises prédites de l'image sous-échantillonnée correspondante.

Troisième question : *L'imprécision de l'image sous-échantillonnée prédite de manière imprécise est-elle un indicateur de la capacité de l'opérateur macsum associé au noyau appris $\hat{\varphi}$ à prédire correctement la sortie ?* Pour répondre à cette troisième question, nous avons calculé, pour chacune des 1970 images de test, la distance moyenne au centre : $\epsilon_k = L_1^C([\underline{F}k, \overline{F}k], F_k)$ et l'écart moyen de l'image à valeurs d'intervalle $\delta_k = \frac{1}{7225} \sum_{i=1}^{85} \sum_{j=1}^{85} \frac{1}{2}(\overline{F}_k(i, j) - \underline{F}_k(i, j))$. Comme on peut le voir sur la Figure (4.9), ces deux valeurs sont fortement corrélées. Le coefficient de corrélation de Pearson entre ϵ et δ est de 0.98. Le coefficient de corrélation de Pearson entre δ et toutes les autres distances varie entre 0.98 et 0.99. Ceci est également illustré sur la Figure (4.10) où l'on peut voir une grande ressemblance entre l'image de l'écart moyen Figure(4.10)(a) et l'image de la distance au centre Figure(4.10)(b).

Enfin, on peut être intéressé par la question de savoir à quel point les images réduites prédites sont proches des véritables images réduites. En considérant la réduction précise en utilisant $P_{\hat{\varphi}}$ comme opérateur d'agrégation, la distance moyenne L_1 calculée sur toutes les images de test est de 41.6. En considérant la réduction imprécise en utilisant $\nu_{\hat{\varphi}}$ comme opérateur d'agrégation, la valeur moyenne de L_1^H est de 183.7, la moyenne de L_1^S est de 77.4, la moyenne de L_1^C est de 46.2 et la moyenne de L_1^N est de 2.4. On remarque ici que l'extension naturelle fournit un meilleur résultat, cela souligne l'importance de l'étape de sélection de la fonction de coût dans le processus d'apprentissage.

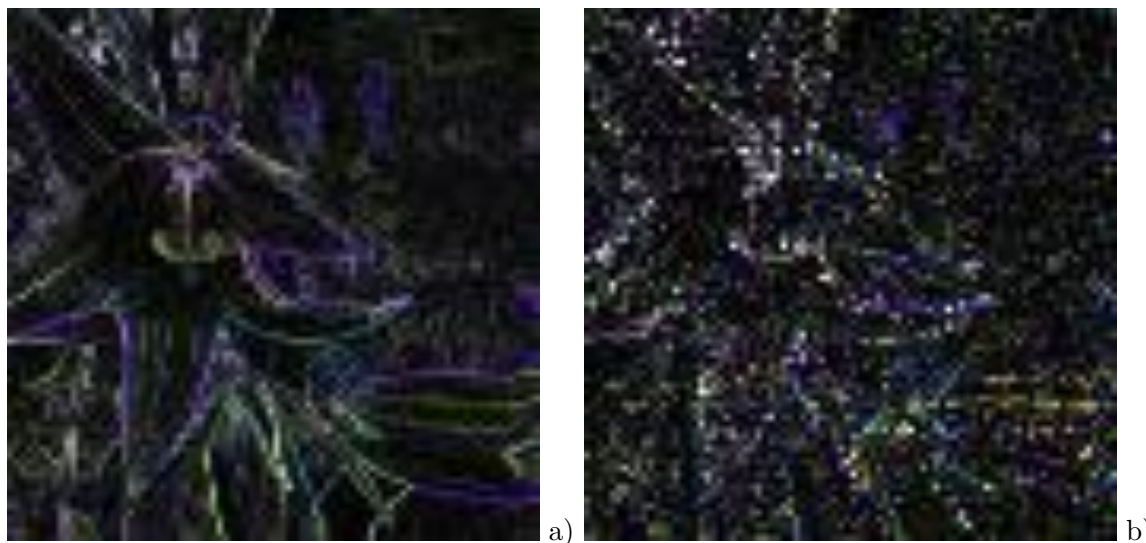


FIGURE 4.10 – Image de l'écart (a) et de la distance au centre (b) pour la 2^{ème} image de la Figure (4.5)

4 Conclusion

Dans ce chapitre, nous avons abordé l'ajustement des paramètres de l'agrégation macsum via l'apprentissage automatique. Nous avons démontré la non-vacuité du cœur macsumitif, ce qui garantit l'existence d'un ensemble d'agrégation linéaire représenté par l'agrégation macsum pour chaque paramètre appris. Par ailleurs, la dérivation de l'agrégation macsum par rapport à ses paramètres est nécessaire pour son apprentissage.

Toujours dans l'optique d'introduire un modèle intervalliste simple d'utilisation, nous avons reformulé le calcul des bornes de l'agrégation macsum de manière à ce qu'elles soient linéaires par rapport aux paramètres. De cette façon, l'application de la méthode de la descente du gradient est algorithmiquement moins coûteuse. Nous avons utilisé cette méthode dans un cas pratique où nous avons utilisé l'agrégation macsum comme modèle d'apprentissage intervalliste dans le cadre du traitement d'image. L'intérêt de l'agrégation macsum dans le domaine de l'apprentissage automatique semble établi. Cependant, malgré sa pertinence, son utilisation soulève plusieurs interrogations qui nécessitent des investigations plus poussées. En effet, l'agrégation macsum est, pour l'instant, limitée à des entrées précises; il serait intéressant de l'étendre à des entrées intervallistes. De plus, nous savons que son intervalle de sortie représente un ensemble convexe de sorties d'agrégation linéaires, mais nous n'avons pour l'instant pas défini de moyen pour quantifier la taille de cet ensemble.

Dans le prochain chapitre, nous établissons une extension de l'agrégation macsum aux entrées intervallistes. De plus, nous définissons un indice de cardinalité du noyau macsumitif, nous donnant une mesure de la quantité d'agrégations linéaires représentées par l'agrégation macsum.

Chapitre 5

Extension et granularité de l'agrégation macsum

Objectifs

L'essence de ce chapitre est d'explorer et d'élargir les horizons de l'agrégation macsum, en particulier dans le contexte des données imprécises. Cette imprécision, inhérente à nos observations et mesures, est souvent représentée sous forme d'intervalle, ce qui nous incite à chercher des moyens de traiter efficacement ces données. Ce chapitre se consacre principalement à deux volets majeurs. Le premier est l'extension de l'agrégation macsum aux entrées sous forme d'intervalle, élargissant ainsi le champ d'application de ce modèle. Le second vise à approfondir notre compréhension des propriétés de l'agrégation macsum, avec un accent particulier sur la détermination de la quantité d'agrégation linéaire qu'il représente.

Sommaire

1	Introduction	59
2	Extension de l'agrégation macsum aux entrées intervallistes	59
2.1	Préliminaires	59
2.2	Extension disjonctive aux entrées intervallistes	60
2.3	Extension conjonctive aux entrées intervallistes	64
2.4	Exemple : Prévission des températures dans une région à topographie variée	67
3	Granularité du coeur macsumitif d'un paramètre	68
4	Conclusion	73

1 Introduction

Nos connaissances, issues principalement d'observations ou de mesures, sont naturellement empreintes d'incertitude en raison des limites inhérentes à notre capacité de percevoir et de comprendre entièrement un phénomène. Cette incertitude peut conduire à une certaine imprécision dans nos données. Prenons un exemple simple : lorsqu'on essaie d'estimer l'âge d'une personne, donner une fourchette d'âge (comme entre 30 et 40 ans) offre plus de certitude qu'une estimation précise. De manière extrême, indiquer un âge compris entre 0 et 200 ans serait absolument certain, mais manquerait cruellement de précision. La clé réside donc dans l'équilibre entre certitude et précision. Affirmer par exemple que la température se situe entre 20 et 25 degrés est une déclaration à la fois suffisamment précise et dotée d'une certaine certitude. Aujourd'hui, à l'ère du numérique, des quantités phénoménales de données sont produites chaque jour, et bon nombre d'entre elles sont représentées sous forme d'intervalles. De nombreuses bases de données reflètent un équilibre judicieux entre imprécision et certitude et peuvent ainsi être très informative [Miller, 1997]. Cela souligne l'importance de développer des modèles aptes à traiter ce type de données. L'imprécision dans les données peut aussi bien être au niveau des entrées que des sorties relevant la pertinence de construire des modèles qui prennent des intervalles en entrée et génèrent un intervalle en sortie.

Dans ce chapitre, notre premier point de discussion porte sur l'extension de l'agrégation macsum aux entrées intervallistes [Hmidy, 2023a]. En élaborant une telle extension, nous espérons élargir le spectre des systèmes pour lesquels l'agrégation macsum constitue un modèle adéquat.

Ensuite, nous envisageons d'explorer davantage les propriétés de l'agrégation macsum. Nous déterminons un indice de cardinalité permettant de déterminer la quantité d'agrégation linéaire qu'il représente [Strauss, 2022a].

2 Extension de l'agrégation macsum aux entrées intervallistes

Jusqu'ici, l'agrégation macsum a été définie pour des entrées précises. Dans l'optique d'élargir la gamme de systèmes pouvant être modélisés par l'agrégation macsum, on se propose, dans cette section, d'étendre l'utilisation de l'agrégation macsum aux entrées intervallistes.

2.1 Préliminaires

Avant de se lancer dans l'extension de l'agrégation macsum aux entrées intervallistes, il est essentiel de se familiariser avec certaines notations et définitions qui sont utilisées tout au long de cette section.

Tout comme dans les chapitres précédents, notre ensemble de référence est $\Omega = \{1, \dots, n\}$.

Un intervalle réel est un ensemble de nombres réels compris entre une borne inférieure et une borne supérieure. Il est noté comme $\bar{x} = [x, \bar{x}]$. Rappelons que nous notons \mathbb{R} l'ensemble de tous les intervalles réels. En établissant une analogie avec les vecteurs, un vecteur d'intervalles réels est un élément de \mathbb{R}^n et est noté $\bar{x} = (\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n)$.

Nous introduisons la notion d'appartenance d'un vecteur de réels à un vecteur d'intervalles réels. Soient $x \in \mathbb{R}^n$ et $\bar{x} \in \mathbb{IR}$, on dit que x appartient à \bar{x} , noté $x \in \bar{x}$, si chaque élément x_i du vecteur x appartient à l'intervalle correspondant \bar{x}_i pour tout $i \in \Omega$.

Dans nos travaux, nous sommes amenés à considérer des opérations de conjonction et disjonction. Concernant les intervalles, ces opérations peuvent faire l'objet de deux approches : l'approche naturelle et l'approche extensive. Soient \bar{x}, \bar{y} deux intervalles de \mathbb{IR} , l'union naturelle entre \bar{x} et \bar{y} est donnée par :

$$\bar{x} \cup \bar{y} = \{z \in \mathbb{R} \mid z \in \bar{x} \text{ ou } z \in \bar{y}\}$$

et l'intersection naturelle par :

$$\bar{x} \cap \bar{y} = \{z \in \mathbb{R} \mid z \in \bar{x} \text{ et } z \in \bar{y}\}.$$

L'union extensive entre \bar{x} et \bar{y} est donnée par :

$$\bar{x} \tilde{\cup} \bar{y} = [\min(\underline{x}, \underline{y}), \max(\bar{x}, \bar{y})]$$

Il est à noter que l'union naturelle et l'union extensive de deux intervalles \bar{x} et \bar{y} sont égales si $\bar{x} \cap \bar{y} \neq \emptyset$. De plus, $\bar{x} \tilde{\cup} \bar{y}$ est l'enveloppe convexe de $\bar{x} \cup \bar{y}$. Une enveloppe convexe d'un ensemble de points est le plus petit ensemble convexe qui contient tous les points de cet ensemble. Deux intervalles sont dit duaux l'un de l'autre, si l'un est nécessairement vide quand l'autre ne l'est pas et vice versa.

2.2 Extension disjonctive aux entrées intervallistes

Le terme "disjonctif" puise ses origines du latin "disiunctivus", issu de "disiungere", signifiant "séparer" ou "dissocier". Dans ce contexte, le préfixe "dis-" évoque la séparation, tandis que "iungere" renvoie à l'idée de "joindre". De ce fait, "disjonctif" symbolise l'idée de dissocier des éléments d'une entité globale. À l'opposé, l'agrégation vise à assembler. Lorsque nous parlons d'agrégation disjonctive, nous faisons référence à un rassemblement qui n'omet aucun élément, rassemblant intégralement les constituants d'une entité. Dans un contexte logique ou mathématique, la disjonction est une opération qui prend le statut de vraie si, et seulement si, au moins l'un de ses opérands est vrai. Cette notion de disjonction trouve un écho dans les opérations d'union d'ensembles, où tous les éléments des ensembles combinés sont pris en compte sans répétition. De même, l'agrégation disjonctive aborde chaque valeur d'un intervalle comme une entité valide, en reconnaissant et en incorporant chaque élément sans discrimination, tout comme l'union englobe toutes les singularités des ensembles qu'elle combine.

Lorsque l'on parle d'étendre une agrégation à des entrées intervallistes en adoptant une approche disjonctive, on se réfère à la démarche visant à traiter l'ensemble des informations contenues dans chaque intervalle sans en omettre aucune. Au lieu de vouloir parvenir à une représentation synthétique, l'approche disjonctive s'attache à reconnaître et à valoriser la totalité

de l'information présente dans chaque intervalle.

Cette union offre une vision exhaustive des résultats possibles de l'agrégation, garantissant que l'intégrité de l'information initiale demeure intacte et que l'ensemble des scénarios envisageables est pleinement considéré. Ainsi, nous considérons dans cette partie une agrégation prudente, dite disjonctive, qui conserve toutes les sorties intervallistes des agrégations macsum pour chaque $x \in \bar{x}$.

Contrairement à l'agrégation macsum, l'agrégation linéaire est à valeur précise. Elle peut être étendue aux entrées intervallistes $\bar{x} \in \mathbb{R}$ par :

$$\begin{aligned} \mathcal{A}_{P_\psi}(\bar{x}) &= \{\check{C}_{P_\psi}(x) / x \in \bar{x}\} \\ &= \left[\inf_{x \in \bar{x}} \check{C}_{P_\psi}(x), \sup_{x \in \bar{x}} \check{C}_{P_\psi}(x) \right], \end{aligned} \quad (5.1)$$

Cet ensemble est convexe, car \bar{x} est convexe et \mathcal{A}_{P_ψ} est linéaire [Borwein, 2006].

Rappelons que l'agrégation macsum représente un ensemble convexe d'agrégations linéaires. Cette propriété nous permet d'exploiter la facilité d'étendre les agrégations linéaires aux entrées intervallistes afin d'étendre également l'agrégation macsum. Cette extension peut être définie par $\forall \bar{x} \in \mathbb{R}^n$:

$$\begin{aligned} \mathcal{D}_{\nu_\varphi}(\bar{x}) &= \bigcup_{x \in \bar{x}} \mathcal{A}_{\nu_\varphi}(x) \\ &= \bigcup_{x \in \bar{x}} [\check{C}_{\nu_\varphi^c}(x), \check{C}_{\nu_\varphi}(x)] \\ &= \{\mathcal{A}_{P_\psi}(x) / x \in \bar{x}, \psi \in \mathcal{M}(\varphi)\}. \end{aligned} \quad (5.2)$$

Remarque. Étant donné que l'intégrale de Choquet par rapport à l'opérateur macsum est continue comme démontré dans [Grabisch, 2000] (Proposition 5.39) et que \bar{x} est un sous-espace compact de \mathbb{R}^n , nous avons que $\bigcup_{x \in \bar{x}} [\check{C}_{\nu_\varphi^c}(x), \check{C}_{\nu_\varphi}(x)] = \tilde{\bigcup}_{x \in \bar{x}} [\check{C}_{\nu_\varphi^c}(x), \check{C}_{\nu_\varphi}(x)]$.

Ainsi, l'équation (5.2) peut être réécrite pour tout $\bar{x} \in \mathbb{R}^n$:

$$\mathcal{D}_{\nu_\varphi}(\bar{x}) = \tilde{\bigcup}_{x \in \bar{x}} [\check{C}_{\nu_\varphi^c}(x), \check{C}_{\nu_\varphi}(x)] = \left[\inf_{x \in \bar{x}} \check{C}_{\nu_\varphi^c}(x), \sup_{x \in \bar{x}} \check{C}_{\nu_\varphi}(x) \right]. \quad (5.3)$$

La convexité de $\mathcal{D}_{\nu_\varphi}(\bar{x})$ implique que pour tout couple de points dans l'ensemble, l'intervalle qui les relie est également contenu dans l'ensemble.

Les propositions 2.1 et 2.2 énoncées ci-dessous sont d'une importance fondamentale pour le calcul de l'extension disjonctive. Elles fournissent un cadre formel pour la relation entre les vecteurs φ et ψ .

Proposition 2.1. *Soit $\varphi \in \mathbb{R}^n$, soit $\psi \in \mathcal{M}(\varphi)$, $\forall i \in \Omega$, $\varphi_i > 0 \Rightarrow \psi_i \geq 0$ et $\varphi_i < 0 \Rightarrow \psi_i \leq 0$.*

Preuve. Soit $i \in \Omega$. Par construction $\nu_\varphi^c(\{i\}) \leq P_\psi(\{i\}) = \psi_i \leq \nu_\varphi(\{i\})$.

— Si $\varphi_i > 0$, alors $\nu_\varphi^c(\{i\}) = \min_{k \in \{i\}} \varphi_k^- + \underline{\alpha} - \max_{k \in \{i\}} \varphi_k^+ = \underline{\alpha} - \max_{k \in \{i\}} \varphi_k^+$.

Ainsi $0 \leq \nu_\varphi^c(\{i\}) \leq \psi_i$.

— Si $\varphi_i < 0$, alors $\nu_\varphi(\{i\}) = \max_{k \in \{i\}} \varphi_k^+ + \underline{\alpha} - \min_{k \in \{i\}} \varphi_k^- = \underline{\alpha} - \min_{k \in \{i\}} \varphi_k^-$.

Ainsi $\psi_i \leq \nu_\varphi(\{i\}) \leq 0$.

□

Proposition 2.2. Soit $\varphi \in \mathbb{R}^n$, soit $\psi \in \mathcal{M}(\varphi)$, $\forall i \in \Omega$, $\varphi_i = 0 \Rightarrow \psi_i = 0$.

Preuve. Soit $j \in \Omega$ tel que $\varphi_j = 0$. On a $\nu_\varphi(\{j\}) = 0 + \min_{i \in \Omega} \varphi_i^- - \min_{i \in \Omega \setminus j} \varphi_i^-$.

Comme $\varphi_j = 0$, soit $\min_{i \in \Omega} \varphi_i^- < 0$ et alors $\min_{i \in \Omega} \varphi_i^- = \min_{i \in \Omega \setminus j} \varphi_i^-$,

ce qui donne $\nu_\varphi(\{j\}) = 0$.

Soit $\min_{i \in \Omega} \varphi_i^- = \varphi_j = \min_{i \in \Omega \setminus j} \varphi_i^- = 0$, ce qui donne aussi $\nu_\varphi(\{j\}) = 0$.

De plus, on a $\nu_\varphi^c(\{j\}) = 0 + \max_{i \in \Omega} \varphi_i^+ - \max_{i \in \Omega \setminus j} \varphi_i^+$.

Prouver que $\max_{i \in \Omega} \varphi_i^+ = \max_{i \in \Omega \setminus j} \varphi_i^+$ et donc que $\nu_\varphi^c(\{j\}) = 0$ peut être fait de la même manière.

Comme $\psi \in \mathcal{M}(\varphi)$, on a $\forall A \subseteq \Omega$, $\nu_\varphi^c(A) \leq P_\psi(A) \leq \nu_\varphi(A)$,

Donc $\nu_\varphi^c(\{j\}) \leq P_\psi(\{j\}) \leq \nu_\varphi(\{j\})$,

Ainsi $0 \leq \psi_j \leq 0$ i.e. $\psi_j = 0$.

□

En conséquence, si un élément d'un paramètre est non-négatif (ou respectivement non-positif), alors l'élément correspondant dans n'importe quel paramètre de son coeur sera également non-négatif (ou respectivement non-positif).

La domination de l'opérateur macsum sur les opérateurs additifs développée dans le chapitre précédent nous donne que pour $\varphi \in \mathbb{R}^n$, $\forall \psi \in \mathcal{M}(\varphi)$, $\forall x \in \mathbb{R}^n$ on a $\mathcal{A}_{P_\psi}(x) \in \mathcal{A}_{\nu_\varphi}(x)$ c'est à dire $\check{\mathcal{C}}_{P_\psi}(x) \in [\check{\mathcal{C}}_{\nu_\varphi^c}(x), \check{\mathcal{C}}_{\nu_\varphi}(x)]$.

Nous pouvons reformuler les bornes de l'intervalle de sortie de l'agrégation macsum en utilisant cette propriété de domination d'un ensemble d'agrégations linéaires.

Premièrement, pour tout $x \in \mathbb{R}$, $\check{\mathcal{C}}_{\nu_\varphi^c}(x)$, est définie comme l'infimum sur l'ensemble ψ appartenant à $\mathcal{M}(\varphi)$. En termes simples, cela signifie que pour un x donné, nous considérons toutes les valeurs possibles de $\check{\mathcal{C}}_{P_\psi}(x)$ où ψ est un élément de $\mathcal{M}(\varphi)$, et nous prenons la plus petite de ces valeurs. Formellement, c'est écrit comme $\check{\mathcal{C}}_{\nu_\varphi^c}(x) = \inf_{\psi \in \mathcal{M}(\varphi)} \check{\mathcal{C}}_{P_\psi}(x)$.

Deuxièmement, pour tout $x \in \mathbb{R}$, $\check{\mathcal{C}}_{\nu_\varphi}(x)$ est défini de manière similaire, mais cette fois nous prenons le supremum sur l'ensemble ψ appartenant à $\mathcal{M}(\varphi)$. Autrement dit, nous considérons toutes les valeurs possibles de $\check{\mathcal{C}}_{P_\psi}(x)$ et nous prenons la plus grande de ces valeurs. C'est écrit comme $\check{\mathcal{C}}_{\nu_\varphi}(x) = \sup_{\psi \in \mathcal{M}(\varphi)} \check{\mathcal{C}}_{P_\psi}(x)$.

En utilisant ces formulations, nous pouvons réécrire l'équation (5.3), de la manière suivante :

$$\mathcal{D}_{\nu\varphi}(\bar{x}) = \left[\inf_{x \in \bar{x}} \inf_{\psi \in \mathcal{M}(\varphi)} \check{C}_{P_\psi}(x), \sup_{x \in \bar{x}} \sup_{\psi \in \mathcal{M}(\varphi)} \check{C}_{P_\psi}(x) \right] \quad (5.4)$$

$$= \left[\inf_{\psi \in \mathcal{M}(\varphi)} \inf_{x \in \bar{x}} \check{C}_{P_\psi}(x), \sup_{\psi \in \mathcal{M}(\varphi)} \sup_{x \in \bar{x}} \check{C}_{P_\psi}(x) \right]. \quad (5.5)$$

Ici, $\mathcal{D}_{\nu\varphi}(\bar{x})$ est formulé en termes de la valeur infimum et supremum de $\check{C}_{P_\psi}(x)$ pour tous x dans le vecteur d'intervalle \bar{x} et pour tous ψ dans l'ensemble $\mathcal{M}(\varphi)$.

Dans ce contexte, nous allons introduire une notation qui va nous simplifier l'écriture de cette expression.

Soit $\bar{x} \in \mathbb{R}^n$ un vecteur d'intervalles réels et soit $\varphi \in \mathbb{R}^n$.

Nous définissons $x^* \in \mathbb{R}^n$ comme étant le vecteur tel que, pour tout $i \in \Omega$, $x_i^* = \bar{x}_i$ si $\varphi_i \geq 0$ et $x_i^* = \underline{x}_i$ si $\varphi_i < 0$.

De même, nous définissons $x_* \in \mathbb{R}^n$ comme le vecteur tel que, pour tout $i \in \Omega$, $x_{*i} = \underline{x}_i$ si $\varphi_i \geq 0$ et $x_{*i} = \bar{x}_i$ si $\varphi_i < 0$.

Dans cette définition, nous introduisons deux vecteurs, x^* et x_* , définis en fonction d'un vecteur d'intervalles réels \bar{x} et d'un vecteur de paramètres réels φ .

Chaque élément de ces vecteurs, x_i^* et x_{*i} , est défini en fonction de l'élément correspondant φ_i .

Si φ_i est supérieur ou égal à 0, alors x_i^* prend la valeur de la borne supérieure de l'intervalle correspondant à \bar{x}_i et x_{*i} prend la valeur de la borne inférieure.

Si φ_i est inférieur à 0, alors x_i^* prend la valeur de la borne inférieure de l'intervalle de l'intervalle correspondant à \bar{x}_i et x_{*i} prend la valeur de la borne supérieure.

Poursuivant sur la même lancée, nous introduisons maintenant une proposition qui découle naturellement des résultats que nous avons établis jusqu'à présent.

Proposition 2.3. *Pour tout $\bar{x} \in \mathbb{R}^n$, tout $\varphi \in \mathbb{R}^n$, et tout $\psi \in \mathcal{M}(\varphi)$, nous avons $\sup_{x \in \bar{x}} \check{C}_{P_\psi}(x) = \check{C}_{P_\psi}(x^*)$ et $\inf_{x \in \bar{x}} \check{C}_{P_\psi}(x) = \check{C}_{P_\psi}(x_*)$.*

Preuve. Compte tenu des propositions 2.1 et 2.2, nous savons que pour tout $i \in \Omega$, si $\varphi_i \geq 0$ alors $\psi_i \geq 0$ et si $\varphi_i < 0$ alors $\psi_i \leq 0$.

Par construction, $\check{C}_{P_\psi}(x) = \sum_{i \in \Omega} \psi_i^+ . x_i + \sum_{i \in \Omega} \psi_i^- . x_i$.

Donc, pour tout $x \in \bar{x}$, nous avons :

$$\check{C}_{P_\psi}(x) = \sum_{i \in \Omega} \psi_i . x_i \leq \sum_{i \in \Omega} \psi_i^+ . \bar{x}_i + \sum_{i \in \Omega} \psi_i^- . \underline{x}_i = \check{C}_{P_\psi}(x^*).$$

Cela nous donne que $\sup_{x \in \bar{x}} \check{C}_{P_\psi}(x) = \check{C}_{P_\psi}(x^*)$.

Le fait que $\inf_{x \in \bar{x}} \check{C}_{P_\psi}(x) = \check{C}_{P_\psi}(x_*)$ peut être démontré de manière similaire. \square

Dans cette proposition, nous montrons que les valeurs supérieure et inférieure de $\check{C}_{P_\psi}(x)$ pour tous vecteur x dans le vecteur d'intervalle \bar{x} sont atteintes exactement en x^* et x_* respectivement.

Cette proposition nous permet de simplifier le calcul des bornes supérieures et inférieures de l'intervalle de sortie de l'extension disjonctive de l'agrégation macsum.

Proposition 2.4. $\forall \bar{x} \in \mathbb{R}^n, \forall \varphi \in \mathbb{R}^n, \mathcal{D}_{\nu_\varphi}(\bar{x}) = [\check{\mathcal{C}}_{\nu_\varphi^c}(x_*), \check{\mathcal{C}}_{\nu_\varphi}(x^*)]$.

Preuve. On sait de [Strauss, 2022b] que $\forall x \in \mathbb{R}^n, \forall \varphi \in \mathbb{R}^n, \check{\mathcal{C}}_{\nu_\varphi}(x) = \sup_{\psi \in \mathcal{M}(\varphi)} \check{\mathcal{C}}_{P_\psi}(x)$, et de la Proposition 2.3 que $\sup_{x \in \bar{x}} \sup_{\psi \in \mathcal{M}(\varphi)} \check{\mathcal{C}}_{P_\psi}(x) = \sup_{\psi \in \mathcal{M}(\varphi)} \check{\mathcal{C}}_{P_\psi}(x^*)$. Ainsi, $\sup_{x \in \bar{x}} \check{\mathcal{C}}_{\nu_\varphi}(x) = \check{\mathcal{C}}_{\nu_\varphi}(x^*)$. On peut prouver que $\inf_{x \in \bar{x}} \check{\mathcal{C}}_{\nu_\varphi^c}(x) = \check{\mathcal{C}}_{\nu_\varphi^c}(x_*)$ de la même manière. \square

Ainsi, la relation entre le paramètre de l'agrégation macsum et les éléments du coeur macsumitif de ce paramètre permet de simplifier le calcul de l'agrégation macsum sur des entrées intervallistes.

Dans la section suivante, nous développons une extension moins prudente mais plus informative. Lorsqu'on parle d'une extension "moins prudente mais plus informative", on fait référence à une approche qui peut sembler moins exhaustive en termes de valeurs qu'elle englobe, mais qui, paradoxalement, peut fournir des informations plus précises ou pertinentes (moins un intervalle est large plus il est dit "précis").

L'agrégation disjonctive, en englobant toutes les valeurs possibles, peut aboutir à des intervalles très larges. Ces intervalles larges, bien qu'ils représentent fidèlement l'ensemble des valeurs possibles, peuvent être moins utiles en pratique, car ils ne permettent pas de distinguer des informations pertinentes de celles qui le sont moins.

2.3 Extension conjonctive aux entrées intervallistes

Le terme "conjonctif" provient du latin "conjunctivus", qui est dérivé de "conjungere", signifiant "lier" ou "joindre". Ici, le préfixe "con-" renvoie à l'idée d'association entre des éléments similaire, tandis que "jungere" suggère l'action de joindre. Par conséquent, "conjonctif" incarne l'acte d'unir des éléments qui auraient des propriétés en commun. À l'instar de l'agrégation qui cherche à assembler, l'agrégation conjonctive va au-delà en s'assurant que certaines conditions sont respectées pour former un ensemble unifié.

Dans le cadre logique ou mathématique, la conjonction est une opération qui est validée comme vraie uniquement lorsque tous ses opérandes le sont. Cette dynamique de conjonction est comparable aux intersections d'ensembles en mathématiques, où seuls les éléments communs aux ensembles considérés sont retenus.

Évoquant l'extension d'une agrégation à des entrées intervallistes avec une approche conjonctive, nous faisons référence à la méthode qui analyse chaque information dans l'intervalle en s'assurant qu'elles répondent à des critères communs.

Une agrégation conjonctive, assure ainsi une vision plus informative des résultats.

Nous allons établir deux approches distinctes.

La première approche, que nous allons appeler approche par concentration. Cette approche se traduit mathématiquement par l'équation (5.6) :

$$\begin{aligned} \mathcal{C}_{\nu_\varphi}^{\triangleleft}(\bar{x}) &= \bigcap_{x \in \bar{x}} \mathcal{A}_{\nu_\varphi}(x), \\ &= \bigcap_{x \in \bar{x}} \{\mathcal{A}_{P_\psi}(x) \mid \psi \in \mathcal{M}(\varphi)\}. \end{aligned} \quad (5.6)$$

Dans cette approche, l'intersection est effectuée sur l'ensemble des variables x appartenant à un ensemble de vecteurs.

La seconde approche, se nomme l'approche par dispersion. Elle est structurée différemment comme le montre l'équation (5.7) :

$$\begin{aligned} \mathcal{C}_{\nu_\varphi}^{\triangleright}(\bar{x}) &= \bigcap_{\psi \in \mathcal{M}(\varphi)} \mathcal{A}_{\psi_\psi}(\bar{x}), \\ &= \bigcap_{\psi \in \mathcal{M}(\varphi)} \{\mathcal{A}_{P_\psi}(x) \mid x \in \bar{x}\}. \end{aligned} \quad (5.7)$$

Ainsi, il existe deux manières possibles de créer cette agrégation conjonctive : soit en effectuant la conjonction sur les vecteurs d'entrée x appartenant au vecteur d'entrée à intervalles \bar{x} , prenant ainsi la contrepartie conjonctive de l'Équation (5.4), soit en effectuant la conjonction sur les vecteurs ψ appartenant au noyau *macsum* du paramètre *macsum* φ , prenant ainsi la contrepartie conjonctive de l'Équation (5.5). Considérons le calcul de l'équation (5.6). Si $\mathcal{C}_{\nu_\varphi}^{\triangleleft}(\bar{x}) \neq \emptyset$, alors :

$$\begin{aligned} \mathcal{C}_{\nu_\varphi}^{\triangleleft}(\bar{x}) &= \bigcap_{x \in \bar{x}} \mathcal{A}_{\nu_\varphi}(x), \\ &= \bigcap_{x \in \bar{x}} [\check{\mathcal{C}}_{\nu_\varphi^c}(x), \check{\mathcal{C}}_{\nu_\varphi}(x)], \\ &= \left[\sup_{x \in \bar{x}} \check{\mathcal{C}}_{\nu_\varphi^c}(x), \inf_{x \in \bar{x}} \check{\mathcal{C}}_{\nu_\varphi}(x) \right], \\ &= \left[\sup_{x \in \bar{x}} \inf_{\psi \in \mathcal{M}(\varphi)} \check{\mathcal{C}}_{P_\psi}(x), \inf_{x \in \bar{x}} \sup_{\psi \in \mathcal{M}(\varphi)} \check{\mathcal{C}}_{P_\psi}(x) \right]. \end{aligned} \quad (5.8)$$

De la même manière, le calcul de l'équation (5.7), si $\mathcal{C}_{\nu_\varphi}^{\triangleright}(\bar{x}) \neq \emptyset$, conduit à :

$$\begin{aligned} \mathcal{C}_{\nu_\varphi}^{\triangleright}(\bar{x}) &= \bigcap_{\psi \in \mathcal{M}(\varphi)} \mathcal{A}_{P_\psi}(\bar{x}), \\ &= \left[\sup_{\psi \in \mathcal{M}(\varphi)} \inf_{x \in \bar{x}} \check{\mathcal{C}}_{P_\psi}(x), \inf_{\psi \in \mathcal{M}(\varphi)} \sup_{x \in \bar{x}} \check{\mathcal{C}}_{P_\psi}(x) \right]. \end{aligned} \quad (5.9)$$

En comparant les équations (5.8) et (5.9), une observation notable est que la borne supérieure de $\mathcal{C}_{\nu_\varphi}^{\triangleright}(\bar{x})$ coïncide précisément avec la borne inférieure de $\mathcal{C}_{\nu_\varphi}^{\triangleleft}(\bar{x})$, et la borne inférieure de $\mathcal{C}_{\nu_\varphi}^{\triangleright}(\bar{x})$ est équivalente à la borne supérieure de $\mathcal{C}_{\nu_\varphi}^{\triangleleft}(\bar{x})$.

On a que si $\mathcal{C}_{\nu_\varphi}^{\triangleleft}(\bar{x}) = \emptyset$, alors on peut en déduire que $\mathcal{C}_{\nu_\varphi}^{\triangleright}(\bar{x})$ ne peut pas être vide. De même, si

$\mathcal{C}_{\nu_\varphi}^{\triangleright}(\bar{x}) = \emptyset$, alors cela implique que $\mathcal{C}_{\nu_\varphi}^{\triangleleft}(\bar{x})$ ne peut pas être vide.

En ce sens, on peut considérer que ces deux approches sont duales l'une de l'autre. Remarquons maintenant que, pour les mêmes raisons mentionnées dans la section 2.2,

$\sup_{x \in \bar{x}} \check{\mathcal{C}}_{\nu_\varphi^c}(x) = \check{\mathcal{C}}_{\nu_\varphi^c}(x^*)$ et $\inf_{x \in \bar{x}} \check{\mathcal{C}}_{\nu_\varphi}(x) = \check{\mathcal{C}}_{\nu_\varphi}(x_*)$. On a donc :

$$\begin{aligned}\mathcal{C}_{\nu_\varphi}^{\triangleleft}(\bar{x}) &= [\check{\mathcal{C}}_{\nu_\varphi^c}(x^*), \check{\mathcal{C}}_{\nu_\varphi}(x_*)], \text{ et} \\ \mathcal{C}_{\nu_\varphi}^{\triangleright}(\bar{x}) &= [\check{\mathcal{C}}_{\nu_\varphi}(x_*), \check{\mathcal{C}}_{\nu_\varphi^c}(x^*)].\end{aligned}$$

Pour l'extension conjonctive de l'agrégation macsum, lorsque les entrées sont valuées par intervalle, nous proposons de considérer l'expression suivante :

$$\mathcal{C}_{\nu_\varphi}(\bar{x}) = \left[\min(\check{\mathcal{C}}_{\nu_\varphi^c}(x^*), \check{\mathcal{C}}_{\nu_\varphi}(x_*)), \max(\check{\mathcal{C}}_{\nu_\varphi^c}(x^*), \check{\mathcal{C}}_{\nu_\varphi}(x_*)) \right].$$

Cette formule a la particularité de regrouper en une seule expression les deux visions de l'extension conjonctive.

L'autre avantage majeur de cette extension est qu'elle ne nécessite le calcul que de deux intégrales de Choquet. C'est donc une approche relativement économe en termes de calculs. Cela peut s'avérer particulièrement utile dans des contextes où les ressources de calcul sont limitées ou lorsque les entrées sont particulièrement nombreuses.

Remarque. Par le biais de l'approche disjonctive on démontre facilement que $\mathcal{D}_{\nu_\varphi}(\bar{x})$ est bien un ensemble convexe. Considérons deux scénarios distincts pour le calcul de $\mathcal{D}_{\nu_\varphi}(\bar{x})$. Dans le premier cas, si $\mathcal{C}_{\nu_\varphi}^{\triangleleft}(\bar{x}) \neq \emptyset$, alors pour tout couple de valeurs x_1 et x_2 de l'intervalle \bar{x} , l'intersection de $\mathcal{A}_{\nu_\varphi}(x_1)$ et $\mathcal{A}_{\nu_\varphi}(x_2)$ n'est pas vide. Ceci implique que l'ensemble $\mathcal{D}_{\nu_\varphi}(\bar{x})$ est convexe. Dans ce cas, l'ensemble $\mathcal{D}_{\nu_\varphi}(\bar{x})$ est l'union des ensembles $\mathcal{A}_{\nu_\varphi}(x)$ pour toutes les valeurs x dans \bar{x} . Cet ensemble peut être exprimé en termes d'intervalle formé par l'infimum et le supremum de $\mathcal{A}_{\nu_\varphi}(x)$ sur les valeurs x dans \bar{x} .

Dans le second cas, si $\mathcal{C}_{\nu_\varphi}^{\triangleright}(\bar{x}) \neq \emptyset$, alors pour tout couple d'éléments ψ^1 et ψ^2 de $\mathcal{M}(\varphi)$, l'intersection de $\mathcal{A}_{P_{\psi^1}}(\bar{x})$ et $\mathcal{A}_{P_{\psi^2}}(\bar{x})$ n'est pas vide. Ceci implique également que l'ensemble $\mathcal{D}_{\nu_\varphi}(\bar{x})$ est convexe. Dans ce cas, l'ensemble $\mathcal{D}_{\nu_\varphi}(\bar{x})$ est l'union des ensembles $\mathcal{A}_{P_\psi}(\bar{x})$ pour tous les éléments ψ dans $\mathcal{M}(\varphi)$. De la même manière, cet ensemble peut être exprimé en termes d'intervalle formé par l'infimum et le supremum de $\mathcal{A}_{P_\psi}(\bar{x})$ sur les éléments ψ dans $\mathcal{M}(\varphi)$.

L'agrégation conjonctive offre une alternative intéressante à l'agrégation disjonctive. La mise en contraste de deux stratégies d'agrégation conjonctive – l'une axée sur la concentration et l'autre sur la dispersion – souligne la richesse du potentiel de l'agrégation macsum à modéliser divers systèmes.

2.4 Exemple : Prédiction des températures dans une région à topographie variée

La topographie d'une région joue un rôle crucial dans la détermination de son climat. Une région avec des altitudes variées, comme des montagnes élevées juxtaposées à des vallées basses, présente des climats hétérogènes. Prédire la température dans une telle région peut être un défi en raison de ces différences d'altitude.

Considérons une région composée de trois zones géographiques distinctes : une montagne élevée, une plaine modérée et une vallée profonde. Pour ces zones, des prévisions de température basées sur des modèles météorologiques nous donnent les intervalles suivants : $\bar{x} = ([-2, 3], [0, 1], [-5, 8]) \in \mathbb{I}\mathbb{R}^3$.

Ici, chaque intervalle représente les températures minimales et maximales prévues pour une zone spécifique sur une période donnée. Pour pondérer l'importance de chaque zone (peut-être basée sur leur superficie ou leur densité de population), nous utilisons les poids : $\varphi = (-2, 3, -1) \in \mathbb{R}^3$. Nous allons calculer $\mathcal{D}_{\nu_\varphi}(\bar{x})$. Les températures minimales et maximales prévues pour chaque zone sont respectivement $x^* = (-2, 1, -5)$ et $x_* = (3, 0, 8)$.

Tout d'abord, nous trions x^* par ordre croissant pour obtenir $x_{(\cdot)}^* = (x_3^*, x_1^*, x_2^*) = (-5, -2, 1)$. Nous appliquons le même ordre de tri à φ pour obtenir $\varphi_{(\cdot)} = (\varphi_3, \varphi_1, \varphi_2) = (-1, -2, 3)$.

Nous utilisons ensuite la formule pour $\nu_\varphi(A_{(k)})$:

$$\nu_\varphi(A_{(k)}) = \max_{i \in A_{(k)}} \varphi_i^+ - \min_{i \in A_{(k)}^c} \varphi_i^- + \min_{i \in \Omega} \varphi_i^-,$$

où $A_{(k)} = \{(k), \dots, (n)\}$.

En substituant les valeurs dans la formule de $\check{\mathcal{C}}_{\nu_\varphi}(x^*)$, nous obtenons :

$$\begin{aligned} \check{\mathcal{C}}_{\nu_\varphi}(x^*) &= \sum_{k=1}^3 (x_{(k)}^* - x_{(k-1)}^*) \cdot \nu_\varphi(A_{(k)}), \\ &= (-5 - 0) \cdot \nu_\varphi(A_{(1)}) + (-2 - (-5)) \cdot \nu_\varphi(A_{(2)}) + (1 - (-2)) \cdot \nu_\varphi(A_{(3)}), \\ &= -5 + 6 + 9 = 10. \end{aligned}$$

Nous effectuons ensuite la même opération pour x_* en triant par ordre croissant pour obtenir $x_{*(\cdot)} = (x_{*2}, x_{*1}, x_{*3}) = (0, 3, 8)$ et $\varphi_{(\cdot)} = (\varphi_2, \varphi_1, \varphi_3) = (3, -2, -1)$.

En utilisant la formule pour $\nu_\varphi^c(A_{(k)})$, et en substituant les valeurs dans la formule de $\check{\mathcal{C}}_{\nu_\varphi^c}(x_*)$, nous obtenons :

$$\begin{aligned} \check{\mathcal{C}}_{\nu_\varphi^c}(x_*) &= \sum_{k=1}^3 (x_{*(k)} - x_{*(k-1)}) \cdot \nu_\varphi^c(A_{(k)}), \\ &= (0 - 0) \cdot \nu_\varphi^c(A_{(1)}) + (3 - 0) \cdot \nu_\varphi^c(A_{(2)}) + (8 - 3) \cdot \nu_\varphi^c(A_{(3)}), \\ &= 0 - 6 - 5 = -11. \end{aligned}$$

En conclusion, $\mathcal{D}_{\nu_\varphi}(\bar{x}) = [-11, 10]$. En utilisant une approche similaire, nous pourrions éga-

lement obtenir $\mathcal{C}_{\nu_\varphi}(\bar{x}) = [-6, 7]$. Dans cet exemple, on suppose que nous essayons de prédire les températures dans différentes zones géographiques en utilisant un système d'agrégation d'entrées d'intervalles. Les intervalles de température pour chaque zone sont donnés, ainsi que les poids pour chaque zone basés sur leur importance respective. En utilisant l'agrégation macsum, on calcule un intervalle de température agrégé en utilisant les températures minimales et maximales de chaque zone.

Cet exemple illustre la simplicité du calcul de l'agrégation macsum pour les entrées intervallistes. À partir d'un ensemble d'intervalles pour différentes zones géographiques et de poids attribués à chaque zone.

Ainsi, l'agrégation macsum est capable de traiter aussi bien des entrées précises que des entrées intervallistes. Toutefois, un point commun demeure : quelle que soit la nature de l'entrée, sa sortie est invariablement un intervalle, défini par un unique vecteur de paramètre. Dans la section suivante, nous cherchons à quantifier l'ensemble d'agrégations linéaires que représente l'agrégation macsum.

3 Granularité du coeur macsumitif d'un paramètre

Le terme de granularité a été introduit par Pawlak [Pawlak, 1991]. Cette notion a été étendue par Loquin [Loquin, 2008b] pour exprimer le cardinal du coeur d'un noyau maxitif, c'est-à-dire une mesure de son pouvoir de non-résolution. Dans [Loquin, 2008b], la granularité γ d'un noyau maxitif $\pi \in \mathbb{R}^n$ (c'est-à-dire le cardinal de son coeur) est définie par :

$$\gamma(\pi) = \sum_{i=1}^n \pi_i.$$

Le noyau maxitif le moins spécifique étant le noyau vacuous π défini par $\forall i \in \Omega, \pi_i = 1$, et le noyau maxitif le plus spécifique étant l'impulsion de Kronecker $\delta_k (k \in \Omega)$ définie par $\delta_k = 1$ et $\forall i \in \Omega \setminus \{k\}, \delta_i = 0$, on a donc $\forall \pi \in [0, 1]^n$,

$$\gamma(\pi) \in [1, n].$$

Dans cette section, nous avons pour objectif d'étendre la notion de granularité aux paramètres de \mathbb{R}^n . Cette démarche nous permet de caractériser le cardinal du coeur macsumitif d'un paramètre nous donnant ainsi le nombre de paramètres associés à des opérateurs additifs qu'il domine [Loquin, 2008a]. Cette information nous offre un aperçu du nombre d'agrégations linéaires représenté par l'agrégation macsum.

Commençons par présenter une proposition formelle :

Proposition 3.1. Soit $\varphi \in \mathbb{R}^n$, soit $\bar{\alpha} = \max_{i \in \Omega} \varphi_i$ et $\underline{\alpha} = \min_{i \in \Omega} \varphi_i$, alors si $\psi \in \mathcal{M}(\varphi)$,

$$\begin{aligned} \sum_{i \in \Omega} \max(0, \psi_i) &= \bar{\alpha}, \\ \sum_{i \in \Omega} \min(0, \psi_i) &= \underline{\alpha}. \end{aligned}$$

Preuve. Soit $A = \{i \in \Omega \mid \varphi_i \geq 0\}$, $\nu_\varphi(A) = \max_{i \in A} \varphi_i^+ + \underline{\alpha} - \min_{i \in A^c} \varphi_i^- = \max_{i \in \Omega} \varphi_i^+ = \bar{\alpha}$, et $\nu_\varphi^c(A) = \min_{i \in A} \varphi_i^- + \underline{\alpha} - \max_{i \in A^c} \varphi_i^+ = \underline{\alpha}$.

Puisque $P_\psi(A) = \sum_{i \in \Omega} \max(0, \psi_i)$ et $\nu_\varphi^c(A) \leq P_\psi(A) \leq \nu_\varphi(A)$,

$$\sum_{i \in \Omega} \max(0, \psi_i) = \bar{\alpha}.$$

De la même manière, nous avons $\nu_\varphi(A^c) = \nu_\varphi^c(A^c) = \underline{\alpha}$ et donc $P_\psi(A^c) = \sum_{i \in \Omega} \min(0, \psi_i) = \underline{\alpha}$.

□

Nous allons maintenant démontrer que le coeur macsumitif de l'homothétie d'un paramètre est l'ensemble des homothéties des éléments de son coeur pour le même rapport d'homothétie.

Proposition 3.2. Soit $\varphi \in \mathbb{R}^n$, $\psi \in \mathcal{M}(\varphi) \Leftrightarrow \forall \lambda \in \mathbb{R}^*$, $\lambda \cdot \psi \in \mathcal{M}(\lambda \cdot \varphi)$.

Preuve. $\psi \in \mathcal{M}(\varphi)$ signifie que $\forall A \in \Omega$, $\nu_\varphi^c(A) \leq P_\psi(A) \leq \nu_\varphi(A)$. Soit $\phi = \lambda \cdot \varphi$.

— Si $\lambda < 0$,

$$\begin{aligned} \lambda \nu_\varphi(A) &= \lambda \max_{i \in A} \varphi_i^+ + \lambda \min_{i \in \Omega} \varphi_i^- - \lambda \min_{i \in A^c} \varphi_i^- \\ &= \min_{i \in A} (\lambda \varphi_i^+) + \max_{i \in \Omega} (\lambda \varphi_i^-) - \max_{i \in A^c} (\lambda \varphi_i^-) \\ &= \min_{i \in A} \phi_i^- + \max_{i \in \Omega} \phi_i^+ - \max_{i \in A^c} \phi_i^+ \\ &= \nu_\phi^c(A). \end{aligned}$$

La même chose est vraie pour ν^c : $\lambda \nu_\varphi^c(A) = \nu_\phi(A)$.

— Si $\lambda > 0$,

$$\begin{aligned} \lambda \nu_\varphi(A) &= \lambda \max_{i \in A} \varphi_i^+ + \lambda \min_{i \in \Omega} \varphi_i^- - \lambda \min_{i \in A^c} \varphi_i^- \\ &= \max_{i \in A} (\lambda \varphi_i^+) + \min_{i \in \Omega} (\lambda \varphi_i^-) - \min_{i \in A^c} (\lambda \varphi_i^-) \\ &= \max_{i \in A} \phi_i^- + \min_{i \in \Omega} \phi_i^+ - \min_{i \in A^c} \phi_i^+ \\ &= \nu_\phi(A). \end{aligned}$$

La même chose est vraie pour ν^c : $\lambda \nu_\varphi^c(A) = \nu_\phi(A)$.

Par conséquent, $\nu_\varphi^c(A) \leq P_\psi(A) \leq \nu_\varphi(A) \Leftrightarrow \nu_\phi^c(A) \leq P_{\lambda \cdot \psi}(A) \leq \nu_\phi(A)$. □

Nous pouvons à présent introduire une extension de la définition de la granularité proposée par Loquin [Loquin, 2009], aux vecteurs de \mathbb{R}^n .

Definition 3.1. Soit $\varphi \in \mathbb{R}^n$, alors sa granularité est définie comme suit :

- $\gamma(\varphi) = \frac{1}{\bar{\alpha}} \sum_{i \in \Omega} \max(0, \varphi_i) + \frac{1}{\underline{\alpha}} \sum_{i \in \Omega} \min(0, \varphi_i)$, si $\bar{\alpha} > 0$ et $\underline{\alpha} < 0$,
- $\gamma(\varphi) = \frac{1}{\bar{\alpha}} \sum_{i \in \Omega} \max(0, \varphi_i)$, si $\bar{\alpha} > 0$ et $\underline{\alpha} \geq 0$,
- $\gamma(\varphi) = \frac{1}{\underline{\alpha}} \sum_{i \in \Omega} \min(0, \varphi_i)$, si $\bar{\alpha} \leq 0$ et $\underline{\alpha} < 0$,
- $\gamma(\varphi) = 0$, si $\underline{\alpha} = \bar{\alpha} = 0$.

Maintenant que, nous avons posé la définition de la granularité d'un paramètre nous allons établir une relation entre les granularités de deux paramètres. Cette proposition montre comment les relations de domination entre différents paramètres peuvent être caractérisées en termes de granularité.

Proposition 3.3. Soit $\varphi, \phi \in \mathbb{R}^n$, si $\max_{i \in \Omega} \varphi_i = \max_{i \in \Omega} \phi_i$ et $\min_{i \in \Omega} \varphi_i = \min_{i \in \Omega} \phi_i$, et si $\forall i \in \Omega$ nous avons $\max(0, \varphi_i) \leq \max(0, \phi_i)$ et $\min(0, \varphi_i) \geq \min(0, \phi_i)$, alors $\mathcal{M}(\varphi) \subseteq \mathcal{M}(\phi)$ et $\gamma(\varphi) \leq \gamma(\phi)$.

Preuve. Comme $\forall i \in \Omega$, $\max(0, \varphi_i) \leq \max(0, \phi_i)$ et $\min(0, \varphi_i) \geq \min(0, \phi_i)$, alors nous avons : $\forall A \subseteq \Omega$, $\nu_\varphi^c(A) \geq \nu_\phi^c(A)$ et $\nu_\varphi(A) \leq \nu_\phi(A)$.

Ainsi, si $\psi \in \mathcal{M}(\varphi)$ alors $\psi \in \mathcal{M}(\phi)$, c'est-à-dire $\mathcal{M}(\varphi) \subseteq \mathcal{M}(\phi)$.

Maintenant, soit $\bar{\alpha} = \max_{i \in \Omega} \varphi_i = \max_{i \in \Omega} \phi_i$ et $\underline{\alpha} = \min_{i \in \Omega} \varphi_i = \min_{i \in \Omega} \phi_i$.

Supposons que $\bar{\alpha} > 0$ et $\underline{\alpha} < 0$. Comme $\sum_{i \in \Omega} \max(0, \varphi_i) \leq \sum_{i \in \Omega} \max(0, \phi_i)$ et $\sum_{i \in \Omega} \min(0, \varphi_i) \geq \sum_{i \in \Omega} \min(0, \phi_i)$, alors :

$$\frac{1}{\bar{\alpha}} \sum_{i \in \Omega} \max(0, \varphi_i) + \frac{1}{\underline{\alpha}} \sum_{i \in \Omega} \min(0, \varphi_i) \leq \frac{1}{\bar{\alpha}} \sum_{i \in \Omega} \max(0, \phi_i) + \frac{1}{\underline{\alpha}} \sum_{i \in \Omega} \min(0, \phi_i)$$

et ainsi $\gamma(\varphi) \leq \gamma(\phi)$. Les cas où soit $\bar{\alpha} = 0$ soit $\underline{\alpha} = 0$ sont triviaux car l'inégalité revient à comparer respectivement les valeurs positives ou négatives de φ et de ϕ , ce qui découle des présupposés de la proposition. \square

Nous allons désormais nous intéresser à l'identification du paramètre dominant fort $\hat{\varphi} \in \mathbb{R}^n$, tel que $\psi \in \mathbb{R}^n$ appartient à $\mathcal{M}(\hat{\varphi})$, et établir que pour tout $\varphi \in \mathbb{R}^n$ avec ψ appartenant à $\mathcal{M}(\varphi)$, la granularité de φ est au moins la granularité de $\hat{\varphi}$. En termes simples, cela signifie que $\hat{\varphi}$ représente le paramètre le plus spécifique possible (dominant fort) pour ψ .

Proposition 3.4. Soit $\psi \in \mathbb{R}^n$, le paramètre $\hat{\varphi} \in \mathbb{R}^n$, défini dans la Proposition 2.9, est le paramètre le plus spécifique tel que $\psi \in \mathcal{M}(\hat{\varphi})$, c'est-à-dire $\forall \varphi \in \mathbb{R}^n$, si $\psi \in \mathcal{M}(\varphi)$ alors $\gamma(\varphi) \geq \gamma(\hat{\varphi})$.

Preuve. Soit $\hat{\varphi}$ le paramètre défini par $\forall i \in \Omega$, $\hat{\varphi}_i = \sum_{j \in A_i^+} \max(0, \psi_j) - \sum_{j \in A_i^-} \min(0, \psi_j)$, où $A_i^+ = \{j \in \Omega \mid \psi_j^+ \leq \psi_i^+\}$ et $A_i^- = \{j \in \Omega \mid \psi_j^- \geq \psi_i^-\}$.

Soit $B_i^+ = \{j \in \Omega \mid \psi_j > 0 \text{ et } \psi_j^+ \leq \psi_i^+\}$ et $B_i^- = \{j \in \Omega \mid \psi_j < 0 \text{ et } \psi_j^- \geq \psi_i^-\}$.

Nous avons $\forall i \in \Omega$, $\sum_{j \in B_i^+} \psi_j = \sum_{j \in A_i^+} \max(0, \psi_j)$ et $\sum_{j \in B_i^-} \psi_j = \sum_{j \in A_i^-} \min(0, \psi_j)$, ainsi

$$\dot{\varphi}_i = \sum_{j \in B_i^+} \psi_j - \sum_{j \in B_i^-} \psi_j.$$

Par construction, nous avons :

$$\begin{aligned} \forall i \in \Omega, \nu_{\dot{\varphi}}^c(B_i^+) &= \max_{j \in B_i^+} \dot{\varphi}_j^+ = \dot{\varphi}_i^+ \text{ puisque } \{j \mid \psi_j \leq 0\} \subseteq (B_i^+)^c, \\ \forall i \in \Omega, \nu_{\dot{\varphi}}^c(B_i^-) &= \min_{j \in B_i^-} -\dot{\varphi}_j^- = \dot{\varphi}_i^- \text{ puisque } \{j \mid \psi_j \geq 0\} \subseteq (B_i^-)^c. \end{aligned}$$

Maintenant, supposons qu'il existe $\varphi \in \mathbb{R}^n$ tel que $\psi \in \mathcal{M}(\varphi)$ et $\gamma(\varphi) < \gamma(\dot{\varphi})$.

Nous savons de la Proposition 3.1 que, puisque $\psi \in \mathcal{M}(\varphi)$ et $\psi \in \mathcal{M}(\dot{\varphi})$, alors $\bar{\alpha} = \max_{i \in \Omega} \varphi_i = \max_{i \in \Omega} \dot{\varphi}_i = \sum_{i \in \Omega} \max(0, \psi_i)$ et $\underline{\alpha} = \min_{i \in \Omega} \varphi_i = \min_{i \in \Omega} \dot{\varphi}_i = \sum_{i \in \Omega} \min(0, \psi_i)$.

L'inégalité $\gamma(\varphi) < \gamma(\dot{\varphi})$ est équivalente à :

$$\frac{1}{\bar{\alpha}} \sum_{i \in \Omega} \max(0, \varphi_i) + \frac{1}{\underline{\alpha}} \sum_{i \in \Omega} \min(0, \varphi_i) < \frac{1}{\bar{\alpha}} \sum_{i \in \Omega} \max(0, \dot{\varphi}_i) + \frac{1}{\underline{\alpha}} \sum_{i \in \Omega} \min(0, \dot{\varphi}_i).$$

L'inégalité ci-dessus est de la forme $a + b < c + d$ où a, b, c, d sont des valeurs réelles positives.

Donc, nous avons soit $a < c$ soit $a \geq c$, si $a \geq c$ cela implique que $b < d$.

Dans notre contexte ($\underline{\alpha} \leq 0$), nous avons soit :

$$\sum_{i \in \Omega} \max(0, \dot{\varphi}_i) > \sum_{i \in \Omega} \max(0, \varphi_i)$$

ou

$$\sum_{i \in \Omega} \min(0, \dot{\varphi}_i) < \sum_{i \in \Omega} \min(0, \varphi_i).$$

— Si $\sum_{i \in \Omega} \max(0, \dot{\varphi}_i) > \sum_{i \in \Omega} \max(0, \varphi_i)$, alors $\exists k \in \Omega$ tel que $\max(0, \dot{\varphi}_k) > \max(0, \varphi_k)$. Cette propriété est due à la séparation structurelle des valeurs négatives et positives dans les paramètres (i.e. $\forall i \in \Omega, \psi_i \leq 0 \Leftrightarrow \varphi_i \leq 0 \Leftrightarrow \dot{\varphi}_i \leq 0$ et réciproquement).

Comme $\nu_{\dot{\varphi}}^+(B_k^+) = \dot{\varphi}_k^+ = \sum_{j \in A_k^+} \max(0, \psi_j)$ et $\nu_{\varphi}^+(B_k^+) = \varphi_k^+$, nous avons $\nu_{\dot{\varphi}}^+(B_k^+) > \nu_{\varphi}^+(B_k^+)$.

Par conséquent, nous avons $\nu_{\varphi}^+(B_k^+) < \sum_{i \in B_k^+} \psi_i$ donc $\psi \notin \mathcal{M}(\varphi)$.

— Si $\sum_{i \in \Omega} \min(0, \dot{\varphi}_i) \leq \sum_{i \in \Omega} \min(0, \varphi_i)$, la même chose s'applique lorsque l'on considère $k \in \Omega$ tel que $\min(0, \dot{\varphi}_k) \leq \min(0, \varphi_k)$, conduisant à la conclusion que $\nu_{\dot{\varphi}}^-(B_k^-) \geq \sum_{i \in B_k^-} \psi_i$ donc $\psi \notin \mathcal{M}(\varphi)$.

Ainsi, par contraposition, nous pouvons conclure que $\psi \in \mathcal{M}(\varphi) \Rightarrow \gamma(\varphi) \geq \gamma(\dot{\varphi})$. \square

La proposition suivante montre que pour n'importe quel autre paramètre dominant le paramètre ψ , sa granularité ne peut pas être inférieure à celle du dominant faible $\check{\varphi}$.

Proposition 3.5. *Soit $\psi \in \mathbb{R}^n$, le paramètre $\check{\varphi} \in \mathbb{R}^n$, défini par $\forall i \in \Omega$, si $\psi_i \geq 0$, alors $\check{\varphi}_i = \sum_{k \in \Omega} \max(\psi_k, 0)$ sinon $\check{\varphi}_i = \sum_{k \in \Omega} \min(\psi_k, 0)$, est le paramètre le moins spécifique tel que $\psi \in \mathcal{M}(\check{\varphi})$, c'est-à-dire $\forall \varphi \in \mathbb{R}^n$, si $\psi \in \mathcal{M}(\varphi)$ alors $\gamma(\varphi) \leq \gamma(\check{\varphi})$.*

Preuve. Commençons par démontrer que $\psi \in \mathcal{M}(\check{\varphi})$.

Soit $\mathcal{P}(\Omega) = 2^\Omega$ l'ensemble de toutes les parties de Ω et soit $\mathcal{P}^+(\Omega) = \{A \subseteq \Omega \mid \forall i \in A, \psi_i \geq 0\}$, $\mathcal{P}^-(\Omega) = \{A \subseteq \Omega \mid \forall i \in A, \psi_i < 0\}$, et $\mathcal{P}^\pm(\Omega) = \{A \subseteq \Omega \mid \exists i \in A \text{ tel que } \psi_i \geq 0 \text{ et } \exists j \in A \text{ tel que } \psi_j < 0\}$.

Par construction, ces trois sous-ensembles forment une partition de $\mathcal{P}(\Omega)$. Soit $A^+ = \{i \in \Omega \mid \psi_i > 0\}$, $A^- = \{i \in \Omega \mid \psi_i < 0\}$ et $A \subseteq \Omega$.

Soit $\underline{\alpha} = \sum_{k \in \Omega} \min(\psi_k, 0) = \min_{k \in \Omega} \check{\varphi}_k$ et $\bar{\alpha} = \sum_{k \in \Omega} \max(\psi_k, 0) = \max_{k \in \Omega} \check{\varphi}_k$.

— Si $A \in \mathcal{P}^\pm(\Omega)$ et $A^c \in \mathcal{P}^\pm(\Omega)$, nous avons :

$$\begin{aligned} \nu_{\check{\varphi}}(A) &= \max_{i \in A} \check{\varphi}_i^+ + \min_{i \in \Omega} \check{\varphi}_i^- - \min_{i \in A^c} \check{\varphi}_i^- \\ &= \bar{\alpha} + \underline{\alpha} - \underline{\alpha} \\ &= \bar{\alpha}, \end{aligned}$$

et

$$\begin{aligned} \nu_{\check{\varphi}}(A^c) &= \min_{i \in A} \check{\varphi}_i^- + \max_{i \in \Omega} \check{\varphi}_i^+ - \max_{i \in A^c} \check{\varphi}_i^+ \\ &= \underline{\alpha} + \bar{\alpha} - \bar{\alpha} \\ &= \underline{\alpha}. \end{aligned}$$

Ainsi, puisque par construction $\underline{\alpha} \leq \sum_{i \in A} \psi_i \leq \bar{\alpha}$, nous avons $\nu_{\check{\varphi}}(A^c) \leq P_\psi(A) \leq \nu_{\check{\varphi}}(A)$.

— Si $A \in \mathcal{P}^-(\Omega)$ et $A^c \in \mathcal{P}^\pm(\Omega)$, nous avons : $\nu_{\check{\varphi}}(A) = \underline{\alpha} - \underline{\alpha} = 0$, et $\sum_{i \in A} \psi_i \leq 0$.

Maintenant, puisque $A \in \mathcal{P}^-(\Omega)$, $A^+ \subseteq A^c$, nous avons $\nu_{\check{\varphi}}^c(A) = \underline{\alpha} + \bar{\alpha} - \bar{\alpha} = \underline{\alpha}$ et $\sum_{i \in A} \psi_i \geq \underline{\alpha}$.

Ainsi, $\nu_{\check{\varphi}}^c(A) \leq P_\psi(A) \leq \nu_{\check{\varphi}}(A)$.

— Si $A \in \mathcal{P}^+(\Omega)$ et $A^c \in \mathcal{P}^\pm(\Omega)$, nous avons : $\nu_{\check{\varphi}}(A) = \bar{\alpha}$ et $\sum_{i \in A} \psi_i \leq \bar{\alpha}$.

De plus, comme $A \in \mathcal{P}^+(\Omega)$, $A^- \subseteq A^c$, $\nu_{\check{\varphi}}^c(A) = \bar{\alpha} - \bar{\alpha} = 0$ et $\sum_{i \in A} \psi_i \geq 0$.

Ainsi, $\nu_{\check{\varphi}}^c(A) \leq P_\psi(A) \leq \nu_{\check{\varphi}}(A)$.

— Si $A \in \mathcal{P}^-(\Omega)$ et $A^c \in \mathcal{P}^+(\Omega)$, nous avons : $\nu_{\check{\varphi}}(A) = \underline{\alpha}$, $\nu_{\check{\varphi}}^c(A) = \underline{\alpha}$ et $\sum_{i \in A} \psi_i = \underline{\alpha}$, donc $\nu_{\check{\varphi}}^c(A) = P_\psi(A) = \nu_{\check{\varphi}}(A)$.

Les cas où $(A \in \mathcal{P}^\pm(\Omega) \text{ et } A^c \in \mathcal{P}^+(\Omega))$ ou $(A \in \mathcal{P}^\pm(\Omega) \text{ et } A^c \in \mathcal{P}^-(\Omega))$ ou $(A \in \mathcal{P}^+(\Omega) \text{ et } A^c \in \mathcal{P}^-(\Omega))$ peuvent être prouvés de la même manière que ci-dessus.

Les cas où $A \in \mathcal{P}^+(\Omega)$ et $A^c \in \mathcal{P}^+(\Omega)$ (ce qui implique $\forall i \in \Omega, \psi_i \geq 0$) ainsi que les cas où $A \in \mathcal{P}^-(\Omega)$ et $A^c \in \mathcal{P}^-(\Omega)$ (ce qui implique $\forall i \in \Omega, \psi_i \leq 0$) sont triviaux.

Ainsi, $\psi \in \mathcal{M}(\check{\varphi})$.

Maintenant, supposons qu'il existe $\varphi \in \mathbb{R}^n$ tel que $\psi \in \mathcal{M}(\varphi)$ et $\gamma(\check{\varphi}) > \gamma(\varphi)$. Nous savons qu'avec $\psi \in \mathcal{M}(\check{\varphi})$ et $\psi \in \mathcal{M}(\varphi)$ alors : $\underline{\alpha} = \max_{i \in \Omega} \check{\varphi}_i = \max_{i \in \Omega} \varphi_i$ et $\underline{\alpha} = \min_{i \in \Omega} \check{\varphi}_i = \min_{i \in \Omega} \varphi_i$.

Par conséquent, $\gamma(\varphi) > \gamma(\check{\varphi})$ implique qu'il existe un $k \in \Omega$ tel que $\max(0, \varphi_k) \geq \max(0, \check{\varphi}_k)$ (i.e. $\varphi_k \geq \alpha$) ou qu'il existe un $k \in \Omega$ tel que $\min(0, \varphi_k) \leq \min(0, \check{\varphi}_k)$ (i.e. $\varphi_k \leq \underline{\alpha}$), ce qui contredit l'hypothèse, donc $\gamma(\check{\varphi}) \leq \gamma(\varphi)$.

□

Remarque. $\forall \varphi \in \mathbb{R}^n, \gamma(\varphi) \in [0, n]$.

4 Conclusion

Dans ce chapitre l'agrégation macsum a été enrichie par le développement de deux modèles d'extension de l'agrégation macsum aux données sous forme d'intervalles. Ces modèles, ont des propriétés et des comportements distincts, l'un étant plus précis que l'autre. Ceci est significatif, car cela pourrait être interprété comme un mécanisme de contrôle de la précision recherchée. En d'autres termes, le choix de l'extension offre la possibilité d'ajuster la précision de l'intervalle de sortie en fonction des besoins spécifiques de l'étude. Ce développement, en lien direct avec la nature souvent imprécise des données dans le monde réel, nous offre la possibilité d'étendre l'utilisation de l'agrégation macsum à des scénarios plus incertains. Par ailleurs dans ce chapitre nous avons introduit la notion de granularité de l'opérateur macsum. Nous avons ainsi défini un indicateur permettant de caractériser la quantité d'agrégations linéaires représenté par l'agrégation macsum pour un paramètre donné.

Conclusion et perspectives

Dans ce manuscrit, nous avons proposé, sous le nom de macsum, une modélisation assez nouvelle d'un système avec un modèle agrégeant des données réelles et produisant un intervalle. Ce modèle agrégatif a trois spécificités qui en font l'originalité. La première de ces originalités est que chaque donnée est associée à un poids comme dans le cas des agrégations par somme pondérée. La deuxième est que cette agrégation représente en fait un ensemble convexe d'agrégations par somme pondérée, et que donc l'intervalle de sortie n'est autre que l'intervalle de toutes les sorties qui auraient été obtenues par les sommes pondérées représentées. La troisième est que ce modèle utilise une intégrale de Choquet par rapport à une fonction d'ensemble non normalisée, non monotone et à valeur dans \mathbb{R} . De cette façon, l'agrégation macsum peut être calculée avec des paramètres prenant n'importe quelle valeur réelle. Lorsque les paramètres respectent les contraintes de normalisation et de positivité, l'agrégation macsum correspond à une agrégation maxitive. C'est dans ce sens que l'agrégation macsum est une généralisation de l'agrégation maxitive introduite par Loquin. L'idée est d'obtenir un modèle d'apprentissage intervalliste avec lequel on peut aisément appliquer la méthode de descente de gradient pour la phase d'apprentissage. Dans cette optique, nous avons levé les contraintes de l'agrégation maxitive en introduisant une fonction d'ensemble non monotone et concave qui généralise la mesure de possibilité. De plus, en nous basant sur les travaux de Dubois et Rico [Dubois, 2016], nous avons formulé l'agrégation macsum de manière à réduire la complexité du calcul de sa dérivée par rapport à ses paramètres. Cette reformulation a permis de simplifier l'utilisation de l'agrégation macsum comme modèle d'apprentissage. Nous avons également expérimenté dans le domaine du traitement d'image l'apprentissage de l'agrégation macsum. Cette expérimentation a montré son potentiel de représentativité, et a établi que l'apport de cette thèse est à la fois théorique et pratique. La généralisation de l'agrégation maxitive a permis de proposer un modèle d'apprentissage qui représente un ensemble convexe de modèles linéaires tout en conservant la facilité d'utilisation d'un modèle linéaire simple.

Une autre originalité réside dans l'utilisation des propriétés de cette agrégation pour élaborer des extensions aisément calculables, adaptées aux entrées intervallistes. Nous sommes parvenus à établir quatre extensions. Une extension disjonctive qui englobe l'information totale fournie par les agrégations macsum calculées à partir des éléments de l'intervalle d'entrée considéré. Et deux autres conjonctives qui donnent un intervalle plus restreint, mais plus informatif. Les deux extensions conjonctives s'avèrent être duales l'une de l'autre, ce qui nous a conduit à établir une

quatrième extension qui est une synthèse des deux extensions conjonctives et permet d'obtenir un intervalle de sortie qui n'est jamais vide.

Finalement, cette thèse a introduit un nouveau modèle d'apprentissage permettant de modéliser une pluralité de systèmes avec des entrées précises ou imprécises. Une perspective future est l'apprentissage de systèmes à entrées et sorties imprécises en considérant une fonction de coût qui compare des intervalles. De plus, la simplicité d'utilisation de l'agrégation macsum, comparable à celle d'un modèle linéaire, suggère une piste intéressante : envisager des réseaux de neurones intervallistes où l'on remplace la somme pondérée de chaque neurone par l'agrégation macsum. Dans cette même démarche, on peut envisager des réseaux de neurones employant les extensions conjonctives et/ou disjonctives, permettant ainsi de traiter des entrées imprécises. En outre, on a vu que l'agrégation macsum peut représenter un ensemble convexe de produits de convolution. On peut envisager d'utiliser l'agrégation macsum dans les couches de convolution des réseaux de neurones convolutifs. D'autre part, une limite claire de l'agrégation macsum est la contrainte concernant la similitude du gain des modèles linéaires qu'il représente. Une perspective intéressante est de s'affranchir de cette limite. Une autre limite est le manque de contrôle de la taille de son intervalle de sortie. On peut tenter de rajouter, dans le processus d'apprentissage, l'écart entre les bornes de l'intervalle de sortie comme critère à minimiser. En conclusion, les possibilités offertes par ce modèle nouvellement introduit sont vastes. De nombreuses questions et défis restent en suspens, offrant de nombreuses pistes pour de futures recherches. D'autant plus que la capacité à représenter un ensemble convexe d'agrégations linéaires avec un seul paramètre, découle de la concavité de l'opérateur macsum. Explorer d'autres fonctions d'ensemble concaves peut ouvrir la voie vers la création de modèles intervallistes plus adaptés à certains systèmes, moins gourmands en calculs ou offrant une meilleure explicabilité et interprétabilité, voire une combinaison de ces avantages. On peut également envisager que l'impact de ces travaux aille au-delà du domaine de l'apprentissage, élargissant l'application de l'intégrale de Choquet avec des fonctions d'ensemble non-monotones dans des domaines où elle est déjà prédominante, comme la théorie de la décision. Cette démarche pourrait conduire à une exploration approfondie de l'application des fonctions d'ensemble non-monotones dans divers domaines.

Bibliographie

- [Agrawal, 2013] J. G. AGRAWAL, V. S. CHOURASIA et A. K. MITTRA. « State-of-the-art in stock prediction techniques ». *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering* 2.4 (2013), p. 1360-1374 (cf. p. 7).
- [Ahn, 2007] B. AHN. « The OWA aggregation with uncertain descriptions on weights and input arguments ». *IEEE Transactions on Fuzzy Systems* 15.6 (2007), p. 1130-1134 (cf. p. 21).
- [Baarspul, 1990] M. BAARSFUL. « A review of flight simulation techniques ». *Progress in Aerospace Sciences* 27.1 (1990), p. 1-120 (cf. p. 7).
- [Barone, 1978] J. BARONE et A. NOVIKOFF. « A history of the axiomatic formulation of probability from Borel to Kolmogorov : part I ». *Archive for History of Exact Sciences* 18 (1978), p. 123-190 (cf. p. 13).
- [Beliakov, 2007] G. BELIAKOV, A. PRADERA et T. CALVO. *Aggregation functions : a guide for practitioners*. Springer, 2007 (cf. p. 13).
- [Borwein, 2006] J. BORWEIN et A. LEWIS. « Convex analysis ». *Convex Analysis and Nonlinear Optimization*. CMS Books in Mathematics. Springer, 2006 (cf. p. 27, 61).
- [Campagner, 2022] A. CAMPAGNER, L. FAMIGLINI et F. CABITZA. « Re-calibrating machine learning models using confidence interval bounds ». *Modeling Decisions for Artificial Intelligence* 32 (2022), p. 132-142 (cf. p. 19).
- [Charles, 2016] F. CHARLES. « Credible interval estimates for official statistics with survey nonresponse ». *Journal of Econometrics* 191.2 (2016), p. 293-301 (cf. p. 20).
- [Chen, 2013] T. CHEN. « An interactive method for multiple criteria group decision analysis based on interval type-2 fuzzy sets and its application to medical decision making ». *Fuzzy Optimization and Decision Making* 12 (2013), p. 323-356 (cf. p. 9).
- [Choquet, 1954] G. CHOQUET. « Theory of capacities ». *Annales de l'Institut Fourier* 5 (1954), p. 131-295 (cf. p. 15).
- [Chudý, 2020] M. CHUDÝ, S. KARMAKAR et W. B. WU. « Long-term prediction intervals of economic time series ». *Empirical Economics* 58 (2020), p. 191-222 (cf. p. 9).
- [Cunningham, 2008] P. CUNNINGHAM, M. CORD et S. J. DELANY. *Supervised learning*. Springer, 2008, p. 21-49 (cf. p. 7).
- [Delbracio, 2013] M. DELBRACIO. « Two problems of digital image formation : recovering the camera point spread function and boosting stochastic renderers by auto-similarity filtering ». Thèse de doct. 2013 (cf. p. 51).
- [Denneberg, 1994] D. DENNEBERG. *Non-additive measure and integral*. Kluwer Academic Publishers, Dordrecht, 1994 (cf. p. 13, 15, 32).
- [Dubois, 2016] D. DUBOIS et A. RICO. « Axiomatisation of discrete fuzzy integrals with respect to possibility and necessity measures » (2016), p. 94-106 (cf. p. 41, 74).
- [El Naqa, 2015] I. EL NAQA et M. J. MURPHY. *What is machine learning ?* 2015, p. 3-11 (cf. p. 11).
- [Eluyode, 2013] O.S. ELUYODE et D. AKOMOLAFE. « Comparative study of biological and artificial neural networks ». *European Journal of Applied Engineering and Scientific Research* 2.1 (2013), p. 36-46 (cf. p. 9).
- [Ferderer, 1996] J. P. FERDERER. « Oil price volatility and the macroeconomy ». *Journal of Macroeconomics* 18 (1996), p. 1-26 (cf. p. 7).
- [Garczarczyk, 2000] Z-A. GARCZARCZYK. « Interval neural networks ». *IEEE International Symposium on Circuits and Systems* 3 (2000), p. 567-570 (cf. p. 21).

- [Gershman, 2012] S. J. GERSHMAN et D. M. BLEI. « A tutorial on Bayesian nonparametric models ». *Journal of Mathematical Psychology* 56.1 (2012), p. 1-12 (cf. p. 10).
- [Goodfellow, 2016] I. GOODFELLOW, Y. BENGIO et A. COURVILLE. *Deep learning*. Adaptive computation and machine learning. MIT Press, 2016 (cf. p. 9, 49).
- [Grabisch, 2015] M. GRABISCH. « Fuzzy measures and integrals : recent developments. fifty years of fuzzy logic and its applications ». *Fifty years of fuzzy logic and its applications* (2015), p. 125-151 (cf. p. 16).
- [Grabisch, 2016] M. GRABISCH. *Set functions, games and capacities in decision making*. Springer, 2016 (cf. p. 15).
- [Grabisch, 2009] M. GRABISCH, J. MARICHAL, R. MESIAR et E. PAP. « Aggregation functions ». *Cambridge Press* 127 (2009), p. 88-100 (cf. p. 12, 40).
- [Grabisch, 2000] M. GRABISCH, M. SUGENO et T. MUROFUSHI. « Fuzzy measures and integrals : theory and applications ». *Heidelberg : Physica* 37 (2000), p. 25-37 (cf. p. 15, 16, 27, 61).
- [Guo, 2003] G. GUO, H. WANG, D. BELL, Y. BI et K. GREER. « KNN model-based approach in classification ». 12 (2003), p. 986-996 (cf. p. 10).
- [Hastie, 2001] T. HASTIE, R. TIBSHIRANI et J. FRIEDMAN. « The elements of statistical learning ». Springer Series in Statistics (2001) (cf. p. 9, 11, 17, 19, 49).
- [Havens, 2019] T. HAVENS et D. ANDERSON. « Machine learning of Choquet integral regression with respect to a bounded capacity (or non-monotonic fuzzy measure) ». *IEEE International Conference on Fuzzy Systems* 28 (2019), p. 1-6 (cf. p. 17).
- [Herin, 2022] M. HERIN, P. PERNY et N. SOKOLOVSKA. « Learning sparse representations of preferences within Choquet expected utility theory ». 180 (2022), p. 800-810 (cf. p. 17).
- [Hickey, 2001] T. HICKEY, Q. JU et M. H. VAN EMDEN. « Interval arithmetic : From principles to implementation ». *Journal of the ACM* 48.5 (2001), p. 1038-1068 (cf. p. 21).
- [Hladík, 2012] M. HLADÍK et M. ČERNÝ. « Interval regression by tolerance analysis approach ». *Fuzzy Sets and Systems* 193 (2012), p. 85-107 (cf. p. 22).
- [Hmidy, 2023a] Y. HMIDY, A. RICO et O. STRAUSS. « Extending the macsum aggregation to interval-valued inputs » (2023), p. 338-347 (cf. p. 59).
- [Hmidy, 2023b] Y. HMIDY, A. RICO et O. STRAUSS. « Macsum aggregation learning ». *Fuzzy Sets and Systems* 459 (2023), p. 182-200 (cf. p. 39).
- [Hong, 2022] L. HONG. « Conformal prediction credibility intervals ». *North American Actuarial Journal* 3.6 (2022), p. 453 (cf. p. 20).
- [Janssen, 2015] P. JANSSEN et R. STOUFFS. « Types of parametric modelling ». 78.7 (2015), p. 76-84 (cf. p. 9, 10).
- [Jaulin, 2001] L. JAULIN, M. KIEFFER, O. DIDRIT et E. WALTER. *Applied interval analysis with examples in parameter and state estimation, robust control and robotics*. Springer, 2001, p. 321-328 (cf. p. 21).
- [Jiang, 2008] B. JIANG, X. ZHAN et T. CAI. « Estimating the confidence interval for prediction errors of support vector machine classifiers ». *Journal of Machine Learning Research* 9 (2008), p. 521-540 (cf. p. 19).
- [Jiang, 2020] H. JIANG, Y. WANG, X. LI, H. ZHAO et Y. LI. « Space-variant point spread function measurement and interpolation at any depth based on single-pixel imaging ». *Optic Express* 28.7 (2020), p. 9244-9258 (cf. p. 51).
- [Jin-Tsong, 2003] J. JIN-TSONG, C. CHEN-CHIA et S. SHUN-FENG. « Support vector interval regression networks for interval regression analysis ». *Fuzzy Sets and Systems* 138.2 (2003), p. 283-300 (cf. p. 22).
- [Kapur, 1988] J. N. KAPUR. « Mathematical modelling ». *New Age International* 58 (1988), p. 1-259 (cf. p. 9).
- [Kingsford, 2008] C. KINGSFORD et S. L. SALZBERG. « What are decision trees? » *Nature Biotechnology* 26 (2008), p. 1011-1013 (cf. p. 10).
- [Kuleshov, 2018] V. KULESHOV, N. FENNER et S. ERMON. « Accurate uncertainties for deep learning using calibrated regression ». 80 (2018), p. 2796-2804 (cf. p. 19).
- [Loquin, 2008a] K. LOQUIN. *On the use of maxitive kernels in information processing*. 2008 (cf. p. 3, 22, 23, 26, 27, 30, 32, 68).
- [Loquin, 2008b] K. LOQUIN et O. STRAUSS. « On the granularity of summative kernels ». *Fuzzy Sets and Systems* 159.15 (2008), p. 1952-1972 (cf. p. 30, 68).

- [Loquin, 2009] K. LOQUIN et O. STRAUSS. « Noise quantization via possibilistic filtering ». *International Symposium on Imprecise Probability : Theories and Application* 6 (2009), p. 297-306 (cf. p. 70).
- [Manohar, 2016] N. MANOHAR, S. KUMAR et G. HEMANTHA. « Supervised and unsupervised learning in animal classification » (2016), p. 32-44 (cf. p. 8).
- [Mendez, 2019] K.M. MENDEZ, S.N. REINKE et D.I. BROADHURST. « A comparative evaluation of the generalised predictive ability of eight machine learning algorithms across ten clinical metabolomics data sets for binary classification ». *Metabolomics* 15 (2019), p. 150-165 (cf. p. 19).
- [Miller, 1997] R. MILLER et Y. YANG. « Association rules over interval data ». *ACM SIGMOD Record* 26.2 (1997), p. 452-461 (cf. p. 59).
- [Moya, 2022] A. MOYA et A. MERINO. « Building a digital barometer with Arduino to study forces due to atmospheric pressure and weather maps at schools ». *Physics Education* 57.4 (2022) (cf. p. 7).
- [Murofushi, 1994] T. MUROFUSHI, M. SUGENO et M. MACHIDA. « Non-monotonic fuzzy measures and the Choquet integral ». *Fuzzy Sets and Systems* 64.1 (1994), p. 73-86 (cf. p. 15, 16).
- [Mykland, 2003] P. MYKLAND. « Financial options and statistical prediction intervals ». *Ann. Statist.* 31.5 (2003), p. 1413-1438 (cf. p. 20).
- [Narukawa, 2006] Y. NARUKAWA et V. TORRA. « Non-monotonic fuzzy measures and intuitionistic fuzzy sets ». *Modeling Decisions for Artificial Intelligence* 51 (2006), p. 150-160 (cf. p. 16).
- [Nikulchev, 2023] E. NIKULCHEV et A. CHERVYAKOV. « Prediction intervals : a geometric view ». *Symmetry* 15.4 (2023), p. 781 (cf. p. 21).
- [Novak, 2021] K. NOVAK et A. WATNIK. « Imaging through deconvolution with a spatially variant point spread function ». 11731 (2021), p. 24-40 (cf. p. 51).
- [Patel, 1989] J. K. PATEL. « Prediction intervals - a review ». *Communications in Statistics - Theory and Methods* 18.7 (1989), p. 2393-2465 (cf. p. 19).
- [Pawlak, 1991] Z. PAWLAK. *Rough sets : theoretical aspects of reasoning about data*. Kluwer Academic Publisher, 1991 (cf. p. 68).
- [Pekaslan, 2021] D. PEKASLAN et C. WAGNER. « Compositional linear regression on interval-valued data ». *IEEE Symposium Series on Computational Intelligence (SSCI)* 5 (2021), p. 01-08 (cf. p. 21).
- [Ranathunga, 2022] N. RANATHUNGA. « Confidence intervals in general regression models that utilise uncertain prior information ». *Bulletin of the Australian Mathematical Society* 105.3 (2022), p. 516-517 (cf. p. 19).
- [Raschka, 2018] S. RASCHKA. « Model evaluation, model selection, and algorithm selection in machine learning ». 11 (2018), p. 12-23 (cf. p. 19).
- [Rish, 2001] I. RISH. « An empirical study of the naive Bayes classifier ». *Econometrica* 3.22 (2001), p. 41-46 (cf. p. 9).
- [Saulnier,] H. SAULNIER, O. STRAUSS et I. COUSO. « Comparing interval-valued estimations with point-valued estimations » (), p. 595-604 (cf. p. 50, 54).
- [Schmeidler, 1986] D. SCHMEIDLER. « Integral representation without additivity ». *Proceedings of the American Mathematical Society* 97 (1986), p. 255-261 (cf. p. 15, 29, 33).
- [Strauss, 2019] O. STRAUSS et A. RICO. « Where the domination of maxitive kernels is extended to signed values ». *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)* 21 (2019), p. 1-6 (cf. p. 27).
- [Strauss, 2022a] O. STRAUSS, A. RICO et Y. HMIDY. « Macsum : a new interval-valued linear operator ». *International Journal of Approximate Reasoning* 145 (2022), p. 121-138 (cf. p. 26, 59).
- [Strauss, 2022b] O. STRAUSS, A. RICO et Y. HMIDY. « Macsum : a new interval-valued linear operator ». *International journal of approximate reasoning* 145 (2022), p. 121-138 (cf. p. 64).
- [Tehrani, 2010] A. TEHRANI, W. CHENG et E. HÜLLERMEIER. « Preference learning using the choquet integral : the case of multipartite ranking ». *Proceedings of the 20th Workshop Computational Intelligence* (2010), p. 119-130 (cf. p. 17).
- [Theil, 1957] H. THEIL. « Linear aggregation in input-output analysis ». *Econometrica* 25.1 (1957), p. 111-122 (cf. p. 16).
- [Van de Schoot, 2021] R. VAN DE SCHOOT, S. DEPAOLI, R. KING, B. KRAMER, M. KASPAR, G. MAHLET et al. « Bayesian statistics and modelling ». *Nature Reviews Methods Primers* 1 (2021), p. 1-22 (cf. p. 20).
- [Waegenaere, 1997] A. De WAEGENAERE et P. WAKKER. « Choquet integrals with respect to non-monotonic set functions ». *Tilburg Press* 44 (1997), p. 17-21 (cf. p. 15, 16).

- [Wang, 2016] Y. WANG et J. LI. « Credible intervals for precision and recall based on a k-fold cross-calidated beta distribution ». *Neural Computation* 28.8 (2016), p. 1694-1722 (cf. p. 20).
- [Winkler, 1967] R. L. WINKLER. « The assessment of prior distributions in bayesian analysis ». *Journal of the American Statistical Association* 62.319 (1967), p. 776-800 (cf. p. 20).