Evaluation d'une mesure de similitude en classification supervisée : application à la préparation de données séquentielles

Sylvain Ferrandiz

UNIVERSITÉ de CAEN/BASSE-NORMANDIE

U.F.R. : Sciences

ÉCOLE DOCTORALE : SIMEM

THÈSE

présentée par

Sylvain FERRANDIZ

et soutenue

le 23 octobre 2006

en vue de l'obtention du

DOCTORAT de l'UNIVERSITÉ de CAEN

spécialité : Sciences et Technologies de l'Information

(Arrêté du 25 avril 2002)

Apprentissage supervisé à partir de données séquentielles

MEMBRES du JURY

Samy Bengio	**IDIAP de Martigny**	**Rapporteur**
Marc Sebban	**Université de Saint-Etienne**	**Rapporteur**
Bruno Crémilleux	**Université de Caen**	**Directeur**
Marc Boullé	**France Télécom R&D de Lannion**	**Co-encadrant**
François Kauffmann	**Université de Caen**	**Co-encadrant**
Yves Lechevallier	**INRIA Rocquencourt**	**Examinateur**

Groupe de Recherche en Informatique, Image, Automatique et Instrumentation de Caen — CNRS UMR 6072

Mis en page avec la classe thloria.

Remerciements

En ayant pris soin de définir un sujet ouvert à partir d'une problématique industrielle , Marc BOULLÉ m'a permis de mener un travail complet d'innovation, de développement, de validation expérimentale et de diffusion scientifique. Mais c'est aussi pour son investissement quotidien tout au long de ces trois années que je tiens à lui signifier toute ma gratitude.

Il est difficile pour moi de dissocier Bruno CRÉMILLEUX et François KAUFFMANN. S'ils ont accepté respectivement d'encadrer et de co-encadrer universitairement mon travail, ils ont surtout toujours répondu présent à mes sollicitations, conjointement et malgré de sérieux embouteillages dans leurs agendas. Je les remercie pour les précieux conseils formulés lors de nos entrevues. Celles-ci ont grandement contribué à l'amélioration de la présentation de mon travail, tant à l'écrit qu'à l'oral.

Relire et rapporter un manuscrit de thèse demande une implication certaine, sans contrepartie autre que la reconnaissance. La mienne va à Samy BENGIO et Marc SEBBAN, qui ont effectué ce lourd travail (160 pages $\times$ 5.6g/feuille $\approx$ 450g). Je les remercie également de m'avoir accueilli qui à l'IDIAP, qui à l'EURISE, afin de présenter mes résultats à leur équipe. Les questions posées et les discussions engagées ont nourrit ma réflexion et ont débouché sur un enrichissement certain de mon travail.

Je remercie Yves LECHEVALLIER pour avoir accepté d'être membre du jury et pour sa disponibilité. Je le remercie tout autant pour avoir su, en dépit de la courte durée d'une soutenance, émettre des remarques pertinentes et constructives qui alimenteront la suite de mes réflexions.

Le présent travail a été financé par France Télécom R&D et c'est au sein de l'équipe Apprentissage à Base de Connaissance puis de l'équipe Traitement Statistique de l'Information qu'il a été réalisé. Je remercie donc France Télécom R&D de m'avoir fait connaître les joies d'une réorganisation, notamment celle d'élargir le cercle des connaissances. Je ne peux toutes les citer nommément et me contenterai ici de préciser que j'ai énormément apprécié la qualité de l'accueil qui m'a été réservé par chacune d'entre elles.

Plus particulièrement, je remercie Fabrice CLÉROT, Françoise FESSANT, Carine HUE et Vincent LEMAIRE pour leur relecture attentive des ébauches de ce manuscrit, ainsi que Romain TRINQUART pour sa contribution à l'élaboration des supports visuels de la soutenance. Je remercie également très chaleureusement Christine BORDRON et Jocelyne BESNARD, secrétaires toujours efficaces et souriantes.

Un dernier mot afin d'exprimer mon éternelle reconnaissance et mon indéfectible attachement à mes parents, ma grand-mère, mon frère et ma famille. Le travail résumé dans ce manuscrit constitue aussi un aboutissement de leurs efforts.

Table des matières

Introduction générale

Il n'est plus rare aujourd'hui de voir s'accumuler les données au point d'atteindre des volumétries de l'ordre du tera-octet. La capacité de traitement de ces données constitue un goulot d'étranglement : songer par exemple que le nombre d'opérations par seconde effectuées par le processeur d'un ordinateur de bureau est de l'ordre du milliard et que la complexité d'un algorithme de traitement est rarement linéaire en la quantité de données. Si l'on peut simplement attendre (espérer ?) que les avancées technologiques rendent abordables des capacités de traitement supérieures, il est tout aussi naturel de chercher à conserver uniquement l'information utile.

Constatons également que, dans de nombreux cas, la récolte de données n'est pas sous-tendue par un besoin statistique mais a pour objectif l'automatisation de certains traitements. C'est dans un deuxième temps, une fois les données récoltées, que l'idée de leur analyse est soulevée et que sont appliquées des méthodes de réduction de l'information ou d'extraction de connaissance. C'est par exemple le cas pour un opérateur de télécommunication qui mesure et stocke un grand nombre de données relatives aux prestations qu'il fournit. C'est dans un deuxième temps que se pose la question de l'exploitation de ses données pour parfaire la connaissance des clients, du réseau, etc.

L'analyste se trouve alors dans la situation suivante. D'une part, il dispose d'un entrepôt de données déjà alimenté, souvent de grande taille. D'autre part, une question est soulevée par le propriétaire des données. Il doit alors transformer cette question en un problème d'analyse statistique de ces données et mettre en œuvre un processus de fouille de données. La caractéristique principale d'un processus de fouille est la répétition de sa mise en œuvre à partir d'un même entrepôt de données. Cela modifie les us et coutumes de l'analyse de données et nécessite de nouvelles méthodes et méthodologies.

Le travail ici reporté se positionne dans un contexte, celui de la préparation de données séquentielles dans un processus de fouille, s'inscrit dans une démarche, l'approche filtre univariée de la sélection de variables, et vise à faire sauter un verrou, celui de l'évaluation supervisée d'une variable séquentielle.

Contexte

Avec l'émergence des systèmes d'information au tournant des années 90, la récolte de données brutes a été rendue complètement indépendante de toute finalité statistique. Modéliser directement de telles données est devenu impossible. La phase de préparation des données, dont l'objectif est de construire à partir des données brutes une table de

données pour modélisation, est donc devenue une partie charnière et souvent coûteuse en temps d'un processus de fouille de données.

L'évolution des moyens techniques le permettant, une caractéristique est aujourd'hui suivie dans le temps. Avec un peu d'imagination, là où Sir Ronald Aylmer Fisher mesurait "manuellement" les dimensions des sépales et pétales de différents iris, il serait possible aujourd'hui de mesurer automatiquement ces mêmes quantités tout au long de la vie de chacun des iris, ce pour un grand nombre d'iris. A côté des variables usuelles, qu'on qualifie ici de *statiques*, sont donc de plus en plus présentes des variables *séquentielles*. Une telle variable associe à chaque individu une suite de mesures d'une même caractéristique : mesure de l'activité cardiaque en médecine, mesure de la pression en météorologie, mesure de la consommation téléphonique en télécommunication, mesure de la propagation des ondes en sismologie.

De par la décorrélation entre la récolte des données et la modélisation, la précision des mesures et l'échelle des temps de mesure sont sans rapport aucun avec les besoins d'une étude statistique. Seules les contraintes techniques sont limitatives. La préparation de variables séquentielles passe alors nécessairement par une phase de recherche de représentation : d'une représentation brute (les données observées) il faut passer à une représentation cohérente pour la modélisation subséquente, et ce pour chaque caractéristique suivie dans le temps. Au cours de cette recherche de représentation, d'autres problèmes que celui de l'échelle des temps de mesure sont à traiter, comme le bruit sur les mesures, le non alignement des temps de mesure, le facteur d'échelle entre les individus, etc.

Démarche classique

En phase de préparation d'un processus de fouille de données, une part importante du travail est consacrée à la construction et à la sélection des variables descriptives. Afin de réaliser cette tâche, une approche filtre univariée est souvent adoptée par l'analyste, au détriment de l'approche filtre multivariée ou de l'approche enveloppe [Guyon *et al.*, 2006]. Dans l'approche filtre univariée, le degré de pertinence statistique d'une variable est évalué et la variable est conservée si elle est jugée pertinente vis-à-vis de la question posée. Elle est préférée à l'approche filtre multivariée, dans laquelle les variables sont évaluées conjointement et non plus individuellement, cette dernière étant plus difficile à mettre en œuvre à mesure que le nombre de variables croît. L'approche enveloppe, en faisant intervenir le modèle, est quant à elle plus adaptée à la phase de modélisation qu'à la phase de préparation.

Pour mener à bien le travail de sélection dans une approche filtre univariée, il est nécessaire de disposer d'une méthode d'évaluation pour chaque format de variables. S'il existe de nombreuses méthodes pour les variables statiques [Dougherty *et al.*, 1995], [Kohavi and Sahami, 1996], ce n'est pas le cas pour les variables séquentielles.

En pratique, l'analyste est réduit à adopter la méthodologie suivante lorsqu'il doit traiter des variables séquentielles. Il remplace telle variable par un ensemble de variables statiques qu'il "sait" représentatives : moyenne, écart-type, etc. Dès lors, il dispose uniquement de variables statiques et peut adopter une approche filtre de la sélection. Cela

nécessite de recourir soit à une connaissance a priori, soit à un ensemble de validation.

Lorsque l'analyste est amené à répéter un processus de fouille, bien que les analyses exploitent un même entrepôt de données, il est risqué de capitaliser la connaissance. Chaque analyse répond à un objectif différent. La connaissance extraite dans un but particulier n'est pas nécessairement pertinente pour répondre à une autre question. Par exemple, il n'est pas certain que le taux de départ vers la concurrence (ou : *coefficient d'attrition*) soit expliqué par les mêmes variables que l'appétence à un nouveau service.

Le résultat d'une analyse statistique dépend fortement de l'ensemble des individus considéré par l'analyste. Il est reconnu que la qualité de ce résultat croît avec le nombre d'individus. Il est également reconnu qu'un compromis doit être établi avec la fiabilité du résultat. En effet, l'information extraite doit être valide sur des individus non encore considérés. Autrement dit, le résultat doit pouvoir être généralisé. En pratique, le compromis est établi à l'aide d'un ensemble d'individus n'ayant pas servi pour l'analyse, dit de *validation*.

Toute méthode assurant automatiquement la fiabilité de son résultat constitue un plus pour l'analyste. En se passant d'un ensemble de validation, il dispose de plus d'individus et est alors susceptible d'obtenir un meilleur résultat. De plus, une méthode qui ne serait pas dépendante d'une connaissance métier pourrait être utilisée quelle que soit la question posée. La difficulté est de proposer une telle méthode, générique, conjuguant finesse et fiabilité de l'information extraite et n'ayant pas recours à un ensemble de validation.

Contribution

Dans ce mémoire, nous présentons Eva. C'est une méthode d'évaluation fine et fiable qui permet d'inclure les variables séquentielles dans une approche filtre univariée de la préparation de variables. Ceci constitue une alternative à la pratique consistant à remplacer chaque variable séquentielle par un ensemble d'indicateurs statiques. Elle étend les résultats obtenus par Boullé, relatifs à l'évaluation supervisée de variables statiques [Boullé, 2005], [Boullé, 2006].

Eva répond à l'objectif suivant : disposer d'une méthode d'évaluation supervisée d'une variable séquentielle. Mais c'est en réalité une méthode plus générique. Les données séquentielles sont susceptibles de se présenter sous différentes formes et sont sujettes à un nécessaire travail de représentation. Constatant qu'il est toujours possible de définir une mesure de similitude entre les individus (la *matrice de Gram*) à partir d'une représentation, nous abstrayons la question de l'évaluation d'une variable séquentielle en un problème d'évaluation d'une mesure de similitude.

Nous résolvons ce problème en trois temps. Nous commençons par le formuler en un problème de recherche de la meilleure hypothèse dans un ensemble d'hypothèses que nous fixons : les diagrammes de Voronoi induits par les individus et la matrice de Gram. Puis nous proposons un critère d'évaluation supervisée de ces hypothèses. Enfin, nous développons un algorithme de recherche de la meilleure hypothèse. Nous donnons ainsi naissance à Eva, qui évalue la pertinence d'une mesure de similitude relativement à une variable cible catégorielle.

La méthode procède à une évaluation constructive (*c.f.* fig.1). En plus du calcul d'un

indicateur numérique de pertinence, qui s'interprète comme un gain de compression, une partition de l'ensemble des individus est réalisée. Cette construction permet d'envisager d'autres modes d'utilisation que la seule évaluation. Eva est ainsi applicable en tant que règle de sélection d'instances pour la classification par le plus proche voisin. Ou encore, elle produit une estimation des probabilités conditionnelles et autorise l'intégration des variables séquentielles directement dans un classifieur bayésien naïf. Ces différents modes d'utilisation profitent de la qualité de la méthode.

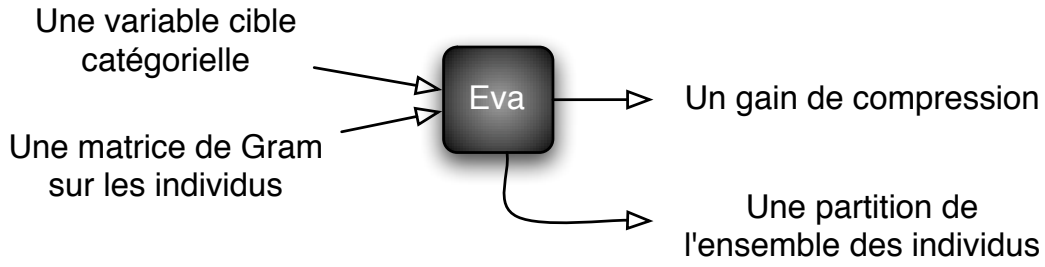

FIG. 1 – Eva, notre méthode d'évaluation.

Eva assure au résultat un niveau élevé de finesse et de fiabilité. La fiabilité provient du critère, qui résulte de l'adoption du principe de minimisation de la longueur de description. Il établit un compromis entre finesse et complexité de la connaissance extraite. La finesse résulte de la qualité combinée du critère et de l'algorithme. Le premier tient compte d'une grande quantité d'information et le second permet d'obtenir rapidement un bon résultat. Enfin, le critère est non paramétrique. La méthode se passe donc d'un ensemble de validation et permet d'intégrer un maximum d'individus dans l'analyse, ce qui augmente encore la finesse du résultat. Les caractéristiques de la méthode sont résumées à la fig.2.

- non paramétrique (1) : pas d'hypothèse paramétrique sur la forme de l'information extraite.
- fiable : l'information extraite est valide en général, pas uniquement sur les individus considérés en apprentissage.
- finesse : l'évaluation est fondée sur la détection de zones homogènes en les classes cibles.
- efficace : l'heuristique d'évaluation fournit rapidement un résultat de qualité.
- non paramétrique (2) : le critère d'évaluation ne nécessite pas l'ajustement d'un paramètre extérieur.

FIG. 2 – Caractéristiques de notre méthode Eva.

Organisation du document

Le mémoire se compose de trois parties. La première pose le contexte, décrit la problématique et oriente le travail vers la recherche d'une méthode d'évaluation d'une mesure de similitude. La seconde présente la solution nouvellement proposée, Eva, propose et étudie des alternatives. Enfin, la troisième illustre la manière dont notre travail résout la problématique initiale, par des expérimentations sur des données de consommation téléphonique.

Deux chapitres constituent la première partie. Dans le chapitre 1, nous décrivons un modèle de processus de fouille de données, le modèle CRISP-DM [Chapman *et al.*, 2000]. La phase de préparation des données d'un tel processus est alors mise en lumière. Nous discutons plus particulièrement la tâche de préparation des variables de la table. Puis nous décrivons l'approche filtre usuellement adoptée par l'analyste. Nous montrons que les variables séquentielles ne peuvent être traitées dans cette approche, faute d'une méthode d'évaluation automatique.

Dans le chapitre 2, nous montrons que les données séquentielles sont soumises à un travail de représentation et décrivons plusieurs méthodes de mise en forme. Ces données peuvent prendre différentes formes : numériques, catégorielles ou probabilistes. L'adaptation des modèles statistiques usuels, afin qu'ils puissent traiter des variables séquentielles en entrée, passe par la considération d'un produit scalaire ou, plus généralement d'une mesure de similitude. Nous présentons cet état de fait et nous l'établissons comme point de départ de notre travail : pour ne pas être dépendant de la représentation, c'est une mesure de similitude qu'il faut chercher à évaluer. Pour résoudre un problème d'évaluation d'une variable séquentielle nous nous proposons de résoudre un problème plus général.

La deuxième partie se consacre à la résolution de ce problème plus général. Elle se découpe en trois chapitres. Dans le chapitre 3, nous décrivons notre formulation du problème, nous proposons un critère d'évaluation et développons un algorithme d'optimisation de ce critère. Autrement dit, nous présentons Eva. Le critère d'évaluation obtenu réalise un compromis non paramétrique entre finesse et fiabilité.

Dans le chapitre 4, nous proposons d'autres critères réalisant un tel compromis. Nous les dérivons de deux approches différentes : la théorie de l'apprentissage statistique et l'inférence bayésienne. Nous montrons que le critère utilisé par Eva est celui assurant une finesse maximale sous contrainte de fiabilité.

Dans le chapitre 5, nous validons expérimentalement la méthode. Le parti pris est celui de la sélection d'instances pour la classification par le plus proche voisin. Nous montrons la qualité de la sélection opérée en comparant avec d'autres méthodes de l'état de l'art. Les expériences menées mettent en avant le degré élevé de fiabilité d'Eva. De plus, nous montrons une nouvelle fois en quoi l'évaluation réalisée est plus fine que celle des méthodes alternatives.

La troisième partie se compose de deux chapitres et revient au problème d'origine : l'évaluation supervisée d'une variable séquentielle. Le chapitre 6 constitue une mise en situation. Nous montrons empiriquement, à travers plusieurs expériences sur un jeu de données réelles de France Télécom contenant des variables séquentielles, les apports de

différents mode d'utilisation de notre méthode : évaluation supervisée, classification supervisée, estimation des probabilités conditionnelles.

Dans le dernier chapitre, le chapitre 7, nous présentons deux propositions de solution pour deux problèmes de représentation d'une variable séquentielle : la sélection supervisée de temps de mesure et le fenêtrage supervisé. A titre de perspectives avancées, nous proposons dans chacun de ces deux cas un critère d'évaluation et suggérons une heuristique d'optimisation. Le but est de montrer la souplesse de notre approche de l'évaluation et sa capacité à fournir des méthodes d'évaluation automatiques, fines, fiables, tout en restant dans le cadre des données séquentielles.

Première partie

Préparation de données et séquentialité

Introduction

Dans cette première partie, nous décrivons le contexte de nos travaux et la problématique que nous cherchons à résoudre. Nous précisons alors la direction prise par ces travaux.

Il existe différentes façons de mener une analyse de données, dépendantes du contexte de l'analyse. Dans le chapitre 1, nous distinguons le processus d'analyse statistique et le processus de fouille de données. Nous mettons ainsi en lumière les différences entre préparation de données pour analyse statistique et préparation de données dans un processus de fouille. Dans le deuxième cas, les variables descriptives sont construites et sélectionnées dans le but de capturer et mettre en forme l'information pertinente vis-à-vis de la question posée indépendamment d'une quelconque modélisation. Nous présentons l'approche filtre de la réalisation de cette tâche.

Nous définissons la notion de variable et proposons une taxonomie simplifiée. Nous précisons ainsi la notion de variable séquentielle. Nous montrons pourquoi de telles variables ne peuvent être filtrées directement et sont soumises à un travail d'extraction d'indicateurs qui, eux, peuvent être filtrés. Nous soulevons le besoin d'une évaluation automatique, seule à même d'éviter ce travail d'extraction et de permettre un traitement direct des variables séquentielles.

Avant la sélection, se présente un problème de représentation dont l'objectif est d'obtenir une représentation homogène des données séquentielles brutes. Dans le chapitre 2, nous montrons qu'il existe de nombreuses méthodes de construction d'une représentation pour les données séquentielles. Ceci renforce d'une part le besoin d'une évaluation automatique, et introduit d'autre part le besoin d'une évaluation générique, non dépendante d'une représentation particulière. Nous montrons que cette indépendance est obtenue, en pratique et en théorie, en définissant une mesure de similitude à partir de la représentation.

1

La préparation de données ou l'art de la mise en forme

Sommaire

Dans ce chapitre, nous explicitons le travail à effectuer en phase de préparation, sa finalité, et les contraintes le régissant. Pour cela, il est nécessaire dans un premier temps de replacer cette phase dans son contexte, celui d'un processus de fouille de données. C'est l'objet de la section 1.1. Dans la section 1.2, nous discutons l'objectif, les contraintes et la pratique de la préparation des variables. Nous définissons la notion de variable séquentielle et décrivons le traitement réservé à ce type de variables en préparation. Nous précisons en quoi nous considérons ce traitement peu satisfaisant. Nous en déduisons l'orientation du travail à effectuer.

1.1 Le processus de fouille de données

L'analyse de données a longtemps été l'apanage de la Statistique. Avec l'apparition des systèmes d'information, de nouvelles pratiques se sont développées. De nouveaux algorithmes ont vu le jour, afin de répondre aux besoins de l'Extraction de Connaissances dans les Bases de Données. Ces méthodes sont parfois rassemblées sous un chapeau "Fouille de données" pour mieux entrer dans une opposition avec la Statistique. Nous adoptons un point de vue plus pragmatique et proposons de qualifier une analyse suivant le contexte dans lequel elle est menée, afin de positionner clairement notre travail.

1.1.1 Analyse statistique et Fouille de Données

L'article [Hand, 1998] discute la relation entre Statistique et Fouille de Données (de l'anglais Data Mining), avec un brin de second degré. Cet article a pour objectif de porter à la connaissance de la communauté statistique les besoins d'une discipline qui, bien que s'attaquant aux mêmes problèmes, s'est développée en dehors de son sein.

La Fouille de Données pour un statisticien. – Du point de vue du statisticien, une analyse de données vise à séparer ce qui relève d'une structure sous-jacente de ce qui ne relève que de fluctuations aléatoires. La fouille de données, dont le but est d'identifier des motifs formés par les données, lui paraît une tentative naïve du fait que n'est pas pris en compte le caractère éventuellement aléatoire de leur apparition. Les volumétries mises en jeu le confortent dans son opinion, dans la mesure où plus on dispose de données, plus la présence de motifs aléatoires est probable. Pour le statisticien, le data-miner est donc quelqu'un qui a tendance à prendre des vessies pour des lanternes.

La Statistique pour un data-miner. – Du point de vue du data-miner, les outils classiques du statisticien ne répondent pas au besoin formulé. D'une part, avec l'augmentation de la volumétrie, tout devient significatif et les tests statistiques perdent de leur attrait. D'autre part, les méthodes statistiques trouvent leur justification dans des théorèmes de validité asymptotique et nécessitent de valider certaines hypothèses, difficiles à vérifier en pratique. Si les contraintes sur le développement d'une théorie solide et la faisabilité des calculs ont fortement guidé les avancées de la Statistique, l'avènement de l'ère numérique a changé la donne. Pour le data-miner, les statisticiens constituent donc une espèce en voie d'extinction.

Chercher à tracer un sillon entre Statistique et Fouille de Données résulte d'une intention louable, mais conduit parfois à creuser un fossé. Si les fossés sont de possibles sources d'amusement [Uderzo and Goscinny, 1990], nous adoptons un autre point de vue, en suivant celui de Hand. Plutôt que de définir la Statistique et la Fouille en se basant sur les outils employés, le fondement probabiliste ou la volumétrie et le type des données traitées, nous nous intéressons au contexte de l'analyse.

Processus d'analyse statistique. – Pour le statisticien, le phénomène analysé prime : une question étant soulevée, les données sont récoltées afin d'étudier cette question. La démarche, résumée par la fig.1.1, est la suivante :

1. Le phénomène à étudier est cerné, les indicateurs à mesurer et les individus sur lesquels les mesurer sont définis,
2. Les données sont récoltées,
3. Les données sont modélisées.

Préparation de données au cours d'une analyse statistique. – Dans un processus d'analyse statistique, les tâches de préparation de données ne sont pas regroupées et ne constituent pas une étape à part entière. Les tâches de définition des individus et des indicateurs sont incluses dans la première étape du processus. Le travail est alors plus un

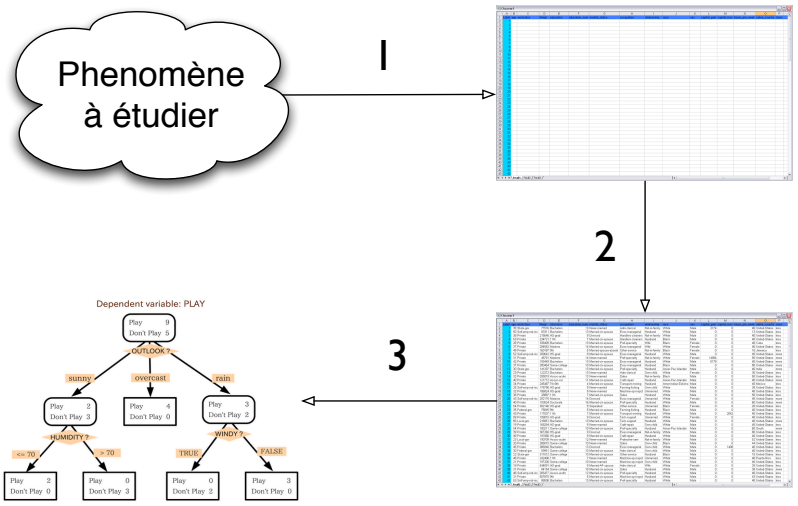

FIG. 1.1 – En Statistique, c'est le phénomène étudié qui prime. 1. Spécification de la table de données. 2. Récolte des données. 3. Modélisation.

travail de spécification que de préparation. Les tâches d'agrégation, de sélection d'individus et d'indicateurs, sont effectuées en phase de modélisation. Elles sont entièrement tournées vers l'amélioration de la performance des modèles.

La démarche statistique est une démarche hypothético-déductive. La spécification de la table repose sur l'hypothèse que les individus sont représentatifs et que les indicateurs sont susceptibles d'expliquer le phénomène étudié. La construction d'agrégats et de prototypes en phase de modélisation repose sur l'hypothèse que les modèles basés sur des données agrégées sont plus performants. La déduction repose donc sur la performance des modèles : une hypothèse est validée si la performance est améliorée.

Processus de fouille de données. – Le contexte dans lequel évolue le data-miner diffère en ce que les données vivent indépendamment de toute finalité statistique : leur analyse vient dans un deuxième temps. Le propriétaire de données récoltées dans un autre but que celui d'une quelconque analyse (pour de la facturation automatique, par exemple) pose une question précise à l'analyste (l'information contenue dans les données peut-elle être exploitée afin d'augmenter la connaissance client ?), celui-ci devant formuler la question en terme d'une analyse statistique puis mener cette analyse. La démarche, résumée par la fig.1.2, est la suivante :

1. L'architecture de l'entrepôt est spécifiée et l'entrepôt est alimenté,
2. Une question est posée,
3. Une table de données plate est préparée,
4. Les données sont modélisées.

Préparation de données dans un processus de fouille. – Dans un processus de fouille de données, l'analyste doit extraire et mettre en forme l'information contenue dans

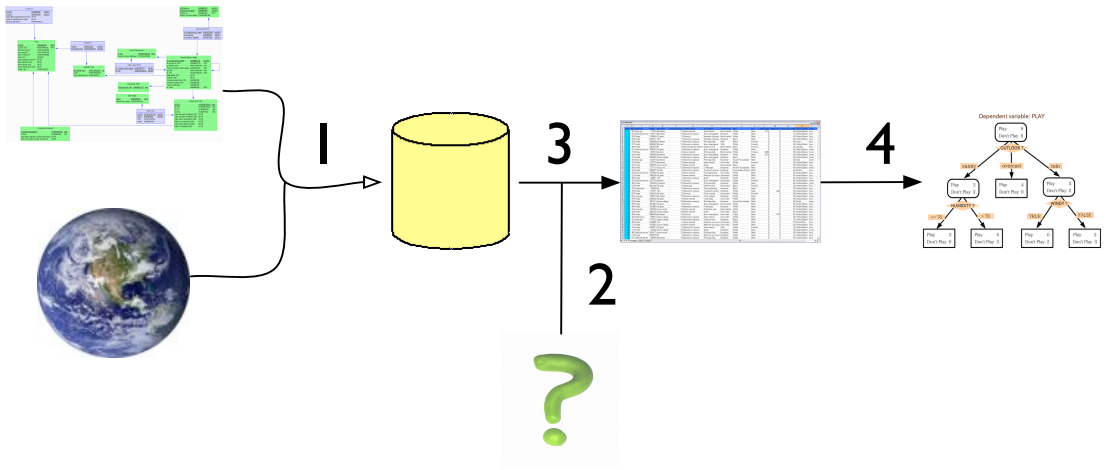

FIG. 1.2 – En Fouille de données, ce sont les données qui priment. 1. Spécification et alimentation de l'entrepôt. 2. Définition d'une question. 3. Préparation d'une table de données plate. 4. Modélisation.

l'entrepôt de données pour ensuite procéder à la modélisation. Cette phase d'extraction constitue l'étape de préparation des données. Elle précède la modélisation et ne repose pas sur l'amélioration de la performance des modèles. Les tâches à effectuer sont celles de spécification, de construction et de sélection d'individus et d'agrégats pertinents vis-à-vis de la question soulevée.

Face à un problème de fouille, la démarche du data-miner est successivement inductive puis déductive. La partie inductive est concentrée dans la phase de préparation, au cours de laquelle l'information contenue dans l'entrepôt de données est extraite et synthétisée. Une fois les données préparées, l'analyste se trouve confronté à une table de données plate. Il entre alors dans une phase déductive analogue à celle d'un processus d'analyse statistique, au cours de laquelle ses choix sont orientés par la performance des modèles.

Les caractéristiques de chacun des deux processus sont résumées dans le tab.1.1.

Analyse statistique	Fouille
- l'analyse est l'objectif premier	- l'analyse vient dans un second temps
- l'acquisition des données	- l'acquisition des données
fait partie de l'analyse	est indépendante de toute analyse
- démarche hypothético-déductive	- démarche inductive et déductive
Assurer la fiabilité des hypothèses validées	

TAB. 1.1 – Caractérisation des processus d'analyse statistique et de fouille (Hand).

1.1.2 CRISP-DM : un modèle de processus de fouille

Nous avons distingué deux processus d'analyse de données. Dans le premier, que nous qualifions de statistique, la récolte des données est effectuée après avoir défini le but de l'analyse et est donc orientée dans ce sens. Dans le second, un entrepôt de données est à disposition et son exploitation statistique vient seulement dans un second temps.

Un entrepôt peut donner lieu à plusieurs études, avec des buts à chaque fois différents, et un processus de fouille de données se caractérise par la répétition de sa mise en œuvre. C'est pourquoi différents intervenants industriels ont mis leurs connaissances en commun afin de standardiser le processus de fouille. Il en résulte un guide, nommé CRISP-DM pour CRoss Industry Standard Process for Data Mining [Chapman *et al.*, 2000], utile à toute entité souhaitant mener à bien un projet de fouille de données (*c.f.* fig.1.3).

Notons que d'autres modèles de processus existent, naturellement peu différents sur le fond mais prenant parfois d'autres formes. Là où le guide CRISP-DM parle de "fouille de données", et nous avec lui, le modèle décrit dans [Fayyad *et al.*, 1996] emploie les termes "extraction de connaissance à partir des données" et réserve celui de "fouille de données" pour ce que le guide CRISP-DM qualifie de "modélisation". Nous adoptons le guide CRISP-DM car celui-ci est plus complet et plus explicite. Nous le décrivons ici afin de fixer le vocabulaire et de positionner clairement notre travail.

Phases d'un processus de fouille. – Le modèle CRISP-DM propose de découper tout processus de fouille en 6 phases.

La première, d'*appréhension des besoins* (de l'anglais business understanding), fixe les objectifs industriels et les critères de succès, évalue les ressources, les contraintes et les hypothèses nécessaires à la réalisation des objectifs, traduit les objectifs et critères industriels en objectifs et critères techniques, et décrit un plan de résolution afin d'atteindre les objectifs techniques.

La seconde, de *compréhension des données* (de l'anglais data understanding), réalise la collecte initiale des données, en produit une description, étudie éventuellement quelques hypothèses à l'aide de visualisations et vérifie le niveau de qualité des données.

La troisième, de *préparation des données*, consiste en la construction d'une table de données pour modélisation. Nous nous y intéressons plus particulièrement dans la section suivante.

La quatrième phase, de *modélisation*, procède à la sélection de techniques de modélisation, met en place un protocole de test de la qualité des modèles obtenus, construit les modèles et les évalue selon le protocole de test.

Cette dernière tâche interfère avec la cinquième phase, d'*évaluation*, qui estime si les objectifs industriels ont été atteints, s'assure que le processus a bien suivi le déroulement escompté et détermine la phase suivante : retour en arrière ou *déploiement*.

Cette sixième phase propose un plan de déploiement du modèle, ainsi qu'un plan de contrôle et de maintenance, produit un rapport final et effectue une revue de projet.

Orthogonalement au découpage en phases, le guide CRISP-DM fournit une typologie des problèmes de fouille de données. En pratique, plusieurs types de problèmes sont à traiter dans un même projet de fouille.

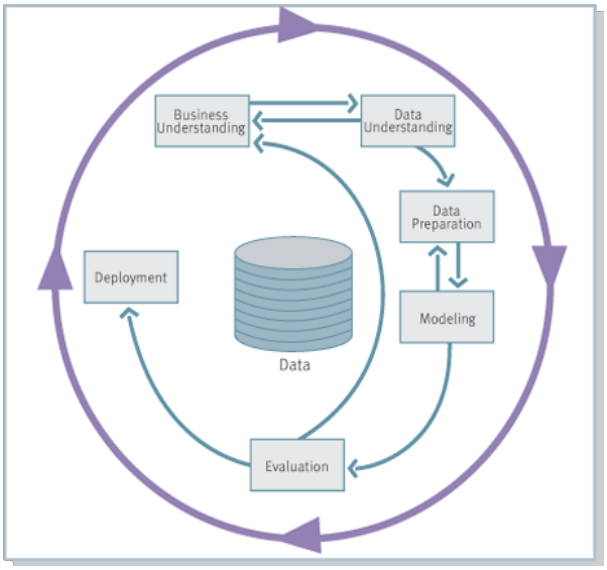

FIG. 1.3 – Modèle CRISP-DM : les 6 phases d'un processus de fouille.

Problèmes de fouille. – Six problèmes sont dégagés (*c.f.* fig.1.4).

Le premier cherche à produire une description concise des données, sous forme agrégée et/ou visuelle. Si résoudre ce type de problème peut constituer l'objectif d'un projet de fouille, il est plus fréquent de le voir apparaître comme un sous-objectif à atteindre durant les premières phases du projet.

Un problème de segmentation est posé dès lors qu'on cherche à constituer des groupes d'individus partageant certaines caractéristiques. Le but d'une segmentation est soit de détecter des groupes naturels, faciles à analyser, soit de réduire la volumétrie.

La description de concepts construit des règles de décision afin de caractériser de manière compréhensible certains comportements.

La classification travaille avec un ensemble d'étiquettes prédeterminé et cherche à étudier la relation entre les individus et les étiquettes. L'objectif technique d'un processus de fouille s'attaquant à un problème de classification est souvent de construire un *classifieur*, *i.e.* un procédé d'attribution d'une étiquette à un individu quelconque.

La prédiction est un problème analogue, où une mesure numérique vient remplacer la notion d'étiquette. On parle alors de *régresseur* en lieu et place d'un classifieur.

Enfin, l'analyse de dépendances s'intéresse à la description des dépendances entre facteurs. Cette analyse peut être probabiliste (corrélation, information mutuelle) ou déterministe (ensembles d'items).

1.1.3 Préparation de données dans un processus de fouille

Nous décrivons plus précisément la phase de préparation des données d'un processus de fouille de données. Cette dernière fait le lien entre les données à disposition et la phase de modélisation. Elle a pour objectif de construire une table de données plate pour la modélisation subséquente.

– Description, résumé des données,
– Segmentation (ou classification non supervisée),
– Description de concepts,
– Classification (ou classification supervisée),
– Prédiction (ou régression),
– Analyse des dépendances.

FIG. 1.4 – Typologie des problèmes de fouille.

Tâches de préparation. – Cinq tâches sont à remplir en préparation, d'après le modèle CRISP-DM (*c.f.* fig.1.5). La tâche d'intégration consiste à croiser l'information contenue dans différentes tables, ou d'autres sources, afin de créer les lignes et les colonnes de la future table. La tâche de construction vise à définir les individus, *i.e.* les unités sur lesquelles portent les mesures, et les variables, *i.e.* les caractéristiques mesurées sur les individus. L'évaluation de l'intérêt des individus et variables construits constitue le cœur de la tâche de sélection. La tâche de nettoyage a pour but de détecter et corriger les éventuelles anomalies survenues au cours de l'intégration des données et de traiter les valeurs manquantes. Lorsque les techniques de modélisation envisagées l'imposent, une tâche de formatage de la table de données est effectuée.

– Intégration,
– Construction,
– Sélection,
– Nettoyage,
– Formatage.

FIG. 1.5 – Tâches de préparation

Dans un processus d'analyse statistique, la définition des individus et des variables précède la récolte des données. Les tâches de sélection, de nettoyage et de formatage font quant à elles partie de la modélisation, les choix étant orientés par la performance prédictive du modèle. Dans un processus de fouille, ces tâches sont regroupées dans la phase de préparation et précèdent la modélisation.

Contraintes sur la préparation. – Le type d'information que l'analyste cherche à extraire et la forme qu'il peut lui donner sont, virtuellement, limités uniquement par son imagination. En pratique, l'extraction et la mise en forme sont soumises à deux contraintes : celle sur les ressources et celle sur le passage à l'échelle des méthodes de modélisation.

D'une part, dans un environnement industriel, le temps alloué à une étude est nécessairement limité. Les données brutes ne sont pas toujours accessibles facilement et

rapidement.

D'autre part, les algorithmes de modélisation sont rarement linéaires en le nombre de lignes et colonnes de la table. De manière plus insidieuse, l'analyste doit éviter de tomber dans le piège de la dimension, qui conduit à considérer un nombre de variables trop élevé pour le nombre d'individus à disposition. L'information portée par les individus est noyée dans un espace de représentation de taille inadaptée et de nombreuses techniques de modélisation ont les pires difficultés à produire un modèle pertinent.

Dans un processus de fouille, la phase de préparation est donc une étape d'extraction et de mise en forme de l'information contenue dans les données brutes, orientée dans le sens de la question soulevée. Une partie importante du travail consiste à construire des variables descriptives pertinentes relativement à cette question, sous contrainte de ressources et de capacité de traitement par les modèles.

1.2 Préparation des variables

Dans un processus de fouille, une table de données plate doit être élaborée pour modélisation à partir des données dont on dispose. C'est l'objet de la phase de préparation. Une grande partie de celle-ci est consacrée à la définition des variables de la table. La taille de l'ensemble des variables possibles est limitée uniquement par l'imagination de l'analyste, les contraintes techniques et les contraintes statistiques.

Nous introduisons une taxonomie adaptée à nos besoins, très restreinte, afin de qualifier sans ambiguïté les variables que nous manipulons dans ce document. Nous discutons la question de leur sélection, pour nous concentrer sur l'approche filtre univariée, souvent adoptée en pratique. Lorsque les variables sont structurées, comme c'est le cas des variables séquentielles, il est usuel de ne pas les traiter directement. Nous discutons plus en détail cet état de fait, ses limitations. Nous justifions alors la direction prise par notre travail.

1.2.1 Typologie des variables

Nous introduisons une taxonomie non exhaustive des variables.

Définitions. – Nous notons $\mathcal{I} = [\![1, N]\!]$ l'ensemble des N individus. Une *variable* est une fonction de $\mathcal{I}$ dans un ensemble $\mathbb{X}$. Nous appelons cet ensemble *espace de représentation* de la variable.

Définitions. – Soit $X : \mathcal{I} \to \mathbb{X}$ une variable. Si $\mathbb{X} = \mathbb{X}_1 \times \cdots \times \mathbb{X}_R$ avec $R \in \mathbb{N}^*$, X est dite *statique*. Elle est dite *multidimensionnelle* si $R > 1$ et *unidimensionnelle* sinon. Si les $\mathbb{X}_r$ sont des alphabets, X est dite *catégorielle*. Si les $\mathbb{X}_r$ sont des parties de $\mathbb{R}$, on dit que X est *numérique*.

Définitions. – Soit $X : \mathcal{I} \to \mathbb{X}$ une variable. Si $\mathbb{X}$ est un ensemble de suite

$$\{(t_r, x_r)_{1 \leq r \leq R};\ R \in \mathbb{N}^*, t_0 < \cdots < t_R \in \mathbb{R}, x_0, \ldots, x_R \in \mathbb{X}'\},$$

X est dite *séquentielle*. Autrement dit, X est séquentielle lorsqu'une suite de mesures dans un espace $\mathbb{X}'$ est associée à chaque individu. Comme pour les variables statiques, nous qualifions X de *multidimensionnelle* lorsque $\mathbb{X}'$ est un produit cartésien d'au moins deux ensembles et d'*unidimensionnelle* sinon. De manière analogue au cas statique, sont définies les notions de variable séquentielle catégorielle et numérique, suivant que $\mathbb{X}'$ est un produit cartésien d'alphabets ou de parties de $\mathbb{R}$.

Nous n'allons pas plus loin dans cette taxonomie, bien que ce soit possible. En l'état, celle-ci répond à nos besoins. Par commodité, nous introduisons de manière informelle les notions de type et de format.

Type et format d'une variable. – Le *type* d'une variable désigne son caractère numérique ou catégoriel. Le *format* d'une variable désigne son caractère statique ou séquentiel. Le Tableau 1.2 donne des exemples de réalisation de notre taxonomie restreinte.

	Format	
Type	Statique	Séquentiel
Catégoriel	catégorie socio-professionnelle	historique des services souscrits
Numérique	âge	historique de consommation électrique

TAB. 1.2 – Exemples d'instanciation de notre taxonomie simplifiée des variables.

1.2.2 Pratique de la construction de variables

Dans l'étape de préparation des données d'un processus de fouille, la recherche des variables capturant l'information pertinente est soumise à différentes contraintes.

Contraintes sur la préparation des variables. – Comme nous l'avons déjà vu, la prise en compte d'un nombre peu élevé de variables risque de manquer l'information et la considération d'un grand nombre de variables risque de noyer cette information. Un compromis doit être établi.

Il s'agit aussi de contrôler la taille de la représentation, pour une simple raison de complexité algorithmique : celle des méthodes de modélisation est rarement linéaire en le nombre de variables.

Enfin, il est préférable que le résultat du processus de construction dépende le moins possible d'hypothèses difficiles à vérifier, afin que l'information extraite soit celle contenue dans les données et non l'expression d'une connaissance extérieure à la question traitée, éventuellement inadaptée.

En phase de préparation des données d'un processus de fouille, construire une variable, c'est définir une succession de transformations à appliquer aux données brutes.

La question de la représentation. – A partir des données brutes, une variable $X : \mathcal{I} \rightarrow \mathbb{X}$ est construite en choisissant un espace de représentation $\mathbb{X}$ et en définissant un procédé

de mesure X projetant chaque individu dans $\mathbb{X}$. Dans un processus de fouille, ce procédé peut contenir un grand nombre d'étapes, faites de transformations successives. Pour les données statiques numériques, se pose par exemple la question de la normalisation : centrer puis réduire, ou normaliser les valeurs entre 0 et 1, ou passer en statistique de rang, etc. Pour les données séquentielles, les représentations possibles sont nombreuses, comme nous le verrons au chapitre 2.

Une fois la variable $X : \mathcal{I} \to \mathbb{X}$ obtenue, se pose la question de son intérêt relativement à la question étudiée. On entre alors dans le cadre de la sélection de variables. De nombreux articles ont tenté de cerner les pratiques de la sélection, notamment [Blum and Langley, 1997], [Kohavi and John, 1997] et [Guyon and Elisseeff, 2003]. Dans l'article [Kohavi and John, 1997], une distinction est opérée entre approche *enveloppe* (de l'anglais wrapper) et approche *filtre* (de l'anglais filter).

Approche enveloppe. – L'approche enveloppe consiste à évaluer l'impact d'une modification de l'ensemble de variables sur la performance d'un modèle. Une technique de modélisation étant spécifiée, elle est appliquée à différents ensembles de variables et l'ensemble de variables conduisant au modèle le plus performant est conservé. Chaque évaluation nécessite l'ajustement d'un modèle, ce qui se révèle coûteux en temps.

Cette approche, en faisant intervenir le modèle dans l'évaluation, est plus adaptée à la phase modélisation qu'à la phase de préparation d'un processus de fouille.

Approche filtre. – Dans l'approche filtre, l'intérêt des variables est estimé indépendamment d'une quelconque modélisation. Dans [Blum and Langley, 1997], il est proposé de construire les méthodes de filtrage en deux temps. Tout d'abord, l'utilisateur définit un critère de pertinence d'une variable ou d'un ensemble de variables puis il adopte une heuristique de sélection. Comme précisé dans [Guyon and Elisseeff, 2003], on parle d'évaluation *multivariée* lorsque le critère d'évaluation porte sur un ensemble de variables, et d'évaluation *univariée* lorsque chaque variable est évaluée individuellement.

L'approche filtre est naturellement adaptée à la phase de préparation d'un processus de fouille. En pratique, l'analyste adopte une approche filtre univariée plutôt que multivariée. Il définit un certain nombre de variables, les évalue individuellement et élimine celles déclarées non pertinentes. La préparation des variables est ainsi plus souple, plus facile à mettre en œuvre et à remettre en question. A contrario, l'approche multivariée est d'autant plus coûteuse que le nombre de variables croît.

Cette approche ne peut être adoptée que si l'analyste dispose d'une méthode d'évaluation de la pertinence pour chaque format de variable, non biaisée en fonction du format. Dans le cas statique, de nombreuses méthodes satisfaisantes existent (à base de tests statistiques ou exploitant des notions d'information mutuelle). Dans le cas séquentiel, non.

Traitement usuel des variables séquentielles. – En préparation, les variables séquentielles sont traitées de manière particulière. Pour chaque variable séquentielle, un ensemble de variables statiques est calculé à partir de la variable séquentielle. L'analyste ne considère ainsi plus que des variables statiques, qu'il peut traiter dans une approche filtre univariée. Par exemple, pour chaque individu, on calcule la moyenne des valeurs

séquentielles, la valeur minimale, la valeur maximale, le nombre de pics, etc, et la variable séquentielle est remplacée par ces variables statiques. Elle est "saucissonée".

Limites. – Tout d'abord, la question du biais est posée. Les variables statiques obtenues à partir d'une variable séquentielle sont plus élaborées que les variables statiques natives, donc susceptibles de contenir plus d'information. De plus, l'approche filtre univariée ne permet pas d'exploiter la dépendance statistique des indicateurs définis à partir d'une même variable séquentielle.

Ensuite, en extrayant certaines variables statiques plutôt que d'autres, l'analyste exploite une connaissance a priori susceptible d'être inadaptée au phénomène étudié. Il peut inversement procéder au calcul d'un grand nombre de variables statiques, sans a priori. Mais il se heurte alors à des contraintes algorithmiques et statistiques sur le nombre de variables à conserver, et à une contrainte sur le délai alloué à l'étude.

S'il est possible d'automatiser la partie algorithmique de l'extraction des indicateurs [Kaddous and Sammut, 2005], cela reste coûteux et ne résout pas la question de leur définition. Celle-ci fait toujours intervenir l'analyste, et repose sur une connaissance a priori non nécessairement pertinente pour le problème considéré.

1.3 Conclusion

Le processus de fouille de données se distingue du processus d'analyse statistique par le caractère secondaire des analyses à effectuer. Dans un processus de fouille, la phase de préparation des données est une étape charnière dont l'objectif est d'extraire de l'entrepôt de données l'information pertinente vis-à-vis de la question posée. Cette étape fait la transition entre des données brutes récoltées dans un autre but que celui de leur analyse et une table de données plate contenant l'information pour modélisation. La difficulté est d'assurer que seule de l'information pertinente et fiable a été extraite, et non des artefacts dépendant de l'échantillon.

La construction des variables descriptives capturant l'information constitue une part importante du travail de préparation. La recherche n'est pas limitée par l'imagination de l'analyste mais contrainte par les ressources disponibles et les limites algorithmiques et statistiques des techniques de modélisation envisagées. Une approche filtre univariée de la sélection des variables est souvent adoptée par l'analyste. Nous avons noté qu'une variable séquentielle ne peut être traitée dans cette approche, faute d'une méthode automatique d'évaluation. Une telle variable nécessite un traitement particulier que nous considérons peu satisfaisant.

Le travail décrit dans ce document se place dans le contexte de la préparation de données pour un problème de classification supervisée. L'objectif est de proposer une méthode d'évaluation supervisée d'une variable séquentielle (*c.f.* fig.1.6). La méthode, pour être utile, doit répondre à des contraintes d'automaticité, de finesse, de fiabilité et doit introduire un minimum de connaissance extérieure aux données. Plus précisément, une méthode automatique évite de perdre du temps et des données à ajuster un paramètre. Une méthode fine et fiable assure que l'information extraite est statistiquement valide et

généralisable à tout individu non encore observé. Enfin, en exploitant un minimum d'hypothèses statistiques, on assure que l'information obtenue vient des données et uniquement des données.

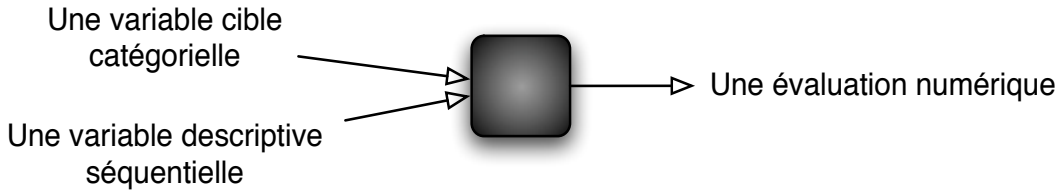

FIG. 1.6 – Evaluation recherchée : évaluation supervisée d'une variable séquentielle.

Avant d'établir une telle méthode, il nous faut cerner plus précisément certains aspects du traitement des données séquentielles et ainsi préciser le contexte de leur évaluation. C'est l'objet du chapitre 2.

2

Traitement de la séquentialité : vers la définition d'une mesure de similitude

Sommaire

Ce chapitre est construit suivant deux axes. Le premier est celui de la construction d'un espace de représentation pour des données séquentielles. Le second focalise sur les procédés permettant de rendre les méthodes traitant des variables séquentielles indépendantes de leur représentation.

Nous présentons plusieurs méthodes de construction d'un espace de représentation pour des données séquentielles dans la section 2.1. Dans la section 2.2, nous présentons différents procédés d'intégration des variables séquentielles dans les techniques classiques de modélisation.

2.1 Transformations pour représentation

L'objectif d'un travail de représentation est d'homogénéiser les données brutes. Nous présentons ici les outils disponibles pour ce travail dans le cas des données séquentielles. Nous distinguons les transformations *continues*, conduisant à une représentation numérique, les transformations *symboliques*, conduisant à une représentation catégorielle, et les transformations *probabilistes*, construisant un modèle probabiliste pour chaque individu.

2.1.1 Transformations continues

Les transformations continues reposent sur une intuition simple : à travers une suite de mesures effectuées sur un même individu, c'est une courbe qui est échantillonnée. Cette intuition est exploitée de deux manières : une formelle et une empirique. L'analyse de données fonctionnelles [Ramsay and Silverman, 1997] fournit un cadre formel, que nous introduisons afin d'unifier la présentation. Dans ce cadre, la représentation d'une suite de mesures devient un problème de projection sur un système orthonormé de fonctions. Nous présentons ce problème ainsi que 3 systèmes. Puis nous présentons des réalisations moins formelles de la même intuition, comme la segmentation ou le fenêtrage.

Projection sur un système orthonormé

L'analyse de données fonctionnelles se place dans le contexte suivant : une fonction réelle f_n définie sur un intervalle réel T est associée à chaque individu n. Cela revient à considérer une variable dont l'espace de représentation $\mathbb{X}$ est un ensemble de fonctions : $\mathbb{X} = \{f : T \to \mathbb{R}\}$.

Espace de Hilbert. – Soit H un ensemble. Si H est muni d'une structure d'espace vectoriel réel et d'un produit scalaire, il est dit *pré-hilbertien*. Si de plus toute suite de Cauchy d'éléments de H converge, *i.e.* si H est *complet*, H est dit de Hilbert.

Espace pré-hilbertien fonctionnel. – L'ensemble $\mathbb{X} = \{f : T \to \mathbb{R}\}$ est un espace vectoriel réel muni d'un produit scalaire $\langle \cdot, \cdot \rangle$ défini par

$$\forall f, g \in \mathbb{X}, \langle f, g \rangle = \int_T fg. \tag{2.1}$$

Autrement dit, c'est un espace pré-hilbertien. Notons que le sous-espace des fonctions intégrables est de Hilbert.

L'existence d'un produit scalaire assure l'existence d'une métrique euclidienne. La particularité des espaces fonctionnels est, en général, leur dimension infinie. Certains possèdent une base orthonormale.

Base orthonormale. – Soit H un espace de Hilbert. Une famille $(u_\alpha)_{\alpha \in A}$ est une *base orthonormale* de H si

$$\forall \alpha, \beta \in A, \langle u_\alpha, u_\beta \rangle = \begin{cases} 1 & \text{si } \alpha = \beta, \\ 0 & \text{si } \alpha \neq \beta. \end{cases} \tag{2.2}$$

et si l'ensemble des combinaisons linéaires finies d'éléments de $\{u_\alpha\}$ est dense dans H.

Décomposition de Fourier. – Si on se place dans l'espace de Hilbert des applications continues par morceaux et périodiques de période 2π, une base orthonormale classique est composée des fonctions $(\varphi_n(t) = \exp(\mathrm{i}nt))_{n \in \mathbb{Z}}$.

Toute fonction f périodique de période 2π s'écrit $f = \sum \lambda_n(f)\varphi_n$. Les coordonnées $\lambda_n(f)$ d'une fonction f dans cette base sont appelées *coefficients de Fourier* de f et s'écrivent $\lambda_n(f) = \langle f, \varphi_n \rangle$.

Cette base convient à la représentation de phénomènes périodiques dont on connaît la période. En pratique, la période d'un phénomène périodique n'est pas nécessairement 2π et doit être estimée.

Décomposition de Haar. – Pour traiter le cas non périodique, on dispose des fonctions suivantes, dites *de Haar*. Pour $j, k \in \mathbb{Z}$, la fonction de Haar d'indices j, k est définie par l'égalité

$$\psi_{jk}(x) = \mathbb{1}_{L_{jk}}(x) - \mathbb{1}_{R_{jk}}(x), \tag{2.3}$$

avec L_{jk} l'intervalle $\left]\frac{j}{2^k}, \frac{2j+1}{2^{k+1}}\right]$, R_{jk} l'intervalle $\left]\frac{2j+1}{2^{k+1}}, \frac{j+1}{2^k}\right]$. Les premières fonctions de Haar sont représentées sur la fig.2.1.

Toute fonction f réelle définie et intégrable sur un intervalle de $\mathbb{R}$ s'écrit $f = \sum \lambda_{jk}(f)\psi_{jk}$. Les coordonnées $\lambda_{jk}(f)$ d'une fonction f dans cette base sont appelées *coefficients de Haar* de f et s'écrivent $\lambda_{jk}(f) = \langle f, \psi_{jk}\rangle$.

Un intérêt de la famille de Haar est d'autoriser la représentation de fonctions non nécessairement périodiques, en capturant une information locale à l'aide d'éléments de base à supports compacts. Un reproche souvent formulé à l'égard de cette transformation est de fournir une approximation non continue. La théorie des ondelettes permet de construire des bases orthonormées composées d'éléments continus.

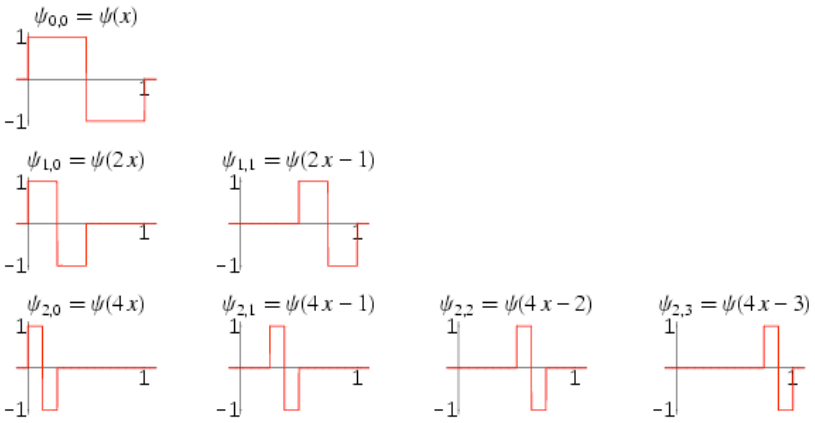

FIG. 2.1 – Premières fonctions de Haar.

Décomposition en B-splines. – Pour un ensemble de $m+1$ temps de mesure $t_0 \leq \cdots \leq t_m$, une *B-spline* de degré k est définie par la formule de récurrence

$$\begin{aligned} b_{j,0}(t) &= \mathbb{1}_{[t_j, t_{j+1}]}(t), \\ b_{j,k}(t) &= \frac{t - t_j}{t_{j+k} - t_j} b_{j,k-1}(t) + \frac{t_{j+k+1} - t}{t_{j+k+1} - t_{j+1}} b_{j+1,k-1}(t). \end{aligned} \tag{2.4}$$

Plus généralement, une fonction *spline* est une fonction polynômiale par morceaux sur les segments $[t_j, t_{j+1}]$. Toute fonction f réelle définie et intégrable sur un intervalle de $\mathbb{R}$

s'écrit $f = \sum \lambda_{jk}(f) b_{jk}$. Les coordonnées $\lambda_{jk}(f)$ d'une fonction f dans cette base s'écrivent $\lambda_{jk}(f) = \langle f, b_{jk} \rangle$.

Les fonctions de base B-splines sont des fonctions continues par morceaux qu'on manipule aisément, car polynômiales. De par cette souplesse d'utilisation, la famille des B-splines a largement été considérée ces dernières années.

De la théorie à la pratique. – Soit H un espace de Hilbert fonctionnel. Dans les trois cas présentés, on dispose d'une base orthonormée $(u_\alpha)_{\alpha \in A}$ permettant d'écrire tout élément f de H sous la forme $f = \sum_\alpha \lambda_\alpha(f) u_\alpha$, avec $\lambda_\alpha(f) = \langle f, u_\alpha \rangle$. Ainsi, pour une fonction f, on obtient une représentation homogène par l'intermédiaire de ses coefficients $\lambda_\alpha(f)$.

En pratique, deux choix interviennent : celui de la famille et celui du nombre de coefficients à conserver. De plus, les coefficients ne sont qu'approximés, du fait qu'on dispose uniquement d'un ensemble fini de mesures et non d'une fonction. Pour cela, l'intégrale définissant le produit scalaire est remplacée par une somme sur les temps de mesure.

Exemple d'utilisation. – Afin de diminuer les coûts de stockage et le temps de calcul pour la recherche d'un plus proche voisin, il a été envisagé d'appliquer dans une première étape la transformation de Fourier [Agrawal *et al.*, 1993a] ou celle de Haar [Chan and Fu, 1999] puis de calculer la distance sur les coefficients obtenus. Le choix du nombre de coefficients à conserver est alors central, réalisant un compromis entre qualité de l'estimation de la distance et coûts de stockage et de calcul.

Segmentation et fenêtrage

L'idée de voir une fonction à travers une suite de mesure a été exploitée de manière moins formelle. Par exemple, il est proposé dans [Pavlidis, 1974] de considérer une suite de segments définie à partir de la suite de mesures en lieu et place de la suite elle-même. Sur le plan formel, cela correspond à une représentation linéaire par morceaux. Diverses possibilités s'offrent alors : considérer des fonctions constantes, imposer la continuité, imposer de passer par les valeurs mesurées, etc.

Segmentation. – En général, quelles que soient les contraintes considérées, on cherche à représenter la série à l'aide d'un nombre minimal de segments. C'est ce qu'on entend par *segmentation*. Cela permet de réduire la dimension du problème et d'éliminer le bruit (*c.f.* fig.2.2).

Un compromis devant clairement être trouvé, deux questions se posent : quel critère doit-on optimiser et quel algorithme d'optimisation doit-on appliquer ?

Exemple d'application. – Dans [Keogh and Smyth, 1997], une fusion itérative des segments voisins est envisagée suivant les moindres carrés, sous contrainte puisque la représentation minimisant les moindres carrés est la représentation initiale. Les contraintes usuelles consistent à fixer le nombre final de segments et/ou la marge d'erreur. Il s'agit essentiellement d'éliminer les temps de mesures "inutiles". Une extension consiste à attribuer un poids à chaque segment, signifiant son importance vis-à-vis de la série, ce qui

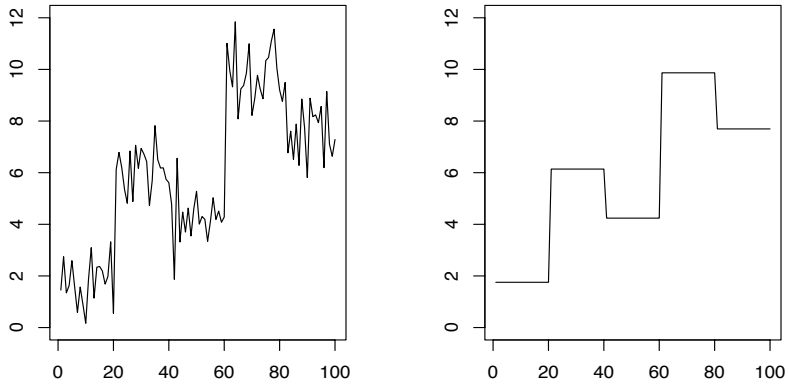

FIG. 2.2 – Exemple de segmentation d'une série.

permet de ne s'intéresser qu'aux segments les plus importants [Keogh and Pazzani, 1998]. Dans le même ordre d'idée, il est proposé dans [Guralnik and Srivastava, 1999] d'opérer une sélection incrémentale des points de changement de régime, par minimisation de la vraisemblance. Sur chaque intervalle, une régression linéaire est effectuée et c'est à partir de cette régression qu'est calculée la vraisemblance.

Fenêtrage. – La notion de *fenêtrage* repose sur le découpage des suites de mesures en sous-suites. Tronçonner les suites de mesures permet de travailler localement dans chaque fenêtre et indépendamment entre les fenêtres. Le fenêtrage consiste alors à faire passer la question de la représentation d'un niveau global à un niveau local.

Par exemple, c'est un procédé souvent employé pour traiter des suites de données très longues issues du domaine médical, comme les électro-cardiogrammes, les électro-encéphalogrammes, ou du domaine des réseaux, comme les log d'alarmes. La segmentation peut également être vue comme une technique de fenêtrage, un segment de droite étant construit sur chaque fenêtre.

Exemple d'application. – Dans [Gavrilov *et al.*, 2000], différents procédés de normalisation sont couplés avec un fenêtrage des séries. Par exemple, dans chaque fenêtre les mesures sont centrées et réduites ou bien rapportées à l'étendue. Une idée encore plus simple consiste à remplacer une suite de mesure dans une fenêtre par la moyenne de ces mesures.

Contraintes d'utilisation. – Que ce soit pour employer une segmentation ou un fenêtrage, des choix doivent être effectués par l'utilisateur et des paramètres doivent être ajustés. Par exemple, l'utilisateur doit sélectionner un type de représentation linéaire par morceaux, lorsqu'il envisage une segmentation, puis ajuster le nombre de segments. Lors-

qu'il emploie un fenêtrage, il s'agit de fixer le nombre de fenêtres et de poser ces fenêtres aux bons endroits. Puis il faut encore déterminer le travail à effectuer dans chaque fenêtre.

2.1.2 Transformations symboliques

En fouille de données séquentielles catégorielles, les algorithmes décrits dans [Agrawal *et al.*, 1993b] et [Agrawal and Srikant, 1995] permettent d'extraire de nombreux motifs séquentiels d'une base de données. Afin d'appliquer ces algorithmes également dans le cas de données numériques, les séries sont préliminairement projetées dans un espace de mots. Le lecteur intéressé trouvera dans [Hugueney, 2003] une synthèse des travaux sur la représentation catégorielle de données séquentielles.

Exemple d'applications. – Le cœur des méthodes est composé par l'alphabet descriptif choisi et la technique de projection. En adoptant une représentation segmentée, une première idée consiste à discrétiser l'ensemble des pentes et à associer un symbole à chaque intervalle. L'alphabet descriptif est appelé *alphabet de gradient* et est fixé par l'utilisateur [Agrawal *et al.*, 1995b], [Qu *et al.*, 1998].

Diverses autres méthodes de symbolisation ont été décrites, plus ou moins sophistiquées. Si on considère l'ensemble des sous-suites de longueur w donnée des séries étudiées, il est possible d'appliquer sur cet ensemble un algorithme de groupement comme celui des nuées dynamiques ou des cartes auto-organisatrices [Das *et al.*, 1998], [Giles *et al.*, 2001]. Un attrait de ce procédé réside dans l'automatisation de la définition de l'alphabet, un symbole étant associé à chaque groupe. Le choix du nombre de symboles n'en reste pas moins à la charge de l'utilisateur, dans la mesure où les méthodes de groupement employées travaillent à nombre de groupes fixé.

Contraintes d'utilisation. – Les transformations symboliques font appel à la connaissance métier de l'analyste, afin de définir le bon alphabet de gradient ou le bon fenêtrage. En renversant le point de vue, on peut dire que si l'analyste ne possède pas de connaissance a priori, il est face à de nombreuses possibilités, qui toutes nécessitent l'ajustement de paramètres sensibles.

2.1.3 Transformations probabilistes

Une approche probabiliste de la représentation de données séquentielles consiste à associer à chaque individu un modèle probabiliste. Nous présentons ici les méthodes à base de modèles markoviens à états cachés [Rabiner, 1989]. D'autres modèles existent, comme les modèles auto-régressifs [Gouriéroux and Monfort, 1995] que la méthodologie de Box et Jenkins a conduit à populariser [Box and Jenkins, 1994].

Les modèles markoviens sont particulièrement adapté à l'étude de systèmes adoptant successivement différents états $Q_0, \ldots, Q_t, \ldots$, avec émission d'un symbole O_t après chaque transition (*c.f.* fig.2.3). La sortie est une séquence de symboles, cette dernière étant la seule chose observable. Les états sont dits *cachés*. Lorsqu'on travaille sur des observations à valeurs numériques, on peut soit revenir au cas catégoriel en discrétisant l'ensemble des valeurs, soit considérer des modèles markoviens adaptés au cas numérique.

- $S = \{S_1, \ldots, S_n\}$ est l'ensemble des états du système,
- $V = \{v_1, \ldots, v_m\}$ est l'ensemble des émissions observables,
- $A \in \mathcal{M}_n(\mathbb{R})$ est la matrice des transitions, le coefficient a_{ij} définissant la probabilité de passer d'un état i à un état j (*i.e.* $P(Q_{t+1} = S_j / Q_t = S_i)$),
- $B \in \mathcal{M}_{n,m}(\mathbb{R})$ est la matrice d'émission, le coefficient b_{ik} définissant la probabilité d'émission du symbole v_k à l'état i (*i.e.* $P(v_k / Q_t = S_i)$)
- $\pi \in \mathcal{M}_{1,n}(\mathbb{R})$ est le vecteur d'état initial, π_i définissant la probabilité que Q_1 soit dans l'état S_i (*i.e.* $P(Q_1 = S_i)$).

FIG. 2.3 – Définition des paramètres d'un modèle markovien à états cachés.

- tirage de l'état initial q_1 dans S suivant π,
- $t = 1$,
- tirage de O_t dans V suivant la loi d'émission à partir de q_t,
- tirage de l'état q_{t+1} dans S suivant la loi de transition à partir de q_t,
- $t = t + 1$ et retour à la troisième étape si $t \leq T$.

FIG. 2.4 – Procédé de génération d'une séquence par un modèle markovien à états cachés.

Etant donnée la séquence observée $O = O_1 \ldots O_T$ d'éléments de V, un modèle markovien à états cachés peut être vu comme générateur de O, en accord avec la procédure décrite à la fig.2.4.

Propriété de Markov. – Le modèle est markovien dans le sens où la suite des états du système (Q_t) est telle que, pour tout T on a :

$$P(Q_T / Q_{T-1}, \ldots Q_1) = P(Q_T / Q_{T-1}). \tag{2.5}$$

Le succès des modèles markoviens réside dans l'existence d'algorithmes efficaces permettant de calculer la probabilité $P(O/\lambda)$ de génération de la suite O par le modèle λ (problème d'évaluation de la vraisemblance), de choisir la suite d'états $Q = Q_1 \ldots Q_T$ optimale (problème d'optimisation de la suite d'états) et de trouver le modèle λ maximisant la vraisemblance $P(O/\lambda)$ d'une suite d'observations O (problème d'optimisation du modèle).

Exemples d'utilisation. – Pour une suite de mesures, il existe différentes manières d'ajuster un modèle markovien. Par exemple, dans [Ge and Smyth, 2000], les auteurs procèdent à une segmentation et font correspondre un état caché émetteur d'une pente à chaque segment. Ils utilisent donc des modèles markoviens à émission numérique. Si on préfère les modèles à émission catégorielle, il suffit de définir un alphabet de gradient.

Contrainte d'utilisation. – La structure du modèle markovien (ses états cachés et ses probabilités de transition) est fixée à l'aide d'une connaissance a priori [Rabiner, 1989].

2.2 Intégration des variables séquentielles dans les modélisations

Les données séquentielles nécessitent un travail de transformation afin de définir une représentation homogène des données brutes. Nous avons vu qu'il existe un grand nombre de représentations possibles. De nombreux travaux ont été menés afin d'intégrer les variables séquentielles dans les techniques de modélisation classiques. Nous en décrivons ici une sélection. Nous montrons que l'emploi d'un produit scalaire ou d'une mesure de similitude rend l'intégration indépendante d'une représentation particulière.

2.2.1 Cas d'une représentation probabiliste

Dans le cadre probabiliste non supervisé, les techniques de représentation à base de modèles probabilistes individuels mènent assez facilement à une modélisation probabiliste de tous les individus. Nous présentons le cheminement suivi afin d'obtenir un modèle de mélange dans le cas des modèles markoviens.

Modèle de mélange. – L'objectif est d'estimer la loi $p(S = s)$ de la variable séquentielle S en supposant que les individus peuvent adopter K comportements (un "comportement" qualifiant une densité de probabilité). Autrement dit, on écrit :

$$p(S = s) = \sum_{k=1}^{K} \pi_k p_k(s), \tag{2.6}$$

où π_k est la probabilité a priori d'adopter le comportement k, et $p_k(s)$ la probabilité d'émission de s par le modèle.

Les questions alors soulevées portent sur le nombre de groupes, le type de modèle pour les groupes et l'estimation des modèles.

Exemple. – Dans [Smyth, 1997], les p_k sont des modèles markoviens à états cachés. L'apprentissage du modèle se décompose comme suit. Tout d'abord, un modèle markovien est estimé pour chaque individu. La matrice de log-vraisemblance est ensuite calculée (le coefficient (i, j) est le logarithme de la probabilité d'émission de la i^{eme} suite par le modèle de la j^{eme} suite), et symétrisée. Un algorithme de groupement est appliqué à partir de cette matrice afin de constituer K groupes. Le nombre de groupes K est obtenu par estimation du taux d'erreur en classification. Enfin, un modèle markovien est estimé dans chaque groupe.

La structure des modèles markoviens ajustés est supposée fixée a priori. Le travail exposé dans [Li, 2000] complète la procédure d'apprentissage de modèles de mélange markoviens en ajoutant une étape d'apprentissage de la structure du modèle dans chaque

groupe. L'heuristique de parcours de l'ensemble des topologies possibles pour le modèle est incrémentale : on commence par un modèle à un état, et le nombre d'états est augmenté tant que la valeur du critère croît. La recherche de la taille optimale des modèles dans l'algorithme global provoque une augmentation de la complexité algorithmique.

Extension. – Le formalisme probabiliste traitant les modèles de mélange a été étendu pour prendre en considération des variables statiques conjointement à une variable séquentielle. Suivant [Smyth, 1999], notons X la variable multidimensionnelle correspondant aux attributs statiques et S la variable séquentielle. Le modèle de mélange à estimer est alors de la forme

$$P(X = x, S = s) = \sum_{k=1}^{K} P_k(x, s)p_k = \sum_{k=1}^{K} P_k(x)P_k(s/x)p_k \tag{2.7}$$

avec K le nombre de groupes. Le choix du sens d'application de la formule de Bayes est guidé par l'intuition que le séquentiel est plus une conséquence du statique que l'inverse mais aussi par la simplicité de modélisation.

Les catégories de modèles paramétriques ayant été fixées pour X (loi gaussienne multidimensionnelle, par exemple) et S (modèle markovien, par exemple), une procédure de type EM est proposée. Notons que, que ce soit pour X ou S, le nombre de mesures est le même pour chaque individu. Dans le cas contraire, on trouve dans [Cadez *et al.*, 2000] une extension de cette méthodologie.

Le formalisme probabiliste des modèles markoviens, qui permet d'estimer indifféremment un modèle à partir d'un individu ou de plusieurs, est une clé permettant de modéliser une variable séquentielle. Indépendamment, le formalisme probabiliste des modèles de mélange est adapté à la prise en compte de modèles probabilistes de diverses origines, ce qui permet d'intégrer une variable séquentielle dans un modèle de mélange.

Contraintes. – Dans la réalité, il est plus fréquent d'avoir à intégrer plusieurs variables séquentielles. La méthodologie présentée n'apporte pas de réponse dans ce cas. De plus, l'approche probabiliste est lourde à mettre en œuvre, puisqu'il faut estimer puis agréger les modèles individuels, éventuellement adapter leur structure, ajuster des paramètres. C'est donc coûteux en temps. Enfin, le formalisme étant probabiliste, il est nécessaire d'adapter les méthodes ne travaillant pas dans ce cadre, comme les SVM.

2.2.2 Cas d'une représentation fonctionnelle

Lorsqu'on travaille avec une variable statique multidimensionnelle numérique, le cadre formel de l'analyse est l'espace euclidien. L'analyse de données fonctionnelles, en se plaçant dans un contexte hilbertien fonctionnel, permet une étude formelle des variables séquentielles. Les objets sont des fonctions, la dimension devient infinie et l'intégrale $\int$ remplace la somme $\sum$ dans la définition du produit scalaire.

Une fois dans le contexte hilbertien, toute méthode travaillant uniquement avec le produit scalaire, la norme ou la distance et n'effectuant que des manipulations algébriques dans l'espace de représentation admet techniquement une version fonctionnelle. Il est

à la charge de l'analyse de données fonctionnelles de montrer que la substitution est théoriquement valide.

Adaptation des modèles classiques. – Commençons par l'algorithme des nuées dynamiques (ou K-means en anglais) : la phase d'estimation nécessite la recherche du plus proche prototype et n'implique qu'un calcul de distances, la phase de maximisation nécessite le calcul des barycentres des groupes et n'implique que des manipulations algébriques. On trouve dans [Abraham *et al.*, 2003] l'adaptation des résultats de convergence de cet algorithme au cas hilbertien. On adapte de la même manière les algorithmes SOM et LVQ de Kohonen (voir par exemple [Rossi *et al.*, 2004]) et les modèles linéaires classiques en statistique [James, 2002].

Pour les perceptrons multi-couches, il suffit de constater qu'un neurone classique travaille avec une combinaison linéaire de ses entrées. La notion de neurone fonctionnel est définie en remplaçant le produit scalaire euclidien par un produit scalaire fonctionnel [Conan-Guez, 2003]. Dès lors, il suffit de placer dans la première couche ces neurones fonctionnels pour prendre en compte les variables séquentielles (un neurone fonctionnel par variable séquentielle).

Pour les méthodes ne travaillant qu'avec le produit scalaire, *i.e.* les méthodes à noyau, la transition est d'autant facilitée que le seul travail à effectuer consiste à définir un noyau. C'est le cas par exemple des séparateurs à vaste marge, des réseaux à fonctions à base radiale, des méthodes non paramétriques comme la classification par plus proche(s) voisin(s) ou les fenêtres de Parzen.

La plupart des méthodes classiques de modélisation admettent donc une variante fonctionnelle. C'est possible car les modèles travaillent avec un produit scalaire ou une mesure de similitude. Il reste à estimer une représentation fonctionnelle f_n pour chaque individu n. Ceci est effectué par projection sur des fonctions de base en nombre fini bien choisies $u_1, \ldots, u_R$. La projection consiste à écrire $f = \sum_r \lambda_r(f_n) u_r$ et il suffit d'estimer les coefficients $\lambda_r(f_n)$. Ceux-ci sont définis comme le produit scalaire $\langle f_n, u_r \rangle = \int f_n u_r$. En pratique, l'intégrale est approximée par une somme finie sur les mesures associées à l'individu n.

Si on considère des systèmes $u_1, \ldots, u_R$ orthonormés c'est pour se ramener complètement au cas euclidien. Si deux fonctions f et g admettent pour décomposition respective $\sum \lambda_r(f) u_r$ et $\sum \lambda_r(g) u_r$ dans un système orthonormé (u_r), le produit scalaire $\int fg$ de f et g est égal à $\sum \lambda_r(f)\lambda_r(g)$ et la distance entre f et g associée est égale à la racine de $\sum (\lambda_r(f) - \lambda_r(g))^2$.

Constat. – Sur le plan formel, la considération d'un produit scalaire et d'une distance euclidienne adaptés permet d'intégrer les variables séquentielles dans les modèles classiques. Cela n'élimine pas le problème de la représentation (choix d'une famille de fonctions de base, choix de leur nombre, approximation des coefficients). Mais cela permet de définir un procédé d'intégration des variables séquentielles dans les modèles sans se soucier de la représentation effective.

2.2.3 Exemples de mesures de similitude pour données séquentielles

L'utilisation d'une métrique hölderienne, comme la distance euclidienne, peut a priori paraître inadaptée dans le cas d'une variable séquentielle. En effet, ce type de distances ne tient pas compte du caractère séquentiel des mesures. Nous présentons différentes mesures de similitude adaptées aux données séquentielles.

Distance élastique. – Dans [Berndt and Clifford, 1996], une approche dynamique est proposée (*c.f.* fig.2.5). Soient $S = (s_1, \ldots, s_n)$ et $T = (t_1, \ldots, t_m)$ des suites de mesures à comparer. On suppose disposer d'une matrice $(\delta(s_i, t_j))_{ij}$ dont le coefficient (i, j) mesure la similitude entre la i^{eme} mesure de S et la j^{eme} mesure de T.

En considérant le graphe complet sur $\mathbb{N}_n^* \times \mathbb{N}_m^*$, le nœud (i, j) étant de poids $\delta(s_i, t_j)$, la similitude entre S et T est définie comme la longueur du chemin de longueur minimale sur ce graphe. Certaines contraintes sont imposées sur les chemins possibles. Par exemple, afin de respecter la séquentialité, tout retour en arrière est interdit.

L'inconvénient majeur est le coût de calcul de cette distance (dite DTW pour Dynamic Time Warping). Les contraintes permettent de réduire cette complexité. Sous certaines contraintes, l'existence de bornes rapidement calculables permet d'élaguer les recherches et de diminuer la complexité en moyenne [Yi *et al.*, 1998]. Il n'en reste pas moins que la méthode est alors paramétrique et dépendante des contraintes imposées.

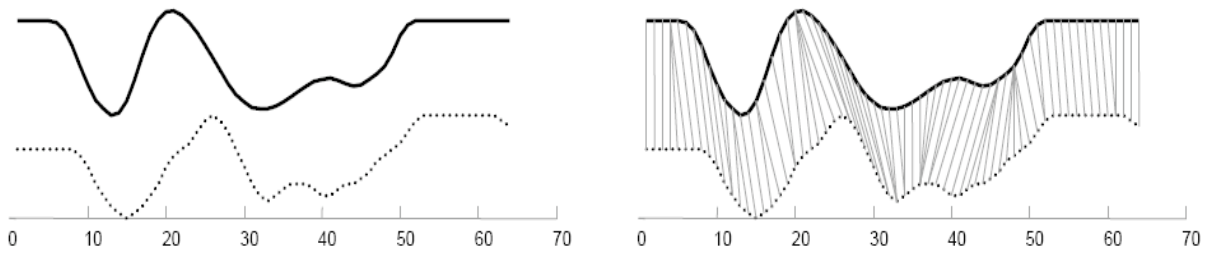

FIG. 2.5 – Mesure de distance élastique entre deux séries. Au lieu d'effectuer une comparaison temps de mesure par temps de mesure, on s'autorise à comparer des valeurs mesurées à des temps différents. Pour le bon fonctionnement de la méthode, des contraintes doivent être imposées.

Distance et segmentation. – Une autre approche permettant de diminuer la complexité de calcul de la distance élastique consiste à définir une distance entre deux segments (comme le carré de l'écart entre les milieux) et à traiter des données segmentées [Keogh and Pazzani, 1999]. Plus la segmentation est réductrice, plus les calculs sont réduits.

Si on travaille avec des séries A et B segmentées en K segments, chaque segment k étant pondéré par $w_A(k)$ et $w_B(k)$, une distance ad hoc est proposée dans [Keogh and

Pazzani, 1998] :

$$\delta(A,B) = \sum_{k=1}^{K} w_A(k) w_B(k) |(A(k-1) - B(k-1)) - (A(k) - B(k))| \tag{2.8}$$

$A(0), \ldots, A(K)$ et $B(0), \ldots, B(K)$ étant les valeurs mesurées.

Distance et fenêtrage. – Afin de gérer le bruit, les différences d'échelle et les translations, il est proposé dans [Agrawal *et al.*, 1995a] de comparer par paire les sous-suites de longueur fixée. Dans chaque fenêtre, la sous-suite est normalisée afin de prendre ses valeurs dans $[-1, 1]$ et on retient les paires de fenêtres similaires (dans un sens défini préalablement, avec prise en compte des différences d'échelle et des translations). Les paires qui ne se recouvrent pas peuvent être jointes, pour former la plus longue sous-suite "commune". La normalisation par fenêtre résout le problème de différence d'échelle et la recherche de paires de fenêtres similaires celui de la translation. Au final, si S et T sont les séries à comparer et si $(S_1, T_1), \ldots, (S_m, T_m)$ est la chaîne de sous-suites similaires la plus longue, la similitude de S et T est mesurée par :

$$\frac{1}{l(S) + l(T)} \sum_{i=1}^{m} l(S_i) + l(T_i), \tag{2.9}$$

où $l(\cdot)$ mesure la longueur d'une sous-suite.

La vraisemblance comme mesure de distances. – Lorsqu'on possède un modèle probabiliste λ_n pour chaque individu n (un modèle markovien à états cachés, par exemple), on peut calculer pour tout couple d'individus (n_1, n_2) la probabilité de génération par le modèle λ_{n_1} de la suite de mesures associée à n_2 (*i.e.* la vraisemblance de n_2 vis-à-vis de λ_{n_1}). Les vraisemblances donnent lieu à une interprétation en terme de mesure de distance : l'écart séparant deux individus est d'autant plus faible que la moyenne des deux vraisemblances est proche de 1. Il est alors techniquement possible d'appliquer toute méthode de modélisation basée sur une matrice de similitude, comme la classification par le plus proche voisin.

En partant de cette idée, il est proposé dans [Bicego *et al.*, 2003] d'apprendre un modèle markovien à états cachés par individu et à représenter chaque instance par le vecteur des vraisemblances relatives à chacun des modèles obtenus. Afin de diminuer la taille de l'espace de représentation, il est possible de ne modéliser qu'un certain nombre d'individus. On obtient ainsi un nouvel espace de représentation. Une mesure de distance comme la métrique euclidienne peut être employée et un modèle travaillant uniquement avec une matrice de distances peut être appliqué.

Constat. – De nombreuses mesures de similitude peuvent être définies et l'analyste doit avoir recours à une connaissance a priori pour faire son choix. De plus, nombreuses sont les situations dans lesquelles il doit ajuster des paramètres. Cela renforce le besoin d'une évaluation automatique de la pertinence d'une mesure de similitude.

2.3 Conclusion

La considération de données séquentielles soulève deux questions. La première est celle de leur représentation, afin de passer des données brutes à des variables séquentielles. Les possibilités de représentation sont nombreuses et variées, souvent paramétriques. Un grand nombre de variables séquentielles peuvent donc être définies à partir des mêmes données séquentielles. Face à cette combinatoire, une méthode d'évaluation fine et fiable est utile pour effectuer le bon choix.

La seconde question a trait à la nécessaire adaptation des procédés de modélisation pour que ceux-ci tiennent compte de la présence de variables séquentielles. Une réponse satisfaisante, car justifiée sur le plan formel et répandue en pratique, consiste à considérer un produit scalaire ou, plus généralement, une mesure de similitude (notion que nous définissons dans le chapitre suivant). C'est ce qui permet d'acquérir une certaine indépendance vis-à-vis de la représentation.

Pour notre part, nous cherchons à proposer une méthode d'évaluation supervisée d'une variable séquentielle. Nous voulons que cette méthode soit indépendante du mode de représentation de cette variable. Nous allons donc nous aussi considérer une mesure de similitude.

Hypothèse sous-jacente à notre travail. – Nous supposons que l'espace de représentation $\mathbb{X}$ est muni d'une mesure de similitude δ.

Notre nouvel objectif est de proposer une méthode d'évaluation automatique, fine et fiable d'une mesure de similitude dans le contexte de la classification supervisée (*c.f.* fig.2.6). C'est un problème plus général que celui envisagé au chapitre 1.

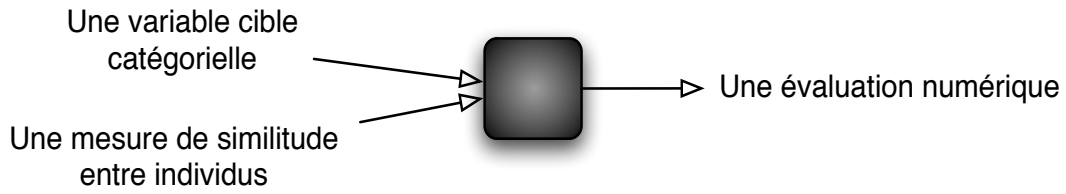

FIG. 2.6 – Evaluation maintenant recherchée : évaluation supervisée d'une mesure de similitude.

Conclusion

Dans cette première partie, nous avons mis en lumière l'étape de préparation des données d'un processus de fouille. L'objectif de cette étape est de mettre en forme l'information statistiquement pertinente vis-à-vis de la question traitée indépendamment d'une modélisation particulière. La tâche de préparation des variables constitue une part importante du travail à réaliser. Nous avons décrit l'approche filtre univariée et montré que les variables séquentielles ne peuvent être traitées directement par cette approche, faute d'une méthode d'évaluation automatique.

Pour les données statiques, la question de la représentation se limite essentiellement au choix d'une normalisation. Pour les données séquentielles, l'hétérogénéité sous laquelle se présentent les données brutes nécessite un travail conséquent de mise en forme. Comme nous l'avons discuté, de nombreuses méthodes de mise en forme existent, donnant lieu à différents types de représentation. Ceci renforce le besoin d'une méthode d'évaluation et soulève le besoin que celle-ci soit indépendante d'une représentation particulière.

La question de l'indépendance vis-à-vis de la représentation se pose également lorsqu'on cherche à adapter les techniques classiques de modélisation pour qu'elles tiennent compte des variables séquentielles. La solution adoptée, sur le plan pratique comme sur le plan formel, consiste à définir un produit scalaire ou, plus généralement, une mesure de similitude sur l'espace de représentation des individus. C'est ce qui permet de faire l'interface entre des variables séquentielles aux formes très différentes et des méthodes de modélisation taillées pour traiter des problèmes euclidiens ou métriques.

C'est pourquoi, dans un cadre supervisé, nous cherchons à résoudre un problème plus général que celui de l'évaluation d'une variable séquentielle, et orientons notre travail vers la recherche d'une méthode d'évaluation d'une mesure de similitude. La solution est susceptible d'être appliquée dans d'autres contextes. Comme nous l'avons souligné, la méthode doit être automatique, la plus fine possible et la plus fiable possible.

Nous quittons donc maintenant le domaine de l'analyse de données séquentielles, pour nous intéresser aux mesures de similitudes et à leur évaluation. Les variables séquentielles font leur retour dans la troisième partie.

Deuxième partie

Evaluation supervisée d'une mesure de similitude

Introduction

Nous présentons dans cette partie notre contribution principale : Eva, une méthode d'évaluation supervisée d'une mesure de similitude. Notre plan de recherche était le suivant :

1. formuler la question de l'évaluation d'une mesure de similitude en un problème de recherche de la meilleure hypothèse,
2. proposer un critère d'évaluation de ces hypothèses,
3. développer un algorithme de recherche de la meilleure hypothèse.

Nous décrivons Eva au chapitre 3. Nous précisons l'angle sous lequel juger l'intérêt d'une mesure de similitude, ce qui nous amène à définir un ensemble d'hypothèses : les partitions de Voronoi. Puis nous adoptons et adaptons une approche informationnelle de l'évaluation, le principe MDL, et proposons un critère d'évaluation de ces hypothèses. Enfin, nous développons et optimisons un algorithme de recherche de la meilleure partition encapsulant une heuristique gloutonne dans une méta-heuristique de recherche à voisinage variable.

D'autres approches de l'évaluation existent et nous les envisageons au chapitre 4. De la théorie de l'apprentissage statistique développée par Vapnik, nous déduisons un critère régularisant le risque empirique. Par analogie avec la mise en place du critère BIC de Schwartz, nous proposons une pénalisation heuristique de la vraisemblance. Le comportement de chacun des critères est comparé à celui du critère informationnel. Nous faisons ressortir les avantages de ce dernier, notamment le haut niveau de finesse du résultat auquel il conduit.

Au chapitre 5, la méthode est validée empiriquement, par des expériences sur des données réelles et synthétiques. Ceci permet d'illustrer les apports du critère et de l'heuristique, conjointement et séparément. Nous faisons une nouvelle fois ressortir les avantages de son utilisation, notamment le haut niveau de fiabilité du résultat auquel il conduit.

3

Eva, notre méthode d'évaluation

Sommaire

Dans ce chapitre, nous décrivons Eva, une nouvelle méthode d'évaluation supervisée d'une matrice de Gram.

A la section 3.1, nous fixons les notations et définissons les notions de mesure de similitude et de matrice de Gram. Nous introduisons les diagrammes de Voronoi et, plus particulièrement, les partitions de Voronoi induites par une matrice de Gram. Enfin, nous tournons l'évaluation supervisée d'une matrice de Gram en un problème de recherche de la partition de Voronoi la plus informative. Pour résoudre ce problème, il faut mettre en place un critère d'évaluation supervisée de l'information apportée par une telle partition et un algorithme de recherche de la meilleure partition.

Dans la section 3.2, nous présentons le principe de minimisation de la longueur de description, ou principe MDL (de l'anglais Minimum Description Length), et ses caractéristiques génériques. Nous décrivons ensuite un procédé d'instanciation idéal mais non praticable du principe, le critère n'étant pas calculable. Nous présentons l'inférence bayésienne comme une instanciation du principe conduisant à des critères calculables. Cette instanciation passe par la définition d'une distribution a priori et d'une fonction de vraisemblance. Nous adaptons les versions idéales et bayésiennes du principe MDL à notre contexte.

Nous proposons une distribution a priori et une vraisemblance dans la section 3.3. Un critère d'évaluation supervisée des partitions de Voronoi est ainsi obtenu. Dans la section 3.4, nous proposons un nouvel algorithme de recherche de la meilleure partition de Voronoi. Il consiste en l'application d'une heuristique gloutonne encapsulée dans une méta-heuristique de recherche à voisinage variable. Nous proposons des optimisations.

Le matériel contenu dans ce chapitre a fait l'objet de publications. L'article [Ferrandiz and Boullé, 2006d] est consacré au critère introduit à la section 3.3. L'algorithme présenté à la section 3.4 est décrit dans [Ferrandiz and Boullé, 2006c].

3.1 Hypothèses et notations

Nous commençons par poser quelques notations et définir les notions de mesure de similitude et de matrice de Gram. Puis nous introduisons les diagrammes et partitions de Voronoi [Preparata and Shamos, 1986]. Enfin, nous formulons le problème d'évaluation d'une matrice de Gram en un problème de recherche d'une partition informative.

Notations. – Soit $\mathcal{I} = [\![1, N]\!]$ l'ensemble des N *individus* et $\mathcal{L} = [\![1, J]\!]$ l'ensemble des J *étiquettes*. A tout individu $n \in \mathcal{I}$ est associé une étiquette $y_n \in \mathcal{L}$ et on note $Y : n \mapsto y_n$ la variable *cible* ainsi définie.

3.1.1 Noyau, mesure de similitude et matrice de Gram

Définition. – Soit $\mathbb{E}$ un ensemble. Toute application $\delta : \mathbb{E} \times \mathbb{E} \to \mathbb{R}$ est appelée un *noyau* sur $\mathbb{E}$. Si δ est à valeurs dans $\mathbb{R}_+$, on dit que δ est un noyau *positif*.

Nous considérons uniquement des noyaux positifs dans la suite. Tout au long de cette section nous continuons à le mentionner explicitement, pour être implicite ensuite.

Définition. – Soit δ un noyau positif sur un ensemble $\mathbb{E}$. Les deux premières propriétés suivantes sont dites respectivement de *séparation*, de *symétrie*, et la troisième est appelée *inégalité triangulaire* :

$$\begin{aligned} &1.\quad \forall x, y \in \mathbb{E}, \delta(x, y) = 0 \iff x = y, \\ &2.\quad \forall x, y \in \mathbb{E}, \delta(x, y) = \delta(y, x), \\ &3.\quad \forall x, y, z \in \mathbb{E}, \delta(x, z) \leq \delta(x, y) + \delta(y, z) \end{aligned} \tag{3.1}$$

Définition. – Soit δ un noyau positif sur un ensemble $\mathbb{E}$. On dit que δ est une *mesure de similitude* si δ vérifie la propriété de séparation. On dit que δ est *symétrique* si elle vérifie la propriété de symétrie. On dit que δ est une *métrique* si c'est une mesure de similitude symétrique vérifiant l'inégalité triangulaire.

Définition. – Soit δ un noyau positif sur $\mathcal{I}$. On dit que δ est une *matrice de Gram* sur $\mathcal{I}$ si δ est une mesure de similitude symétrique.

Symétrisation. – Soit δ un noyau positif sur un ensemble $\mathbb{E}$. Le noyau δ_s *symétrisé* de δ est défini par la relation

$$\forall x, y \in \mathbb{E}, \delta_s(x, y) = \frac{\delta(x, y) + \delta(y, x)}{2}. \tag{3.2}$$

Le noyau δ_s obtenu est positif et symétrique.

Il est donc toujours possible de définir un noyau positif symétrique à partir d'un noyau positif. En pratique, les noyaux considérés sont presque toujours symétriques. Nous supposons dans la suite, au moins implicitement, les noyaux symétriques.

Proposition 1 (Transfert de propriétés) *Soit $X : \mathcal{I} \to \mathbb{X}$ une variable. Lorsqu'on dispose d'un noyau $\delta_{\mathbb{X}}$ sur $\mathbb{X}$, on obtient un noyau $\delta_{\mathcal{I}}$ sur $\mathcal{I}$ en posant :*

$$\forall n_1, n_2 \in \mathcal{I}, \delta_{\mathcal{I}}(n_1, n_2) = \delta_{\mathbb{X}}(X(n_1), X(n_2)). \tag{3.3}$$

Si le noyau $\delta_{\mathbb{X}}$ est positif, $\delta_{\mathcal{I}}$ l'est également. Si $\delta_{\mathbb{X}}$ vérifie la propriété de symétrie ou l'inégalité triangulaire, alors $\delta_{\mathcal{I}}$ la vérifie également.

Abus de langage. – Le contexte dans lequel est située la proposition précédente est fréquent en analyse de données : une mesure de similitude est définie sur l'espace de représentation et l'analyste manipule en pratique une matrice de Gram sur les individus.

En effet, il dispose d'une variable $X : \mathcal{I} \to \mathbb{X}$ et il définit une mesure de similitude δ sur l'espace de représentation $\mathbb{X}$. Cette mesure induit un noyau positif sur $\mathcal{I}$ et ce noyau est symétrisé. Le cas des individus ayant même image par X est ensuite réglé, par exemple en résumant l'information porté par tels individus et en la faisant porter par un seul d'entre eux afin de ne conserver que ce dernier. L'analyste travaille finalement avec une matrice de Gram sur l'ensemble des individus réduit.

Nous utiliserons dans la suite les termes "matrice de Gram" et "mesure de similitude" de manière interchangeable. Notamment, nous parlerons "d'évaluation d'une mesure de similitude" pour désigner "l'évaluation de la matrice de Gram induite sur l'ensemble des individus par une mesure de similitude sur l'espace de représentation".

Métrique induite par un produit scalaire. – Soit $\mathbb{X}$ un espace vectoriel réel muni d'un produit scalaire euclidien $\langle \cdot, \cdot \rangle$. Le noyau $\delta_{\langle \cdot, \cdot \rangle}$ sur $\mathbb{X}$ défini par la relation

$$\forall x, y \in \mathbb{X}, \delta_{\langle \cdot, \cdot \rangle}(x, y) = \sqrt{\langle x - y, x - y \rangle}, \tag{3.4}$$

est une métrique, dite *euclidienne*, sur $\mathbb{X}$.

Les métriques euclidiennes généralisent la distance euclidienne du plan ou de l'espace. Les propriétés de cette dernière sont conservées et, pour cette raison, il est parfois souhaité travailler uniquement avec de telles métriques. Il s'agit alors d'établir des conditions assurant qu'un noyau est un produit scalaire. Nous en donnons ici deux, extraites de [Scholkopf and Smola, 2001].

Théorème de Mercer. – Soit un noyau symétrique δ sur un compact C de $\mathbb{R}^d$. Si l'opérateur associant à une fonction f la fonction

$$x \mapsto \int_C \delta(x, x')f(x')\mathrm{d}x' \tag{3.5}$$

est semi-défini positif, *i.e.* vérifie pour toute fonction f intégrable

$$\iint_{C\times C} \delta(x, x')f(x)f(x')\mathrm{d}x\mathrm{d}x' \geq 0, \tag{3.6}$$

alors il existe un espace vectoriel $\mathbb{X}$ muni d'un produit scalaire $\langle\cdot,\cdot\rangle$ et une projection $\varphi : C \to \mathbb{X}$ tels que $\delta(x, y) = \langle\varphi(x), \varphi(y)\rangle$ pour tous $x, y \in C$.

Proposition. – Soit δ une matrice de Gram sur $\mathcal{I}$. Si δ est une matrice semi-définie positive, alors il existe un espace vectoriel $\mathbb{X}$ muni d'un produit scalaire $\langle\cdot,\cdot\rangle$ et une variable $X : \mathcal{I} \to \mathbb{X}$ tels que $\delta(n_1, n_2) = \langle X(n_1), X(n_2)\rangle$ pour tous $n_1, n_2 \in \mathcal{I}$.

3.1.2 Diagrammes et partitions de Voronoi

Définition. – Soit δ un noyau sur un ensemble $\mathbb{E}$. Soit $H \subset \mathbb{E}$. Pour tout $x \in H$, on appelle *cellule de Voronoi* de x relativement à δ et H l'ensemble

$$V_\delta(x, H) = \left\{ y \in \mathbb{E};\ x = \arg\min_{x'\in H} \delta(y, x') \right\}. \tag{3.7}$$

L'élément x est appelé *prototype* de la cellule $V_\delta(x, H)$. Les éléments de $V_\delta(x, H)$ sont les éléments de $\mathbb{E}$ pour lesquels x est le plus proche élément de H relativement à δ.

Définition. – Soit δ un noyau sur $\mathbb{E}$. Soit $H \subset \mathbb{E}$. La famille $(V_\delta(x, H))_{x\in H}$ est appelée *diagramme de Voronoi* associé à H relativement à δ et on la note $V_\delta(H)$. Des exemples de tels diagrammes sont donnés à la fig.3.1.

Lorsque le noyau δ est une métrique euclidienne sur un espace vectoriel $\mathbb{E}$, les cellules de Voronoi d'un diagramme de Voronoi sont convexes. De plus, l'intersection de deux cellules de Voronoi est soit vide soit contenue dans un hyperplan de $\mathbb{E}$. Lorsqu'on relâche les hypothèses sur le noyau δ, le diagramme de Voronoi perd ces propriétés. Par exemple, si $\mathbb{E} = \mathbb{R}^d$ et δ est la métrique L_1 (ou : métrique de Manhattan), l'intersection de deux cellules de Voronoi n'est plus nécessairement incluse dans un hyperplan. Comme illustré sur la fig.3.2, elle peut être de mesure de Lebesgue non nulle.

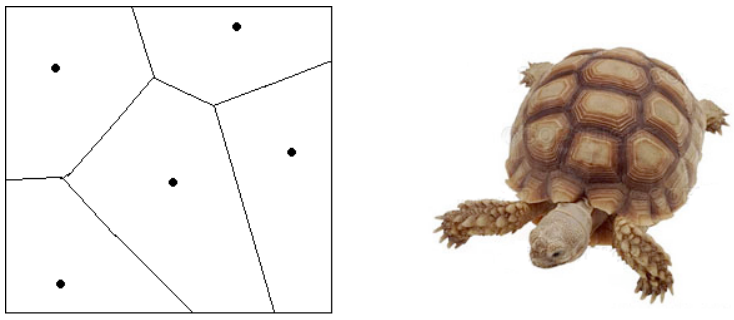

FIG. 3.1 – Exemple de diagrammes de Voronoi pour la métrique euclidienne.

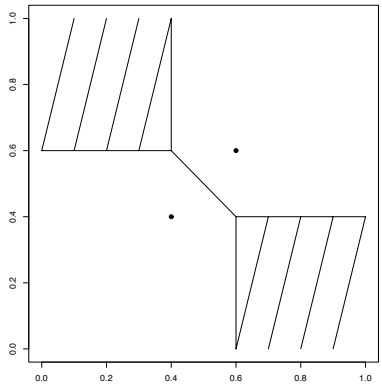

FIG. 3.2 – Configuration particulière de deux points pour laquelle l'intersection des cellules de Voronoi induites par la métrique L_1 est non négligeable au sens de Lebesgue. L'intersection des cellules est hachurée.

Traitement des égalités. – Soit δ un noyau sur $\mathcal{I}$. Si $H \subset \mathcal{I}$, nous souhaitons que la définition d'une cellule de Voronoi soit non ambigüe. Il est pour cela nécessaire de trancher les égalités de distances. Nous adoptons la solution proposée dans [Devroye *et al.*, 1996]. L'idée est d'ajouter une variable de distribution uniforme sur l'intervalle réel $[0, 1]$ et de la faire intervenir dans le calcul de la similitude uniquement en cas d'égalité.

A chaque individu $n \in \mathcal{I}$ est associée une valeur $r(n)$ tirée aléatoirement et uniformément dans l'intervalle réel $[0, 1]$. Si deux individus n_1 et n_2 sont à égale distance de n (*i.e.* $\delta(n, n_1) = \delta(n, n_2)$), l'égalité est tranchée en comparant les valeurs $|r(n) - r(n_1)|$ et $|r(n) - r(n_2)|$: si $|r(n) - r(n_1)| < |r(n) - r(n_2)|$, n_1 est déclaré plus proche de n que n_2, si $|r(n) - r(n_1)| > |r(n) - r(n_2)|$, n_2 est déclaré plus proche de n que n_1. En pratique, on est dans l'une ou l'autre des situations, presque sûrement.

Proposition 2 *Soit δ un noyau positif sur $\mathcal{I}$. Soit $H \subset \mathcal{I}$. Si δ vérifie la propriété de séparation et si on tranche les égalités de distance, alors le diagramme de Voronoi $V_\delta(H)$ est une partition de $\mathcal{I}$.*

Preuve. – Il faut montrer que $\mathcal{I}$ est union disjointe des cellules de Voronoi $V_\delta(x, H)$ et que chaque cellule est non vide.

Comme H est fini, tout individu admet un plus proche prototype et l'ensemble des individus est bien l'union des cellules de Voronoi. Cette union est disjointe du fait que les égalités sont tranchées : pour tout individu, le plus proche prototype est défini de manière unique. Enfin, les cellules sont non vides, car tout prototype appartient à sa propre cellule du fait que le noyau est supposé positif et vérifie la propriété de séparation. ■

A partir de maintenant, lorsque nous parlons de partition de Voronoi c'est que nous nous plaçons implicitement sous les hypothèses de la proposition précédente : le noyau sur $\mathcal{I}$ vérifie la propriété de séparation et le procédé de traitement des égalités est adopté. De plus, quite à symétriser le noyau, nous pouvons considérer que δ est une matrice de Gram sur $\mathcal{I}$.

Notations. – Soit δ une matrice de Gram sur $\mathcal{I}$. On note $\mathcal{H}_\delta(\mathcal{I})$ l'ensemble $\{V_\delta(H);\ H \subset \mathcal{I}\}$ des partitions de Voronoi associées à un sous-ensemble de l'ensemble des individus, relativement à δ.

L'ensemble $\mathcal{H}_\delta(\mathcal{I})$ est de cardinal fini, inférieur à 2^N. Ses éléments dépendent uniquement de l'ensemble des individus $\mathcal{I}$ et de la matrice de Gram δ. Ils vont de la partition en autant de cellules que d'individus à la partition constituée par une unique cellule contenant tous les individus.

Notations. – Soit δ une matrice de Gram sur $\mathcal{I}$. Soit $H \subset \mathcal{I}$. Pour tout $k \in H$, le nombre d'individus appartenant à la cellule $V_\delta(k, H)$ est noté N_k et le nombre de tels individus portant la j^{eme} étiquette est noté N_{kj} $(1 \leq j \leq J)$. Ainsi, $N = N_1 + \cdots + N_K$ et $N_k = N_{k1} + \cdots + N_{kJ}$. Nous notons Y_k la restriction de la variable cible Y à la cellule $V_\delta(k, H)$: $Y_k = Y_{|V_\delta(k,H)}$. Bien que ces quantités dépendent de δ et H, nous ne marquons pas cette dépendance pour ne pas alourdir les notations.

3.1.3 Vers la recherche de la partition la plus informative

Soit δ une matrice de Gram sur $\mathcal{I}$, dont on cherche à évaluer la pertinence vis-à-vis de la variable cible Y. Il s'agit de quantifier cette pertinence de manière statistique, c'est-à-dire à partir des données dont on dispose et uniquement à partir de ces données : $\mathcal{I}$ et Y. Nous sommes dans une situation analogue à celle de la construction d'un test statistique, lorsqu'un ensemble d'hypothèses doit être formulé à partir de la question posée.

Définitions et notations. – Nous appelons *hypothèse* tout élément de $\mathcal{H}_\delta(\mathcal{I})$. Nous notons H_0 la partition de Voronoi constituée par une unique cellule, et nous la qualifions d'*hypothèse nulle*. Nous notons H_{sat} la partition de Voronoi constituée par N cellules et la qualifions d'hypothèse *saturée*.

L'idée sous-jacente à cette définition est la suivante : δ est d'autant plus pertinente qu'elle permet d'isoler des zones dans lesquelles la distribution des fréquences de la cible est différente. Dans ce sens, accepter l'hypothèse nulle et ne faire qu'un seul groupe, c'est

dire que la répartition des individus induite par δ ne fait apparaître aucune différence d'homogénéité de la variable cible. Poursuivant notre analogie avec la mise en place d'un test statistique, nous serions amenés à tester une hypothèse simple, H_0, contre une hypothèse composite, $\mathcal{H}_\delta(\mathcal{I})\backslash\{H_0\}$. Dès lors, nous pourrions éventuellement mettre en place un test du rapport des vraisemblances maximales [Saporta, 1990]. Mais ceci est impossible, pour les raisons suivantes.

Particularités. – Les hypothèses que nous considérons possèdent les caractéristiques particulières suivantes :
- les hypothèses ne sont pas des lois de probabilité,
- les hypothèses dépendent des données.

Nous avons formulé le problème d'évaluation supervisée de la pertinence d'une matrice de Gram en un problème de recherche d'une hypothèse. Les hypothèses adoptées, les partitions de Voronoi de l'ensemble d'individus $\mathcal{I}$, sortent du cadre probabiliste des tests statistiques. L'analogie avec la mise en place d'un test statistique s'arrête donc ici.

Objectif. – Nous devons définir un critère d'évaluation supervisée $c_{\delta,Y} : \mathcal{H}_\delta(\mathcal{I}) \to \mathbb{R}$, jouant le rôle d'une statistique, afin de sélectionner la meilleure hypothèse au sens de ce critère :

$$H_{opt} = \underset{H\in\mathcal{H}_\delta(\mathcal{I})}{\arg\min}\, c_{\delta,Y}(H). \tag{3.8}$$

Dès lors, $c^*_{\delta,Y} := c_{\delta,Y}(H_{opt})$ quantifie la pertinence de la matrice de Gram δ relativement à la variable cible Y.

L'ensemble des hypothèses considéré est très riche bien que fini. A une extrêmité du spectre des possibilités, l'hypothèse saturée correspond à un apprentissage par cœur des données. C'est le point de vue sur les données le plus informatif possible. A l'autre extrêmité, l'hypothèse nulle correspond à l'absence d'apprentissage. C'est le point de vue sur les données le plus grossier. Entre les deux, toute partition de Voronoi autorise un point de vue différent sur la distribution des étiquettes. Ce point de vue est d'autant plus fin que la partition est composée d'un grand nombre de cellules. Mais plus il est proche des données, plus il est susceptible d'être inadapté sur de nouvelles données et moins il est fiable. Le critère doit quantifier les notions de finesse et fiabilité puis établir un compromis.

Exigences. – De notre point de vue, le critère doit

1. dépendre uniquement de $\mathcal{I}$, δ et Y,
2. être non paramétrique.

Le respect du premier point évite d'avoir à dépendre d'hypothèses difficiles à valider en pratique et évite de faire appel à de la connaissance a priori ou à un ensemble de validation. Le respect du deuxième point évite d'avoir à ajuster un paramètre, ce qui nécessiterait d'évaluer l'évaluation et provoquerait une mise en abîme. Plus prosaïquement, cela évite également de faire appel à de la connaissance a priori ou à un ensemble de validation.

Nous considérons trois approches de l'évaluation produisant des critères d'évaluation non paramétriques et réalisant un compromis entre finesse et fiabilité. Nous présentons dans ce chapitre le critère obtenu en adaptant à notre problème le principe de minimisation de la longueur de description.

Remarque. – D'autres hypothèses que des partitions de Voronoi peuvent être considérées. Dans [Ferrandiz and Boullé, 2005a] et [Ferrandiz and Boullé, 2005b], nous avons proposé de partitionner l'ensemble des individus en exploitant des graphes de voisinage [Jaromczyk and Toussaint, 1992], [Ferrandiz and Boullé, 2004]. Ceux-ci sont définis sur $\mathcal{I}$ à l'aide d'une mesure de similitude δ. Une fois le graphe construit sur les individus, nous définissons un ensemble de partitions dépendant du graphe obtenu.

L'algorithme de construction de la plupart de ces graphes est de complexité cubique en le nombre d'individus. De plus, on laisse un choix à l'utilisateur, celui de la structure de graphe (graphe de Gabriel, graphe de voisinage relatif, etc) , qui ne peut être résolu que par une connaissance a priori ou en faisant appel à un ensemble de validation. L'ensemble des partitions de Voronoi est quant à lui défini uniquement par les individus et la matrice de Gram, et ne repose sur aucune construction préliminaire. Cet ensemble de partitions est à la fois plus simple à manipuler et plus générique.

3.2 Evaluation par minimisation de la longueur de description

3.2.1 Enoncé du principe

Un nouveau paradigme d'inférence par induction a progressivement émergé des travaux indépendants de Wallace [Wallace and Boulton, 1968] et Rissanen [Rissanen, 1978], anticipés par Solomonoff [Solomonoff, 1964]. L'acception générique du principe de minimisation de la longueur de description, qu'on trouve par exemple dans [Li and Vitanyi, 1997], est la suivante.

Principe MDL [Li and Vitanyi, 1997]. – Si on dispose d'un ensemble de données D et d'un ensemble d'hypothèses $\mathcal{H}$ relatives à ces données, la meilleure hypothèse est celle minimisant la somme des
- longueur de description de l'hypothèse et
- longueur de description des données à partir de l'hypothèse.

Cet énoncé est générique. Pour obtenir un critère MDL, il faut définir ce qu'on entend par "description" et par "longueur de description". L'instanciation du principe MDL nécessite donc de se poser un certain nombre de questions (*c.f.* fig.3.3).

On trouve dans la littérature de nombreuses réponses à ces questions. Nous ne procédons pas ici à un inventaire, dans la mesure où il existe plusieurs documents faisant la synthèse de certains points de vue (voir entre autres [Grünwald *et al.*, 2005], [Oliver and Hand, 1994], [Oliver and Baxter, 1994], [Baxter and Oliver, 1994], [Gammerman and Vovk, 1999], [Hansen and Yu, 2001], [Lanterman, 2001]). Nous relevons, entre autres, deux

1. Quel ensemble d'hypothèses $\mathcal{H}$?
2. Qu'entend-on par description ?
3. Quelle description effective d'une hypothèse ?
4. Quelle description effective des données à partir d'une hypothèse ?
5. Qu'entend-on par longueur de description ?
6. Quelle longueur de description effective des hypothèses ?
7. Quelle longueur de description effective des données à partir d'une hypothèse ?
8. Quelle méthode de recherche de la meilleure hypothèse ?

FIG. 3.3 – Instancier le principe MDL, c'est répondre à ces questions pour obtenir un critère d'évaluation.

choses. D'une part, les travaux de Rissanen et Wallace se sont concentrés sur la question de l'estimation de densités, avec un penchant pour des hypothèses paramétriques. Autrement dit, les travaux sur le principe MDL se sont placés pour la plupart dans le cadre statistique probabiliste classique. D'autre part, les travaux de Rissanen se sont distingués de ceux de Wallace avec le temps, en s'orientant vers la sélection de modèles (traitant, par exemple, le problème du choix du nombre de densités à mélanger dans un modèle de mélange).

Il existe aujourd'hui diverses instanciations du principe MDL générique. Et le parcours de la littérature sur le sujet permet de constater que beaucoup se placent dans le contexte classique de la statistique, celui de l'estimation de densités. Au final, nous retenons seulement que, du point de vue de Grünwald : "all in all, current versions of MDL that avoid probabilistic assumptions are still in their infancy" [Grünwald *et al.*, 2005].

Pour notre part, nous pensons que la force de l'approche informationnelle est de pouvoir traiter tout type d'hypothèses, et pas seulement les hypothèses statistiques paramétriques classiques. Pour qui applique le principe informationnel, une hypothèse est simplement un objet à décrire. Les hypothèses considérées peuvent être des densités de probabilité, mais pas uniquement. Ont été considérés, par exemple, les arbres de décision [Quinlan and Rivest, 1989] ou les ensembles d'items [Siebes *et al.*, 2006].

3.2.2 Caractéristiques génériques

Avant de s'intéresser à l'obtention d'un critère MDL, nous présentons les propriétés génériques de l'approche [Grünwald *et al.*, 2005].

Le rasoir d'Occam. – L'application du principe MDL conduit à satisfaire le rasoir d'Occam. Ce dernier est un principe de raisonnement considérant, dans sa version originale, qu'il est souhaitable de ne pas valider un nombre d'hypothèses dépassant toute

nécessité. Avec le temps, l'assertion s'est transformée et propose de choisir la plus simple entre plusieurs hypothèses égales par ailleurs. La nouveauté apportée par le principe MDL réside dans le concept de longueur de description. Une hypothèse est simple si elle admet une description concise et une hypothèse est d'autant plus intéressante qu'elle conduit à une description plus concise de la réalité.

Pas de vérité sous-jacente. – Ce principe se démarque sur un point que Rissanen considère comme important, et nous avec lui : il n'est pas nécessaire de supposer qu'il existe une "vraie" hypothèse à retrouver absolument. L'objectif est plutôt de tirer le meilleur parti des données dont on dispose, et uniquement des données, sans sur-apprendre.

Gestion automatique du sur-apprentissage. – Une évaluation MDL prévient automatiquement le sur-apprentissage, sans paramètre à ajuster. En effet, le concept de longueur de description permet de comparer de manière homogène le degré d'adaptation d'une hypothèse aux données et la complexité de l'hypothèse. On notera que cette affirmation relève pour l'instant uniquement d'une vérité sur le papier : il existe des instanciations du principe conduisant à des critères paramétriques [Rissanen, 1989], [Yamanishi, 1998].

Interprétation bayésienne. – Le principe MDL est également censé admettre une interprétation bayésienne. Il semble plus juste de dire que l'inférence bayésienne admet une interprétation informationnelle. Nous revenons sur ce point dans la suite. En effet, l'inférence bayésienne constitue une version "praticable" du principe MDL.

Lien avec l'induction. – Enfin, la recherche d'une hypothèse autorisant une description la plus concise possible est liée, au moins sur le plan formel, à l'obtention d'une hypothèse performante en généralisation (*i.e.* valide sur de nouvelles données). C'est d'ailleurs dans le but d'obtenir une théorie effective de l'induction que les travaux de Solomonoff se sont orientés vers l'obtention d'une version "idéale" du principe MDL.

3.2.3 Instanciation idéale du principe

Un ensemble d'hypothèses $\mathcal{H}$ étant donné, il existe une réponse "idéale" aux questions 2 et 5 de la fig.3.3. Celle-ci est basée sur la notion de complexité algorithmique.

Complexité algorithmique de Kolmogorov [Solomonoff, 1964]. – Soient x et y deux objets. La *complexité algorithmique de y conditionnellement à x* est la longueur du plus court programme prenant en entrée x et calculant y. Cette complexité est définie à une constante près, dépendante du langage utilisé. On la note $K(y|x)$. Si x est l'objet vide ϵ, on note $K(y) = K(y|\epsilon)$ la *complexité algorithmique de y*.

Principe MDL idéal [Vitanyi and Li, 2000]. – Soit $\mathcal{H}$ un ensemble d'hypothèses et soit D un ensemble de données. La version idéale du principe MDL préconise de sélectionner l'hypothèse

$$H_{opt} = \arg\min_{H \in \mathcal{H}} K(H) + K(D|H). \tag{3.9}$$

Intérêt de l'approche idéale. – L'approche idéale possède l'avantage de fournir un cadre à l'étude des propriétés théoriques du principe MDL. Par exemple, elle permet d'étudier l'intérêt prédictif du principe MDL et d'établir un lien entre compression et prédiction. Elle prend ses racines dans le travail de Solomonoff [Solomonoff, 1964]. On trouve une description de ce lien dans [Vitanyi and Li, 2000] et, de manière plus détaillée, dans [Li and Vitanyi, 1997]. Essentiellement, le cadre idéal permet de montrer que rechercher une hypothèse compressive revient à rechercher une hypothèse minimisant l'erreur de prédiction.

Limite de l'approche idéale. – Il n'existe pas de programme retournant pour tout objet donné sa complexité algorithmique. Autrement dit, cette dernière n'est pas calculable. C'est ce qui constitue un obstacle critique à la mise en pratique de l'évaluation idéale.

Cas supervisé avec dépendance partielle aux données. – Le principe MDL est un principe générique dont la forme demande à être précisée en fonction de la situation, ce que nous faisons ici. Dans notre cas, les données D sont de la forme $D = (\mathcal{I}, \delta, Y)$, avec $\mathcal{I}$ l'ensemble des individus, δ une matrice de Gram sur $\mathcal{I}$ et Y une variable cible. L'ensemble d'hypothèses est $\mathcal{H} = \mathcal{H}_\delta(\mathcal{I})$ et dépend partiellement des données. Cette situation n'est pas un sujet classique d'étude dans la littérature. Nous proposons l'adaptation suivante du principe MDL, que nous donnons sous forme idéale :

$$H_{opt} = \underset{H \in \mathcal{H}_\delta(\mathcal{I})}{\arg\min}\ K(H|\mathcal{I}, \delta) + K(Y|H, \mathcal{I}, \delta). \tag{3.10}$$

Les hypothèses étant dépendantes des individus $\mathcal{I}$ et de la matrice de Gram δ, la description d'une hypothèse utilise nécessairement cette connaissance. Nous définissons la longueur de description d'une hypothèse H comme la complexité algorithmique $K(H|\mathcal{I}, \delta)$ de H conditionnellement à $\mathcal{I}$ et δ. Conditionnellement à $H, \mathcal{I}$ et δ, nous proposons ensuite de décrire la variable cible Y. Nous définissons la longueur de description de la cible Y comme la complexité algorithmique $K(Y|H, \mathcal{I}, \delta)$ de Y conditionnellement à $H, \mathcal{I}$ et δ.

3.2.4 L'inférence bayésienne : une instanciation praticable

L'approche bayésienne de l'inférence constitue une instanciation, praticable cette fois-ci, du principe MDL. Elle repose sur le principe suivant.

Principe du maximum a posteriori. – Soient D des données et soit $\mathcal{H}$ un ensemble d'hypothèses. Le principe du maximum a posteriori, abrégé en MAP, préconise de sélectionner l'hypothèse la plus probable connaissant les données :

$$H_{opt} = \underset{H \in \mathcal{H}}{\arg\max}\ p_D(H). \tag{3.11}$$

La question de la définition de la probabilité p_D se pose alors. Dans le cadre de l'inférence bayésienne, cette question est ramenée à la définition d'une distribution de probabilité p sur $\mathcal{H}$ et d'une fonction de vraisemblance q_H sur les données pour tout $H \in \mathcal{H}$.

Inférence bayésienne. – Soient D des données et soit $\mathcal{H}$ un ensemble d'hypothèses. Dans l'approche bayésienne, la quantité $p_D(H)$ est décomposée suivant la formule

$$p_D(H) = p(H)q_H(D), \tag{3.12}$$

avec p une distribution de probabilité sur l'ensemble $\mathcal{H}$ des hypothèses, dite *a priori*, et, pour chaque hypothèse H, q_H une fonction sur les données, dite de *vraisemblance*. En suivant le principe du maximum a posteriori, il est alors préconisé de sélectionner l'hypothèse H qui maximise le produit $p(H)q_H(D)$ d'un a priori et d'une vraisemblance.

Notons que dans le cas où l'expression $\sum_H p(H)q_H(D)$ est définie et finie, la fonction de vraisemblance peut être définie à une constante près : la décision a posteriori reste inchangée. C'est le cas lorsque l'ensemble des hypothèses est de cardinal fini. Il est alors possible de se ramener au cas où p_D est une distribution de probabilité et p_D est appelée distribution *a posteriori*. Pour toute hypothèse H, la quantité $p_D(H)$ est appelée *probabilité de H sachant D*. On la note plus usuellement $p(H/D)$.

Cas supervisé avec dépendance partielle aux données. – Dans notre cas, les données D sont de la forme $D = (\mathcal{I}, \delta, Y)$ et l'ensemble d'hypothèses $\mathcal{H} = \mathcal{H}_\delta(\mathcal{I})$ dépend partiellement des données. Cette situation est particulière et nécessite de retravailler la forme de l'inférence bayésienne. Nous proposons de réécrire la probabilité a posteriori $p(H/D)$ de la manière suivante, pour une hypothèse $H \in \mathcal{H}_\delta(\mathcal{I})$:

$$\begin{aligned} p(H/D) &= p(H/\mathcal{I}, \delta, Y) \\ &= \frac{p(H, \mathcal{I}, \delta, Y)}{p(\mathcal{I}, \delta, Y)} \\ &= \frac{p(\mathcal{I}, \delta)p(H/\mathcal{I}, \delta)p(Y/H, \mathcal{I}, \delta)}{p(\mathcal{I}, \delta, Y)}. \end{aligned} \tag{3.13}$$

En éliminant les termes ne dépendant pas de l'hypothèse H, le principe MAP s'écrit alors :

$$H_{opt} = \arg\max_{H \in \mathcal{H}_\delta(\mathcal{I})} \; p(H/\mathcal{I}, \delta)p(Y/H, \mathcal{I}, \delta). \tag{3.14}$$

L'inférence bayésienne passe dans notre cas par la proposition d'une distribution de probabilité $p_{\mathcal{I},\delta}$ sur l'ensemble des hypothèses et d'une distribution $p_{H,\mathcal{I},\delta}$ sur l'ensemble des étiquetages pour toute hypothèse $H \in \mathcal{H}_\delta(\mathcal{I})$. Afin de ne pas multiplier les notations, nous conservons les appellations définies dans le cas générique : $p_{\mathcal{I},\delta}$ est appelée *distribution a priori* et $p_{H,\mathcal{I},\delta}$ est appelée *fonction de vraisemblance*.

Le lien entre l'inférence bayésienne et le principe MDL est établi par la *correspondance de Shannon*.

Correspondance de Shannon [Shannon, 1948]. – Pour une distribution de probabilité L et un code C q-aire définis sur un univers Ω fini, on note $\bar{l}_L(C)$ la longueur moyenne des mots du code C relativement à L :

$$\bar{l}_L(C) = \sum_\omega L(\omega) l_C(\omega), \tag{3.15}$$

où $l_C(\omega)$ mesure la longueur du mot associé à un élément ω. On note $Ent_q(L)$ l'*entropie de Shannon* de L :

$$Ent_q(L) = -\sum_{\omega} L(\omega) \log_q L(\omega). \tag{3.16}$$

Un premier résultat de Shannon montre que la longueur moyenne de code relativement à L est minorée par l'entropie de L, pour tout code. Plus précisément

$$Ent_q(L) \leq \bar{l}_L(C). \tag{3.17}$$

Un second résultat de Shannon montre qu'il existe un code C_0 pour lequel la longueur moyenne des mots du code atteint quasiment le minimum entropique. Plus précisément

$$\bar{l}_L(C_0) < Ent_q(L) + 1. \tag{3.18}$$

Définitions. – Soit L une distribution de probabilités sur un ensemble fini Ω. Pour $\omega \in \Omega$, la quantité $-\log L(\omega)$ est appelée *longueur de description* de ω. L'unité de mesure en est le *nat* (pour natural digit en anglais), ou le *bit* (de l'anglais binary digit) lorsque le logarithme est en base 2.

Approximation de la complexité algorithmique. – Soit ω un objet. La complexité algorithmique $K(\omega)$ est une quantité universelle car dépendant uniquement de l'objet ω, modulo la restriction de l'ensemble des programmes calculant ω à l'ensemble des programmes préfixes. Si on voit ω comme élément d'un ensemble fini Ω, et si on dispose d'une distribution de probabilité L_Ω sur Ω, il existe un code C_0 sur Ω dont la longueur moyenne des mots est minimale et égale à l'entropie de Shannon de L_Ω. A ω est ainsi associé un mot $C_0(\omega)$ de longueur $-\log L_\Omega(\omega)$.

Intuitivement, l'approximation suivante est réalisée :

$$K(\omega) \approx -\log L_\Omega(\omega). \tag{3.19}$$

L'intuition repose sur la constatation suivante : on peut voir le mot de code $C_0(\omega)$ comme un programme décrivant ω. Au lieu de considérer le programme le plus court décrivant ω, on considère un ensemble C_0 de programmes de longueurs les plus courtes en moyenne relativement à une distribution L_Ω sur un ensemble Ω contenant ω.

Là où la complexité de Kolmogorov définit pour tout objet une longueur minimale universelle car dépendant uniquement de l'objet considéré, la correspondance de Shannon produit un ensemble de longueurs minimales en moyenne relativement à une distribution de probabilité prédéterminée. Si on perd l'universalité, c'est au profit d'un moyen effectif d'obtenir des longueurs de description : en fixant un ensemble dans lequel vivent les objets et en fixant une distribution sur cet ensemble.

Un intérêt supplémentaire de la correspondance de Shannon est de rendre inutile la construction effective d'un code : pour obtenir les longueurs de codage, il suffit de prendre l'opposé du logarithme des probabilités. Répétons-le : il n'est pas nécessaire de coder effectivement les objets.

Approximation bayésienne du principe MDL idéal. – Soit p une distribution a priori sur l'ensemble des hypothèses et, pour toute hypothèse H, soit q_H une fonction de

vraisemblance sur les données. En notant $l(H) := -\log p(H)$ et $l_H(D) := -\log q_H(D)$, l'inférence bayésienne réalise une approximation du principe MDL idéal :

$$K(H) \approx l(H) \text{ et } K(D|H) \approx l_H(D). \tag{3.20}$$

Approximation bayésienne dans notre contexte. – On se place dans notre contexte d'évaluation : supervisé avec hypothèses dépendant partiellement des données. Soit $p_{\mathcal{I},\delta}$ une distribution a priori sur l'ensemble des hypothèses $\mathcal{H}_\delta(\mathcal{I})$ et, pour toute hypothèse $H \in \mathcal{H}_\delta(\mathcal{I})$, soit $p_{H,\mathcal{I},\delta}$ une fonction de vraisemblance sur l'ensemble des variables cibles.

En notant $l_{\mathcal{I},\delta}(H) := -\log p_{\mathcal{I},\delta}(H)$ et $l_{H,\mathcal{I},\delta}(Y) := -\log q_{H,\mathcal{I},\delta}(Y)$, l'inférence bayésienne réalise une approximation du principe MDL idéal (*c.f.* 3.10) :

$$K(H|\mathcal{I},\delta) \approx l_{\mathcal{I},\delta}(H) \text{ et } K(Y|H,\mathcal{I},\delta) \approx l_{H,\mathcal{I},\delta}(Y). \tag{3.21}$$

3.3 Nouveau critère d'évaluation

A partir d'un ensemble d'individus $\mathcal{I}$ et d'une matrice de Gram δ nous avons défini un ensemble de partitions de Voronoi $\mathcal{H}_\delta(\mathcal{I})$. Nous allons maintenant instancier le principe MDL et proposer un critère d'évaluation de type MDL de ces partitions. Pour cela, nous adoptons l'approche bayésienne que nous avons adaptée à notre contexte.

Nous proposons dans ce qui suit une distribution a priori $p_{\mathcal{I},\delta}$ sur l'ensemble des hypothèses $\mathcal{H}_\delta(\mathcal{I})$ et, pour toute hypothèse $H \in \mathcal{H}_\delta(\mathcal{I})$, une fonction de vraisemblance $p_{H,\mathcal{I},\delta}$ sur l'ensemble des variables cibles.

3.3.1 Proposition d'une distribution a priori

Nous proposons ici une distribution a priori $p_{\mathcal{I},\delta}$ sur l'ensemble des hypothèses $\mathcal{H}_\delta(\mathcal{I})$. Soit $H \in \mathcal{H}_\delta(\mathcal{I})$ une hypothèse. Rappelons que K désigne le nombre de cellules de la partition de voronoi associée.

Décomposition de l'a priori. – La formule des probabilités itérées permet d'écrire :

$$p_{\mathcal{I},\delta}(H) = p_{\mathcal{I},\delta}(K,H) = q_{\mathcal{I},\delta}(K)p_{K,\mathcal{I},\delta}(H). \tag{3.22}$$

Pour obtenir une distribution a priori sur les hypothèses, il suffit donc de proposer une distribution $q_{\mathcal{I},\delta}$ sur le nombre de prototypes puis une distribution $p_{K,\mathcal{I},\delta}$ sur les partitions composées de K cellules.

Longueurs de description du nombre de prototypes. – Nous considérons le nombre K de prototypes comme un élément de l'ensemble $[\![1, N]\!]$. Nous choisissons la distribution uniforme sur cet ensemble :

$$q_{\mathcal{I},\delta}(K) = \frac{1}{N}, \tag{3.23}$$

et il existe d'après la correspondance de Shannon un codage de K pour lequel la longueur du mot de code associé est $\log N$. Nous évaluons donc la longueur de description de K à $\log N$ nats.

Longueurs de description des prototypes. – Une partition de Voronoi H en K cellules est définie par un ensemble de K prototypes. Nous considérons cet ensemble de prototypes comme un élément de l'ensemble des K-combinaisons avec répétition dans un ensemble à N éléments. C'est un ensemble de cardinal $\binom{N+K-1}{K}$. Nous adoptons un a priori uniforme sur cet ensemble :

$$p_{K,\mathcal{I},\delta}(H) = \frac{1}{\binom{N+K-1}{K}}. \tag{3.24}$$

D'après la correspondance de Shannon, nous évaluons la longueur de description de l'ensemble $\{k;\ k \in H\}$ à $\log \binom{N+K-1}{K}$ nats.

Ainsi, la longueur de description d'une partition H en K cellules est :

$$l_{\mathcal{I},\delta}(H) = \log N + \log \binom{N+K-1}{K} \text{ nats.} \tag{3.25}$$

3.3.2 Proposition d'une fonction de vraisemblance

Soit $H \in \mathcal{H}_\delta(\mathcal{I})$ une hypothèse et soit K son nombre de cellules. Nous proposons ici une fonction de vraisemblance $p_{H,\mathcal{I},\delta}$ sur l'ensemble des variables cibles, *i.e.* sur l'ensemble des applications de $\mathcal{I}$ dans $\mathcal{L}$.

Décomposition de la vraisemblance. – Soit Y la variable cible et Y_k la restriction de Y à la k^{eme} cellule de Voronoi $V_\delta(k,H)$ $(1 \leq k \leq K)$. Ainsi, Y_k est l'application de $V_\delta(k,H)$ dans l'ensemble des étiquettes qui à tout individu n de la cellule associe son étiquette $Y(n)$. Nous supposons les variables Y_k conditionnellement indépendantes :

$$p_{H,\mathcal{I},\delta}(Y) = \prod_{k=1}^{K} p_{V_\delta(k,H),\delta}(Y_k). \tag{3.26}$$

Autrement dit, l'attribution des étiquettes est considérée indépendante d'un groupe à l'autre. C'est une hypothèse plus faible que l'hypothèse i.i.d usuellement adoptée en statistique pour l'estimation de densité.

Soit $k \in [\![1, K]\!]$. Rappelons que N_{kj} désigne le nombre d'individus dans la cellule k portant l'étiquette j. La formule des probabilités itérées permet d'écrire :

$$\begin{aligned} p_{V_\delta(k,H),\delta}(Y_k) &= p_{V_\delta(k,H),\delta}(N_{k1},\ldots,N_{kJ},Y_k) \\ &= q_{V_\delta(k,H),\delta}(N_{k1},\ldots,N_{kJ}) p_{N_{k1},\ldots,N_{kJ},V_\delta(k,H),\delta}(Y_k) \end{aligned} \tag{3.27}$$

Pour obtenir une fonction de vraisemblance, il suffit alors de proposer une distribution $q_{V_\delta(k,H),\delta}$ sur les fréquences puis une distribution $p_{N_{k1},\ldots,N_{kJ},V_\delta(k,H),\delta}$ sur les étiquetages, pour chaque cellule k.

Longueur de description des fréquences. – Soit $k \in H$. Nous considérons le vecteur des fréquences $(N_{k1}, \ldots, N_{kJ})$ des J classes cibles dans la cellule k comme un élément de l'ensemble des vecteurs de dimension J à coefficients entiers dont la somme vaut N_k. Cet ensemble a pour cardinal $\binom{N_k+J-1}{J-1}$. Nous choisissons la distribution uniforme sur cet ensemble :

$$q_{V_\delta(k,H),\delta}(N_{k1}, \ldots, N_{kJ}) = \frac{1}{\binom{N_k+J-1}{J-1}}. \tag{3.28}$$

La correspondance de Shannon nous permet d'évaluer la longueur de description de $(N_{k1}, \ldots, N_{kJ})$ à $\log \binom{N_k+J-1}{J-1}$ nats.

Longueur de description des étiquettes. – Soit $k \in H$. Nous considérons la variable Y_k comme un élément de l'ensemble des applications de $V_\delta(k, H)$ dans $\mathcal{L}$ dont N_{kj} individus ont pour image j, pour tout j. Dénombrer cet ensemble revient à dénombrer les répartitions de N_k individus dans J boîtes sous contrainte d'en placer exactement N_{kj} dans la j^{eme} $(1 \leq j \leq J)$. C'est un problème multinomial, le nombre de possibilités étant quantifié par le coefficient multinomial $\frac{N_k!}{N_{k1}!\ldots N_{kJ}!}$. Nous obtenons, en choisissant la distribution uniforme :

$$p_{N_{k1},\ldots,N_{kJ},V_\delta(k,H),\delta}(Y_k) = \frac{N_{k1}! \ldots N_{kJ}!}{N_k!}. \tag{3.29}$$

La correspondance de Shannon nous permet d'évaluer la longueur de description de Y_k à $\log \frac{N_k!}{N_{k1}!\ldots N_{kJ}!}$ nats.

La vraisemblance étant spécifiée indépendamment dans chaque cellule, nous obtenons finalement pour longueur de description des données à partir de H

$$l_{H,\mathcal{I},\delta}(Y) = \sum_{k=1}^{K} \log \binom{N_k + J - 1}{J - 1} + \sum_{k=1}^{K} \log \frac{N_k!}{N_{k1}! \ldots N_{kJ}!}. \tag{3.30}$$

3.3.3 Le critère d'évaluation

Nous avons adopté une approche descriptive compressive de l'évaluation, que nous résumons.

Notre approche de l'évaluation. – La démarche que nous proposons pour obtenir un critère d'évaluation se décompose en une phase de spécification des hypothèses et une phase de spécification d'une distribution a priori et d'une fonction de vraisemblance. Les hypothèses spécifiées sont en nombre fini et dépendent des données. La correspondance de Shannon transforme la question de la spécification des longueurs de description en une question de spécification de distributions de probabilité.

Pour spécifier ces distributions, nous proposons de définir un protocole de description hiérarchique, de spécifier l'ensemble dans lequel est placé l'objet décrit à un niveau de la hiérarchie et d'adopter la distribution uniforme sur cette ensemble. Le protocole de description proposé dans le cas des partitions de Voronoi est donné à la fig.3.4.

1. description du nombre K de cellules de la partition,
2. description des K individus définissant la partition,
3. pour chaque groupe, description des J fréquences des étiquettes,
4. pour chaque groupe, description des étiquettes des individus.

FIG. 3.4 – Protocole de description d'une partition et des étiquette à partir de cette partition.

Cette approche de l'évaluation nous a conduit à définir le critère d'évaluation $c_{\delta,Y}^{MDL}(H)$ suivant, pour toute partition de Voronoi H dans $\mathcal{H}_\delta(\mathcal{I})$ (le critère permet notamment de comparer des partitions de tailles différentes) :

$$c_{\delta,Y}^{MDL}(H) = \log N + \log \binom{N+K-1}{K} + \sum_{k=1}^{K} \log \binom{N_k+J-1}{J-1} + \sum_{k=1}^{K} \log \frac{N_k!}{N_{k1}! \dots N_{kJ}!}. \tag{3.31}$$

Le premier terme correspond à la description du nombre de cellules de la partition H, le second à la description des prototypes, le troisième à la description des fréquences des étiquettes dans les cellules et le dernier à la description de l'étiquetage des individus dans les cellules.

Propriété. – Notons que, d'après l'approximation de Stirling ($\log x! \approx x \log x - x + O(\log x)$), le quatrième et dernier terme de la formule se comporte asymptotiquement comme N fois l'entropie conditionnelle de la distribution des étiquettes en connaissance de la fonction d'assignement associée à la partition :

$$\sum_{k=1}^{K} \log \frac{N_k!}{N_{k1}! \dots N_{kJ}!} \approx -N \sum_{k=1}^{K} \sum_{j=1}^{J} \frac{N_{kj}}{N} \log \frac{N_{kj}}{N_k}. \tag{3.32}$$

Pour une cellule contenant des individus portant tous la même étiquette, le coefficient multinomial vaut 1 et la contribution de cette cellule à la quatrième partie du critère est nulle. A l'inverse, plus la répartition des étiquettes se rapproche de l'uniforme, plus la contribution est proche d'un maximum. Le dernier terme du critère mesure donc la finesse du point de vue sur la variable cible que donne une partition, dans un sens entropique. Les trois premiers termes pondèrent ce point de vue en pénalisant les partitions plus complexes (*c.f.* fig.3.5).

Définition. – Nous qualifions de *structurelle* la longueur de description correspondant à la somme des trois premiers termes

$$\log N + \log \binom{N+K-1}{K} + \sum_{k=1}^{K} \log \binom{N_k+J-1}{J-1}. \tag{3.33}$$

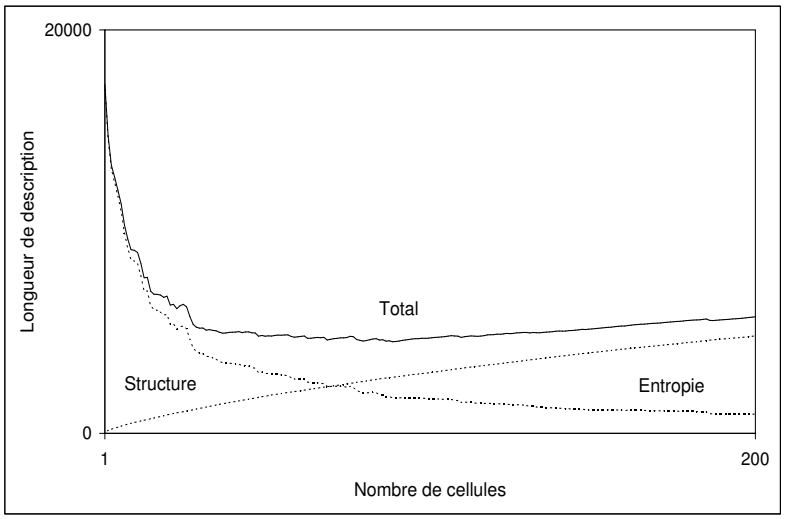

FIG. 3.5 – Le critère réalise un compromis entre complexité de la partition, mesurée par la longueur de description structurelle, et degré de pureté des cellules, mesuré par la longueur de description entropique.

et nous parlons de longueur de description *entropique* au sujet du dernier terme

$$\sum_{k=1}^{K} \log \frac{N_k!}{N_{k1}! \dots N_{kJ}!}. \tag{3.34}$$

Définition. – Notons $c^*_{\delta,Y}$ la valeur minimale du critère $c^{MDL}_{\delta,Y}$ sur l'ensemble des hypothèses considérées. Afin de travailler avec un indicateur normalisé et permettre la comparaison, nous considérons la transformation suivante de $c^*_{\delta,Y}$:

$$g^*_{\delta,Y} = 1 - \frac{c^*_{\delta,Y}}{c^0_{\delta,Y}}, \tag{3.35}$$

où $c^0_{\delta,Y}$ est la valeur du critère $c^{MDL}_{\delta,Y}$ pour l'hypothèse nulle (*i.e.* la partition constituée par un seul groupe). Comme $c^{MDL}_{\delta,Y}$ est une longueur de codage, $g^*_{\delta,Y}$ mesure un gain de compression. Le gain de compression mesure le rapport entre la partition établissant le meilleur compromis entre finesse et fiabilité et la partition la moins fine et la plus fiable.

Le gain de compression $g^*_{\delta,Y}$ est supérieur à 0. En effet, $c^*_{\delta,Y}$ est la valeur minimale atteinte par le critère $c_{\delta,Y}$, nécessairement inférieure à $c^0_{\delta,Y}$. Le gain de compresssion est de plus inférieur à 1. Si $g^*_{\delta,Y} = 0$, la matrice de Gram δ n'apporte aucune information sur la cible. Plus la valeur de $g^*_{\delta,Y}$ est proche de 1, plus la partition est informative et fiable.

3.4 Nouvel algorithme d'optimisation

Nous avons défini un critère supervisé $c^{MDL}_{\delta,Y}(H)$ d'évaluation de toute partition de Voronoi $H \in \mathcal{H}_\delta(\mathcal{I})$. Comme préconisé par le principe MDL, la meilleure partition à considérer est celle minimisant ce critère. L'espace des solutions a pour cardinal 2^N au plus et il est peu réaliste d'envisager une recherche exhaustive. C'est pourquoi nous proposons une nouvelle heuristique, encapsulant une optimisation gloutonne descendante d'un ensemble de prototypes dans une méta-heuristique de recherche à voisinage variable.

3.4.1 Heuristique gloutonne

L'heuristique gloutonne GLOUTON(H) que nous proposons (*c.f.* fig.3.6) s'applique à tout ensemble H de K prototypes préliminairement fixé. Chaque ensemble obtenu par suppression d'un élément de H est évalué. Parmi ces ensembles, celui minimisant le critère est déclaré vainqueur de l'étape. Ce procédé est itéré par application aux vainqueurs successifs jusqu'à l'évaluation finale d'un singleton. Le meilleur ensemble rencontré lors du parcours est renvoyé.

- $S \leftarrow H$
- **Pour** $t = K - 1$ à 1 **Faire**
  - $s_0 \leftarrow \arg\min_{s \in S} c_{\delta,Y}(S \setminus \{s\})$
  - $S \leftarrow S \setminus \{s_0\}$
- **Retourner** le meilleur ensemble de prototypes rencontré

FIG. 3.6 – Heuristique gloutonne.

Complexité initiale. – Cette heuristique est de complexité temporelle un $O(NK^3)$. Au cours de chacune des K étapes, elle évalue au plus K partitions, et le calcul de chaque partition nécessite N recherches du plus proche prototype parmi au plus K éléments.

Nous proposons des optimisations afin de réduire la complexité algorithmique. Elles reposent sur la constatation suivante.

Constat. – A chaque étape, toute suppression d'un prototype conduit à réattribuer à leur plus proche prototype respectif suivant uniquement les individus appartenant à la cellule du prototype supprimé. Les N instances ne sont donc à traiter qu'une fois au cours de l'étape, puisqu'on travaille avec une partition des instances.

Chacune des K étapes de l'heuristique nécessite exactement N traitements (N réaffectations). Nous montrons que chaque traitement peut être effectué avec une complexité temporelle constante.

Proposition 3 *La complexité algorithmique de l'heuristique gloutonne peut être réduite à un $O(NK \log K)$ en temps, au prix d'une complexité un $O(NK)$ en mémoire.*

Preuve. – D'après le constat réalisé ci-dessus, il suffit de montrer que la réaffectation d'un individu à son plus proche prototype suivant s'effectue à coût constant et que la modification de la valeur du critère nécessite un nombre constant d'opérations.

Si l'on dispose pour chaque individu de la liste triée des éléments de H par distance croissante, la réaffectation se fait à coût constant. L'algorithme GLOUTON(H) se voit alors adjoindre une phase d'initialisation, calculant ces N listes triées. Cette phase possède une complexité temporelle un $O(NK \log K)$ (construction de N listes triées de taille K) et une complexité spatiale un $O(NK)$ (stockage de N listes de longueur K).

Les listes sont mises à jour uniquement lorsque nécessaire. Il suffit pour cela de conserver un dictionnaire des prototypes disponibles, *i.e.* les prototypes qui n'ont pas été éliminés dans une étape précédente. Pour une instance, lorsqu'on cherche le plus proche prototype suivant, la liste est parcourue à partir de la tête jusqu'à obtenir un prototype disponible et tout prototype non disponibles rencontré est supprimé de la liste. Comme nous n'envisageons que des suppressions, chaque liste est parcourue une unique fois durant toute l'optimisation. La mise à jour des listes participe pour un $O(NK)$ à la complexité temporelle de l'algorithme.

Pour obtenir un coût constant de traitement d'un individu, il reste à montrer que la valeur du critère est mise à jour à coût constant. Notons que seuls les deux derniers termes du critère dépendent de la répartition des individus dans les cellules. Tout d'abord, il faut soustraire la contribution à la valeur du critère de la cellule du prototype éliminé. Ensuite, la réattribution d'un individu à son plus proche prototype k suivant induit une simple incrémentation unitaire des compteurs N_k et N_{kj_0}, où j_0 est l'indice de la classe à laquelle appartient l'individu. Le terme du critère porté par le prototype k est alors mis à jour en ajoutant $\log(N_k + J) - \log(N_{kj_0} + 1)$. ■

Complexité finale. – Au final, l'algorithme comporte une phase d'initialisation, de complexités un $O(NK \log K)$ en temps et un $O(NK)$ en espace, et une phase d'optimisation de complexité temporelle un $O(NK)$ et de complexité spatiale un $O(NK)$. Remarquons qu'en pratique, du fait que les opérations effectuées au cours de la première étape sont plus élémentaires que celles effectuées dans la seconde, le temps de calcul se répartit de manière équivalente entre les deux étapes. La version optimisée de l'algorithme est décrite dans la fig.3.7.

3.4.2 Méta-heuristique de recherche à voisinage variable

L'heuristique gloutonne possède une complexité temporelle un $O(NK \log K)$ et une complexité spatiale un $O(NK)$, si K est le cardinal de l'ensemble de prototypes optimisé. Afin de conserver une complexité raisonnable, il est naturel d'envisager une application répétée de cette algorithme à des ensembles de prototypes de taille contrôlée.

La première idée venant à l'esprit consiste à effectuer une multi-initialisation aléatoire. A chaque initialisation, un ensemble de prototypes de taille donnée est tiré uniformément et se voit appliquer l'heuristique gloutonne. Comme le laisse supposer l'intuition, nous avons constaté expérimentalement qu'une recherche "aveugle" n'est susceptible d'améliorer la solution que très lentement. Il existe des méta-heuristiques permettant d'orienter

- **Pour** $n = 1$ à N **Faire**
  - $L_n \leftarrow$ la liste des prototypes triés par distance croissante à l'individu n
- $S \leftarrow H$
- $BestLabelCost \leftarrow$ la valeur des deux derniers termes du critère pour S
- **Pour** $t = K - 1$ à 1 **Faire**
  - $LocallyBestLabelCost \leftarrow +\infty$
  - **Pour** $s \in S$ **Faire**
    - $LabelCost \leftarrow BestLabelCost$
    - Mettre à jour $LabelCost$ en soustrayant les termes relatifs à la cellule de s
    - **Pour** $n \in V(s)$ **Faire**
      - $s' \leftarrow$ le plus proche prototype de n après s disponible dans L_n
      - Mettre à jour $LabelCost$ en ajoutant les quantités relatives au passage de n de s vers s'
    - **Si** $LabelCost < LocallyBestLabelCost$
      - $LocallyBestLabelCost \leftarrow LabelCost$
      - $s_0 \leftarrow s$
  - $S \leftarrow S \setminus \{s_0\}$
  - $BestLabelCost \leftarrow LabelCost$
  - Marqué s comme indisponible
- **Retourner** le meilleur ensemble de prototypes rencontré

FIG. 3.7 – L'algorithme glouton optimisé.

plus finement la remise en question aléatoire. Nous appliquons la méta-heuristique de recherche à voisinage variable décrite dans [Hansen and Mladenovic, 2001].

Celle-ci consiste, pour une solution H, à sélectionner une solution H' dans un "voisinage" $\mathcal{V}_t(H)$ de H et à lui appliquer l'heuristique de base, ici l'heuristique gloutonne. Cette recherche est itérée comme décrit à la fig.3.8. Tout repose sur la définition de la famille de voisinages $(\mathcal{V}_t(H))$ d'une partition H. Nous proposons la suivante.

Définition d'une notion de voisinage. – Pour un ensemble H_0 de K prototypes, un *voisin* est toute union disjointe de deux ensembles de prototypes $H = H_1 \bigsqcup H_2$ tels que

- H_1 est inclus dans H_0,
- H_2 est un ensemble d'individus inclus dans l'union des cellules de $V(H_0)$ dont le prototype n'est pas dans H_1.

Si $t \in [0, 1]$, le voisinage $\mathcal{V}_t(H_0)$ contient tous les voisins $H = H_1 \bigsqcup H_2$ de H_0 tels que

- H_1 est constitué d'éléments de H_0 dans une proportion de $1 - t$,
- H_2 est constitué d'éléments de l'union des cellules dont le prototype est dans $H_0 \setminus H_1$, dans une proportion t.

Un exemple de partitions voisines est donné à la fig.3.9. Nous parlons au sujet de $\mathcal{V}_t(H_0)$ de *voisinage de rayon t de l'hypothèse* H_0.

- $S_0 \leftarrow$ une solution initiale
- $R \leftarrow 0$
- **Tant que** $R \leq R_{max}$ **Faire**
  - $S' \leftarrow$ Sélection d'une solution dans $\mathcal{V}_{t_R}(S_0)$
  - $S \leftarrow$ GLOUTON(S')
  - **Si** S est meilleur que S_0
    - $S_0 \leftarrow S$
    - $R \leftarrow 0$
  - **Sinon**
    - $R \leftarrow R + 1$
- **Retourner** le meilleur ensemble de prototypes rencontré

FIG. 3.8 – La méta-heuristique de recherche à voisinage variable.

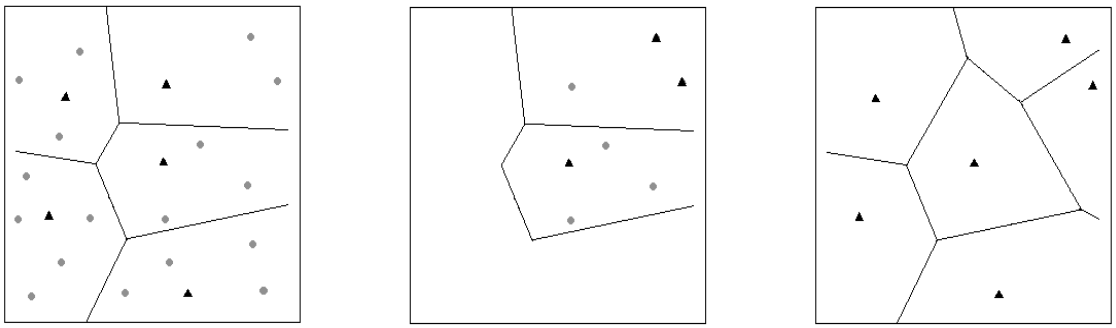

FIG. 3.9 – Exemple de partitions voisines pour $t = 0.35$. (a) Partition de Voronoi induite par cinq prototypes et répartition des individus dans les cellules de la partition. (b) 2 prototypes (soit 35% des prototypes) sont remis en cause et remplacés par 3 instances appartenant à l'union des deux cellules (soit 35% des individus appartenant aux cellules associées) les remplacent. (c) Partition voisine obtenue, basée sur six prototypes.

L'application de la méta-heuristique nécessite de définir la suite des tailles de voisinage à explorer. De plus, le nombre maximal de prototypes doit être contrôlé. Nous souhaitons contrôler ces définitions à l'aide d'un unique paramètre *Niveau*. Le but est de permettre à l'utilisateur de quantifier la durée d'optimisation souhaitée à travers la relation : une incrémentation unitaire du paramètre *Niveau* doit doubler le temps consacré à l'optimisation. Intuitivement, la qualité de la solution dépend en effet exponentiellement du temps d'optimisation. Nous devons tenir compte de cette relation si l'on veut qu'une incrémentation du *Niveau* conduise à un résultat significativement différent.

Pour un *Niveau* donné, nous définissons deux quantités : $R(Niveau)$, le nombre de rayons de voisinages à considérer successivement, et $K_{max}(Niveau)$, la taille maximale

autorisée d'un ensemble de prototypes. Afin de permettre la réimplantation de notre algorithme, nous précisons les relations obtenues. La fig.3.10 permet de visualiser les choix opérés.

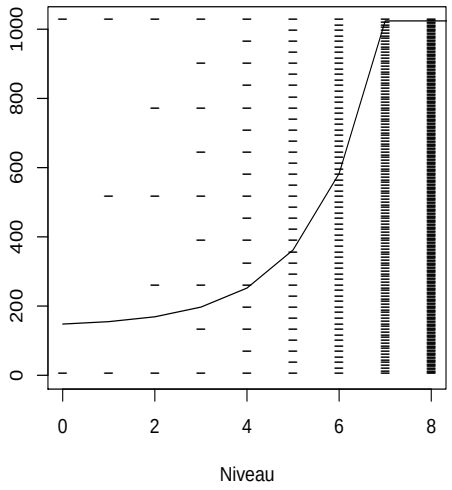

FIG. 3.10 – Visualisation du paramétrage de contrôle de la méta-heuristique. La taille maximale des ensembles de prototypes croît exponentiellement avec le $Niveau$, jusqu'à atteindre le nombre d'instances pour le $Niveau_{max}$ (ici $Niveau_{max} = 7$). Les rayons des voisinages considérées sont deux fois plus nombreux à chaque $Niveau$. Ici, $N = 1024$.

Contrôle de la méta-heuristique. – Pour une valeur de $Niveau$, nous fixons la famille $(t_R)_{0 \leq R \leq R(Niveau)}$ de rayons de voisinages à explorer pour une hypothèse H. La relation est la suivante : pour une incrémentation unitaire du $Niveau$, nous voulons être deux fois plus fin. Ceci nous amène à poser $R(Niveau) = 2^{Niveau}$ et nous définissons alors naturellement la famille $t_1, \ldots, t_{2^{Niveau}}$ des $R(Niveau)$ rayons :

$$t_R = \frac{R}{2^{Niveau}}, 0 \leq R \leq 2^{Niveau}. \tag{3.36}$$

Fixons maintenant la taille maximale autorisée $K_{max}(Niveau)$ de l'ensemble des prototypes. Pour le niveau 0, nous fixons $K_{max}(0) = \frac{N}{\log N}$. La suite $(K_{max}(Niveau))$ est nécessairement bornée par N. Nous n'avons pas d'autre choix que de fixer un niveau maximum $Niveau_{max}$ pour lequel $K_{max}(Niveau_{max}) = N$, par exemple : $Niveau_{max} = \lceil \log N \rceil$. Selon la même idée que précédemment, nous souhaitons alors mettre en place une progression géométrique entre $K_{max}(0)$ et N. Ceci conduit à définir :

$$K_{max}(Niveau) = \frac{\sum_{i=1}^{Niveau} 2^i}{\sum_{i=1}^{Niveau_{max}} 2^i} \times \left(N - \frac{N}{\log N} \right) + \frac{N}{\log N} \tag{3.37}$$

si $0 \leq Niveau \leq Niveau_{max}$, et $K_{max}(Niveau) = N$ sinon.

La contrainte sur le nombre maximal $K_{max}(Niveau)$ de prototypes intervient lorsqu'il s'agit de sélectionner les individus à ajouter : si le nombre d'individus à ajouter conduit à un ensemble de taille trop élevée, nous en sélectionnons juste assez pour atteindre la taille maximale licite égale à $K_{max}(Niveau)$.

Complexité. – Au final, l'algorithme RVVGlouton(0), réalisant une unique passe gloutonne sur un ensemble de $\frac{N}{\log N}$ prototypes sélectionnés uniformément, est de complexité un $O\left(N^2\left(1 - \frac{\log\log N}{\log N}\right)\right)$. Pour $Niveau \geq Niveau_{max}$, la complexité de RVVGlouton($Niveau$) est un $O(2^{Niveau} N^2 \log N)$.

En anticipant sur la suite, notons que le bon comportement de l'heuristique gloutonne et de la méta-heuristique de recherche à voisinage variable conduit à ne considérer que de faibles valeurs de $Niveau$. L'algorithme final est décrit à la fig.3.11.

- $K_{max}(Niveau) \leftarrow$ Calcul du nombre maximal de prototypes autorisé
- $DegreMax \leftarrow 2^{Niveau}$
- $S \leftarrow$ Sélection uniforme d'une solution de taille $K_{max}(0)$
- $S_0 \leftarrow$ GLOUTON(S)
- $Degre \leftarrow 1$
- **Tant que** $Degre < DegreMax$ **Faire**
  - $t \leftarrow Degre/DegreMax$
  - $S' \leftarrow$ Sélection d'une solution sous contrainte de taille dans $\mathcal{V}_t(S_0)$
  - $S \leftarrow$ GLOUTON(S')
  - **Si** S est meilleur que S_0
    - $S_0 \leftarrow S$
    - $Degre \leftarrow 1$
  - **Sinon**
    - $Degre \leftarrow Degre + 1$
- **Retourner** S_0

FIG. 3.11 – L'algorithme final RVVGlouton($Niveau$).

3.5 Conclusion

Nous avons élaboré dans ce chapitre une méthode qui prend en entrée une matrice de Gram, une variable cible catégorielle et renvoie un gain de compression. La méthode obtenue, que nous appelons Eva, évalue l'intérêt d'une matrice de Gram relativement à un problème de classification supervisée (*c.f.* fig.3.12).

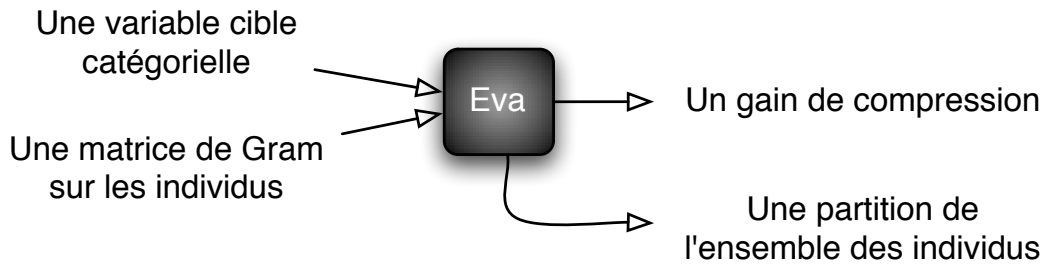

FIG. 3.12 – Eva, notre méthode d'évaluation.

Nous avons formulé la question de l'évaluation en un problème de recherche de la meilleure partition de l'ensemble des individus. Les partitions considérées sont les partitions de Voronoi. Nous avons proposé un nouveau critère d'évaluation de ces objets, en adaptant l'instanciation bayésienne du principe MDL. Nous avons développé une nouvelle heuristique de recherche de la meilleure partition.

Les partitions de Voronoi utilisées sont dépendantes des données, en nombre fini et ne sont ni des densités, ni des classifieurs. Elles n'entrent pas dans le cadre classique de la statistique. Nous avons dû adapter ce cadre afin d'obtenir un critère d'évaluation. Le critère obtenu est non-paramétrique et prend en charge la gestion du sur-apprentissage. D'autres démarches que la nôtre sont envisageables afin d'obtenir des critères prévenant automatiquement le sur-apprentissage. Elles nécessitent l'adaptation des hypothèses au cadre classique de la statistique et nous les présentons, les mettons en œuvre et les comparons au chapitre 4.

Nous procédons à une validation expérimentale de notre méthode au chapitre 5. Nous menons des expériences sur données réelles afin d'illustrer les qualités d'Eva. Nous montrons à l'aide d'expériences sur données synthétiques les apports séparés du critère et de l'algorithme.

Employée en tant que méthode d'évaluation, Eva calcule un gain de compression. Celui-ci quantifie la capacité de la matrice de Gram à produire une description concise de l'étiquetage des individus. Nous atteignons ainsi le but fixé : obtenir une évaluation supervisée automatique d'une matrice de Gram. Nous l'appliquons à la préparation de données séquentielles au chapitre 6.

4

Proposition d'autres critères d'évaluation et comparaison

Sommaire

Dans ce chapitre, nous considérons deux autres approches de l'évaluation. La première, relative à la théorie de l'apprentissage statistique développée par Vapnik, se base sur l'évaluation du risque empirique d'un classifieur. La seconde, relative à l'inférence bayésienne, considère l'évaluation de la vraisemblance d'une densité de probabilité. Afin de prévenir le risque de sur-apprentissage, le risque empirique doit être régularisé, la vraisemblance doit être pénalisée.

Dans la section 4.1, nous introduisons le principe de minimisation du risque empirique, ses limites, et sa version régularisée : le principe de minimisation du risque structurel. A partir de résultats disponibles dans la littérature, nous déduisons un premier critère d'évaluation des classifieurs induits par les partitions de Voronoi. Dans la section 4.2, nous décrivons l'inférence bayésienne de densités de probabilité. Nous montrons que les densités de probabilité induites par les partitions de Voronoi ne peuvent être traitées autrement que de manière heuristique. Nous procédons alors par analogie avec l'approche adoptée par Schwartz [Schwartz, 1978] pour définir un second critère d'évaluation.

Nous discutons les intérêts et les limites de ces deux critères, tout en montrant que le critère MDL proposé au chapitre 3 est le plus fin.

4.1 Le risque empirique et le risque structurel

En classification supervisée, une hypothèse est souvent évaluée par sa qualité prédictive. Nous présentons ici un indicateur de performance prédictive couramment employé : le risque empirique (ou : taux d'erreur en classification). Nous décrivons les procédés classiques d'estimation. Nous présentons la notion de régularisation structurelle introduite par Vapnik, qui permet de régulariser cet indicateur. Dans ce cadre, et en nous basant sur un travail existant, nous proposons un nouveau critère d'évaluation des partitions de Voronoi.

4.1.1 Estimation du risque empirique

Nous nous plaçons ici dans le cadre de la théorie de l'apprentissage statistique supervisé développée par Vapnik [Vapnik, 1996].

Notations. – Soit $D = (x_n, y_n)$ un échantillon de N réalisations indépendantes d'un couple de variables aléatoires (X, Y), X étant à valeurs dans un ensemble $\mathbb{X}$ et Y étant une variable binaire, *i.e.* à valeurs dans $\{0, 1\}$.

Définitions. – On appelle *classifieur* toute application f de $\mathbb{X}$ dans $\{0, 1\}$. On définit le *risque empirique* de f, et on le note $R_{emp}(f, D)$, par

$$R_{emp}(f, D) = 1 - \frac{1}{N} \sum_{n=1}^{N} \mathbb{1}_{\{y_n\}}(f(x_n)). \tag{4.1}$$

Autrement dit, le risque empirique est égal au nombre de mauvaises classifications du classifieur f sur l'échantillon D, rapporté à la taille de l'échantillon.

Principe de minimisation du risque empirique. – Soit $\mathcal{H}(D)$ un ensemble de classifieurs dépendant de l'échantillon. Le principe de minimisation du risque empirique conduit à sélectionner le classifieur f_{opt} tel que

$$f_{opt} = \underset{f \in \mathcal{H}(D)}{\arg\min}\, R_{emp}(f, D). \tag{4.2}$$

Le risque empirique possède deux inconvénients : c'est un estimateur à la fois biaisé et manquant de finesse.

Un estimateur biaisé. – Pour $f \in \mathcal{H}(D)$, le risque empirique $R_{emp}(f, D)$ est un estimateur de la probabilité d'erreur $p(f(X) \neq Y)$. Cet estimateur est biaisé et sous-estime toujours la probabilité d'erreur [Hastie *et al.*, 2001].

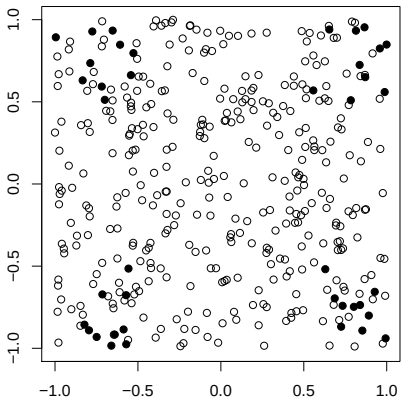

FIG. 4.1 – Illustration d'une limite de la validation. Sur ce jeu de données, les classes sont déséquilibrées : 10% des 400 individus sont dans une classe, les 90% restant dans une autre. Chaque individu de la classe minoritaire placé dans l'ensemble de validation réduit encore la représentation de cette classe en apprentissage.

Procédé de validation. – Si l'évaluation est biaisée, c'est parce que les données servant à ajuster le classifieur et les données sur lesquelles est mesuré le risque empirique sont les mêmes. Pour éliminer le biais, il suffit de séparer l'échantillon (x_n, y_n) en un échantillon d'*apprentissage* $D^{(a)} = (x_n, y_n)_{1 \leq n \leq M}$ et un échantillon de *validation* $D^{(v)} = (x_n, y_n)_{M+1 \leq n \leq N}$. Le classifieur est estimé à l'aide de l'échantillon d'apprentissage et le risque empirique est estimé sur l'échantillon de validation : $R_{emp}(f, D)$ est remplacé par $R_{emp}(f, D^{(v)})$.

Intérêt et limite de la validation. – Le procédé de validation établit un compromis entre qualité et fiabilité du classifieur. En augmentant la taille de l'échantillon d'apprentissage, la probabilité d'erreur des classifieurs considérés décroît. En augmentant la taille de l'échantillon de validation, la variance de l'estimation du risque empirique augmente.

Mais les données de validation sont perdues pour l'estimation du classifieur. Ceci limite a priori la qualité du classifieur estimé. Dans le cas où les classes cibles sont déséquilibrées, la classe minoritaire peut même se retrouver sous-représentée au point de perdre toute signification statistique, comme sur l'exemple illustratif donné à la fig.4.1. Ce cas n'est pas rare en pratique. Par exemple, lorsqu'on s'attaque à des problèmes de détection de fraude, même si l'on dispose de beaucoup d'individus (de l'ordre de 100000), les cas de fraudes avérés sont peu nombreux (de l'ordre de 1000) et peuvent être de différentes natures (ce qui conduit à considérer plusieurs classes de fraude). Chaque individu fraudeur placé dans un ensemble de validation est un individu perdu et de potentielle grande valeur.

Un estimateur peu fin. – La capacité de discernement du risque empirique est limitée. Par exemple, sur le problème de la fig.4.1, le risque empirique n'est pas un bon indicateur

pour choisir le meilleur classifieur. Comme il ne permet de distinguer que des différences de classe majoritaire et comme la classe majoritaire est la même partout pour ce jeu de données, les performances des classifieurs seront toutes identiques. Nous revenons sur ce point dans la suite.

De plus, numériquement, ce taux d'erreur prend au plus N valeurs différentes. Pour chaque instance, la seule information prise en compte est l'appartenance ou non à la "bonne" classe. Si le nombre de classifieurs à comparer est grand devant N, le risque empirique ne peut séparer tous les classifieurs. A titre d'exemple, la sélection d'un sous ensemble de variables descriptives parmi D variables nécessite de considérer 2^D classifieurs, nombre qui dépasse souvent de très loin le nombre d'individus dont on dispose.

4.1.2 Régularisation structurelle du risque empirique

Le risque empirique calculé sur l'ensemble d'apprentissage fournit une estimation biaisée de la probabilité d'erreur d'un classifieur et le protocole de validation possède ses limites. La question de la correction du biais peut être résolue sans passer par ce protocole. Une solution consiste à adopter le point de vue de la théorie de la régularisation. C'est l'objet des travaux de Vapnik que de le développer et nous introduisons maintenant ses résultats.

Un problème mal posé. – La minimisation du risque empirique constitue un problème *mal-posé* : la stabilité de la solution n'est pas garantie. Plus précisément, pour des données proches D_1 et D_2, les solutions obtenues à partir de D_1 ne sont pas nécessairement proches de celles obtenues à partir de D_2.

Régularisation. – Depuis les travaux de Tikhonov [Tikhonov, 1963], on sait que certains problèmes peuvent être ramenés à des problèmes bien posés si l'on cherche à minimiser un risque régularisé. Dans le cas du risque empirique, cela revient à minimiser une expression de la forme

$$R^*(f) = R_{emp}(f) + \lambda\Omega(f), \tag{4.3}$$

où Ω est une *fonctionnelle régularisante* et λ un *coefficient de régularisation.*

Le travail de Vapnik ramène le problème de l'apprentissage par minimisation du risque empirique à un problème bien posé. Il propose une fonctionnelle régularisante pour le risque empirique. Ce travail conduit au principe de minimisation du risque structurel. Nous présentons ce principe. La présentation de la régularisation nécessite l'introduction de quelques notions.

Coefficient de pulvérisation [Vapnik, 1996]. – Soit $D = (x_m, y_m)_M$ un échantillon de M réalisations d'un couple de variables aléatoires (X, Y), X étant à valeurs dans un ensemble $\mathbb{X}$ et Y étant à valeurs dans $\{0, 1\}$. Soit $\mathcal{H}$ un ensemble de classifieurs. Le *coefficient de pulvérisation* de $\mathcal{H}$, noté $N^{\mathcal{H}}(D)$ est le cardinal de l'ensemble $\{(f(x_1), \ldots, f(x_M));\ f \in \mathcal{H}\}$. Intuitivement, ce coefficient quantifie le nombre de partitions en deux groupes de l'échantillon réalisées par les classifieurs de $\mathcal{H}$.

VC-dimension [Vapnik, 1996]. – La *fonction de croissance* associée à $\mathcal{H}$ est la fonction de $\mathbb{N}$ dans $\mathbb{R}$, notée $G^{\mathcal{H}}$, définie par

$$\forall M \in \mathbb{N}, G^{\mathcal{H}}(M) = \log \sup_{D,\#D=M} N^{\mathcal{H}}(D). \tag{4.4}$$

La *VC-dimension* de $\mathcal{H}$, notée $VC(\mathcal{H})$, est alors la plus grande valeur de M pour laquelle $G^{\mathcal{H}}(M) = M \log 2$. La VC-dimension peut être infinie.

Théorème [Vapnik, 1996]. – Soit $D = (x_n, y_n)$ un échantillon de taille N. Si la VC-dimension de $\mathcal{H}$ est finie, pour tout classifieur $f \in \mathcal{H}$ on a, avec une probabilité supérieure à $1 - \eta$,

$$p(f(X) \neq Y) \leq R_{emp}(f) + \sqrt{\frac{VC(\mathcal{H})(1 + \log(2N/VC(\mathcal{H}))) - \log(\eta/4)}{N}}. \tag{4.5}$$

Le principe de minimisation du risque structurel propose de minimiser conjointement le risque empirique et la borne sur l'écart entre cette erreur empirique et l'erreur théorique.

Principe de minimisation du risque structurel [Vapnik, 1996]. – Soit $D = (x_n, y_n)_N$ un échantillon de taille N. Soit $\mathcal{H}_1, \ldots, \mathcal{H}_K, \ldots$ une suite infinie croissante d'ensembles de classifieurs telle que $\mathcal{H} = \bigcup \mathcal{H}_K$ et telle que $VC(\mathcal{H}_K)$ est finie pour tout K. Pour $f \in \mathcal{H}$, on note $VC(f)$ la VC-dimension du plus petit ensemble $\mathcal{H}_K$ contenant f. Pour un risque η fixé, le principe de minimisation du risque structurel (principe SRM en abrégé) préconise de sélectionner le classifieur f_{opt} tel que

$$f_{opt} = \arg\min_{f \in \mathcal{H}(D)} R_{emp}(f, D) + \sqrt{\frac{VC(f))(1 + \log(2N/VC(f))) - \log(\eta/4)}{N}}. \tag{4.6}$$

Notons que la suite des VC-dimensions est croissante. Le risque empirique diminue avec l'augmentation de la VC-dimension alors que, N étant fixé, la régularisation augmente avec elle. La somme de ces deux termes définit le *risque structurel*.

Limites du principe SRM. – Un reproche souvent fait à l'encontre du principe SRM est de travailler avec une borne valide quelle que soit la distribution des données, ce qui peut la rendre inintéressante pour un problème particulier car trop lâche. De plus, il est souvent difficile de calculer la VC-dimension en dehors du cas linéaire. Si l'on arrive parfois à majorer cette quantité, cela éloigne d'autant la borne optimisée de la réalité du jeu de données.

De plus, en l'état, le principe SRM ne permet pas de travailler avec un ensemble de classifieurs dépendant des données. Notons également que tout problème à plus de deux classes doit être transformé afin d'être ramené à un problème à deux classes. La théorie ne considère en effet que ce cas.

4.1.3 Evaluation SRM des partitions de Voronoi

Le principal obstacle à l'application du principe SRM au cas des partitions de Voronoi est le suivant : les hypothèses sont dépendantes de l'échantillon de données, ce dont ne tient pas compte la théorie initiale. Il est proposé dans [Cannon *et al.*, 2002] une adaptation des concepts de la théorie de Vapnik afin de traiter ce cas.

Notation. – Pour un échantillon $D = (x_n, y_n)$, on note $D^{(x)} = \{x_n\}$ et $D^{(y)} = \{y_n\}$.

Notation. – Soit $f : \mathbb{X} \to \{0, 1\}$ un classifieur. On note $I(f) = \{x \in \mathbb{X};\ f(x) = 1\}$.

Notation. – Soit $\mathcal{H}$ un ensemble de classifieurs. On se donne une application associant à tout échantillon D un ensemble $\mathcal{H}(D)$ inclus dans $\mathcal{H}$.

Définition [Cannon *et al.*, 2002]. – Soit $D = (x_n, y_n)_N$ un échantillon de N réalisations indépendantes d'un couple de variables aléatoires (X, Y), X étant à valeurs dans un ensemble $\mathbb{X}$ et Y étant une variable binaire. Si $M \leq N$, on note $\mathcal{N}_M(D)$ le cardinal suivant :

$$\mathcal{N}_M(D) = \#\{D_M^{(x)} \bigcap I(f);\ D_M \subset D, \#D_M = M \text{ et } f \in \mathcal{H}(D_M)\}. \tag{4.7}$$

Intuitivement, ce coefficient quantifie le nombre de partitions en deux groupes de l'échantillon D réalisées par les classifieurs définis à partir d'un échantillon de taille M inclus dans D.

Définition [Cannon *et al.*, 2002]. – Pour $M \leq N$, on définit le *coefficient de pulvérisation* $\mathcal{S}_{M/N}$ de $\mathcal{H}$ par la relation

$$\mathcal{S}_{M/N} = \sup_{D, \#D=N} \mathcal{N}_M(D). \tag{4.8}$$

Notations [Cannon *et al.*, 2002]. – Supposons donnée une application de $\mathcal{H}$ dans $\mathbb{N}$. Si $\mathcal{H}$ est un ensemble de partitions, cette application peut associer à une partition son nombre de groupes, par exemple. Pour un échantillon D et $K \in \mathbb{N}$, notons $\mathcal{H}_K(D)$ l'ensemble des éléments de $\mathcal{H}(D)$ dont l'image est K par cette application. Notons $\mathcal{H}_K$ l'union des $\mathcal{H}_K(D)$ et, pour $M \leq N$, $\mathcal{S}_{M/N}^{(K)}$ le coefficient de pulvérisation de $\mathcal{H}_K$.

Dès lors, il est possible de mettre en place un processus de décision analogue au principe SRM pour traiter des hypothèses dépendantes des données.

Principe SRM avec dépendance aux données [Cannon *et al.*, 2002]. – Il est préconisé de sélectionner le classifieur f de $\mathcal{H}$ minimisant la quantité $R_{emp}(f) + r(K, M)$, où

$$r(K, M) = \sqrt{2 \frac{\log eM}{M \log M} \log \mathcal{S}_{M/N}^{(K)}}, \tag{4.9}$$

avec $M = N/2$ et sous la condition que $f \in \mathcal{H}_K$. e désigne le nombre de Néper.

Nous ne précisons pas ici les conditions d'application du principe et engageons le lecteur intéressé à consulter l'article [Cannon *et al.*, 2002]. Nous notons simplement que ces conditions sont vérifiées par les classifieurs que nous définissons maintenant.

Définition. – Soit (X, Y) un couple de variables aléatoires, X étant à valeurs dans un ensemble $\mathbb{X}$ et Y étant une variable binaire. Soit $D = (x_n, y_n)_N$ un échantillon de N réalisations indépendantes de (X, Y). Soit δ une métrique sur $\mathbb{X}$. Pour tout sous-ensemble de $D^{(x)}$ de cardinal K, en tranchant les égalités, on obtient une partition de Voronoi de $\mathbb{X}$ en K cellules.

Le classifieur associé à cette partition est défini comme suit. Dans chaque cellule, on détermine l'étiquette majoritaire sur l'ensemble des individus de l'échantillon appartenant à la cellule. Pour un individu quelconque, on détermine sa cellule d'appartenance (*i.e.* on recherche le plus proche parmi les K prototypes) et on lui attribue l'étiquette majoritaire dans cette cellule. Nous notons $\mathcal{H}_K(D)$ l'ensemble des classifieurs ainsi obtenus à partir d'un sous-ensemble de cardinal au plus K de D. En reprenant les notations précédentes, $\mathcal{H}_K$ désigne l'union des $\mathcal{H}_K(D)$.

Notons qu'on ne peut appliquer le simple principe de minimisation du risque empirique. En effet, cela conduirait à sélectionner l'hypothèse la plus complexe, de risque empirique nul : 1 individu par cellule, 1 cellule pour chaque individu. Afin d'appliquer le principe SRM, nous déterminons une borne sur le coefficient de pulvérisation.

Proposition 4 (Borne sur le coefficient de pulvérisation) *Pour $K \leq M$, le coefficient de pulvérisation de $\mathcal{S}_{M/N}^{(K)}$ de $\mathcal{H}_K$ est borné par $\binom{N}{M} \sum_{k=1}^{K} \binom{N}{k}$.*

Preuve. – Par définition du coefficient de pulvérisation :

$$\mathcal{S}_{M/N}^{(K)} = \sup_{D, \#D=N} \mathcal{N}_M^{(K)}(D), \tag{4.10}$$

D étant un échantillon de cardinal N et $\mathcal{N}_M^{(K)}(D)$ le cardinal $\mathcal{N}_M(D)$ défini relativement à $\mathcal{H}_K$. On a la majoration suivante du cardinal $\mathcal{N}_M^{(K)}(D)$:

$$\begin{aligned} \mathcal{N}_M^{(K)}(D) &= \#\{D_M^{(x)} \bigcap I_f;\ D_M \subset D, \#D_M = M \text{ et } f \in \mathcal{H}_K(D_M)\} \\ &\leq \# \bigcup_{D_M \subset D, \#D_M = M} \ \bigcup_{f \in \mathcal{H}_K(D_M)} \{D_N \bigcap I_f\} \\ &\leq \binom{N}{M} \sum_{k=1}^{K} \binom{N}{k}. \end{aligned} \tag{4.11}$$

■

Nous sommes maintenant en mesure de proposer un critère d'évaluation de type SRM pour les classifieurs définis à partir des partitions de Voronoi.

Proposition d'un critère d'évaluation SRM. – Soit δ une matrice de Gram sur $\mathcal{I}$. Pour un élément H de $\mathcal{H}_\delta(\mathcal{I})$ de cardinal K, notons $f_H \in \mathcal{H}_K$ le classifieur induit. Si $K \leq \frac{N}{2}$, nous proposons d'évaluer l'hypothèse H par le *critère SRM* suivant (e désigne le nombre de Néper) :

$$c_{\delta,Y}^{SRM}(H) = R_{emp}(f_H) + \sqrt{\frac{4}{N}\frac{\log eN}{\log N}\left(\log\binom{N}{N/2} + \log\sum_{k=1}^{K}\binom{N}{k}\right)}. \tag{4.12}$$

4.1.4 Comparaison avec le critère MDL

Nous avons appliqué la théorie de l'apprentissage statistique afin d'obtenir un critère d'évaluation des partitions de Voronoi. Plus précisément, nous avons proposé une régularisation du risque empirique des classifieurs définis à partir de ces partitions. Le critère proposé prend automatiquement en charge la gestion du sur-apprentissage. L'évaluation respecte les deux exigences suivantes : absence de paramètre à ajuster et non utilisation d'un ensemble de validation. Nous illustrons la différence de comportement entre le critère SRM et le critère MDL à l'aide d'une expérience sur données synthétiques.

Description de l'expérience. – Nous générons uniformément 2000 points dans un carré et considérons deux classes dont la distribution conditionnelle est $(0.9, 0.1)$ dans les coins supérieur droit et inférieur gauche, $(0.6, 0.4)$ dans les autres coins. Dans ce cas, la classe majoritaire est la même en tout point du carré.

Nous définissons la notion de *trajectoire* d'optimisation. Les 2000 individus sont tout d'abord ordonnés, aléatoirement et uniformément. La suite obtenue est parcourue, et l'individu considéré à chaque étape est éliminé. A chaque étape, la partition basée sur l'ensemble des individus restant est évaluée pour un critère donné. Nous visualisons sur la fig.4.2 les trajectoires obtenues pour chacun des critères SRM et MDL.

Différence de comportement. – Le critère SRM est un critère fondé sur des bases théoriques solides, celles de la théorie de l'apprentissage statistique. Mais, pour l'obtenir, il a fallu faire des hypothèses. L'utilisation du critère n'est donc fondée que si :

- le problème de classification est binaire,
- la partition évaluée est constituée par moins de $N/2$ cellules,

Dans tout autre cas, son utilisation n'est plus qu'heuristique. De plus, il repose sur la considération d'une borne, susceptible d'être éloignée de la réalité d'un jeu de données particulier.

Plus que ces limites techniques d'utilisation, c'est le manque de finesse d'un critère basé sur la probabilité d'erreur que nous tenons à souligner. Le critère SRM, parce qu'il emploie le risque empirique comme mesure de qualité, ne s'intéresse qu'au caractère majoritaire d'une étiquette. Il est donc aveugle aux différences de densité conditionnelle du moment que la classe majoritaire reste majoritaire. Sur le jeu de données considéré, cela conduit à sélectionner la partition de Voronoi constituée par une unique cellule.

Le critère MDL autorise une détection plus fine : il ne tient pas uniquement compte du caractère majoritaire d'une étiquette. En considérant la distribution des étiquettes, il

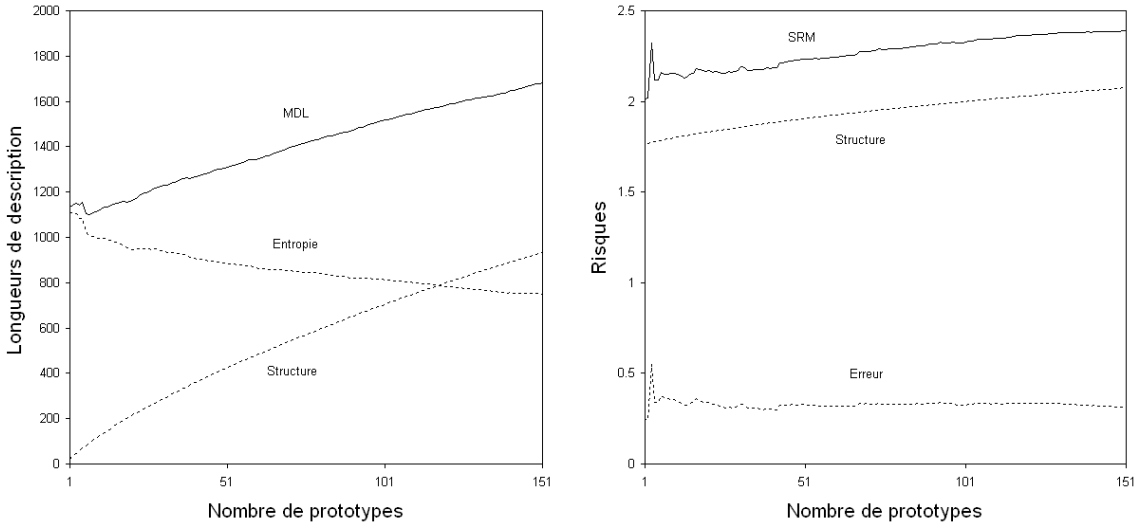

FIG. 4.2 – Les trajectoires d'optimisation des critères MDL et SRM sur un jeu de données synthétiques. La classe majoritaire est la même en tout point du plan, mais sa probabilité d'apparition varie. La meilleure partition, au sens du critère SRM, est celle constituée par un unique groupe.

permet de distinguer plusieurs comportements sur le jeu de données et conduit à sélectionner une partition de Voronoi constituée de quatre groupes. Parce qu'il tient compte dans chaque cellule de la distribution des étiquettes et pas seulement de l'étiquette majoritaire, le critère MDL est plus fin qu'un critère basé sur l'erreur empirique.

Remarque. – Notons que le critère SRM produit les résultats attendus lorsque la classe majoritaire présente des différences significatives sur le jeu de données. Par exemple, si on génère uniformément 2000 points dans un carré et considère deux classes dont la distribution conditionnelle est $(1, 0)$ dans les coins supérieur droit et inférieur gauche, $(0, 1)$ dans les autres coins, le critère SRM conduit à sélectionner une partition de Voronoi en quatre cellules.

4.2 Inférence bayésienne et densités de probabilité

Au cours des années 30, Sir Ronald Aylmer Fisher institutionnalisa le principe de maximisation de la vraisemblance, notamment à travers son article [Fisher, 1936]. La mise en concurrence de différents modèles de densité montre bien vite la limite de ce principe : son biais en faveur des modèles plus complexes. L'approche bayésienne développée par Schwartz [Schwartz, 1978] conduit à une pénalisation des modèles complexes. Nous présentons cette approche et déduisons un critère d'évaluation des densités induites par les partitions de Voronoi.

4.2.1 Maximum de vraisemblance

Commençons par introduire le cadre probabiliste usuel de la statistique [Foata and Fuchs, 2003].

Modèle statistique. – On appelle *modèle statistique* toute famille $\mathcal{M}$, indicée par un ensemble Θ, de probabilités sur un même univers Ω. Le modèle est dit *dominé* par une mesure positive et σ-finie si tout élément p de $\mathcal{M}$ admet une densité f par rapport à μ. Autrement dit, on peut écrire pour tout événement A

$$p(A) = \int_A f \, \mathrm{d}\mu. \tag{4.13}$$

Vraisemblance. – Soit $\mathcal{M}$ un modèle statistique dominé indicé par un ensemble Θ : $\mathcal{M} = \{p_\theta;\ \theta \in \Theta\}$. Pour $\theta \in \Theta$, soit f_θ une densité de p_θ par rapport à la mesure dominante. La *fonction de vraisemblance* du modèle $\mathcal{M}$ est la fonction $L : \Theta \times \Omega \to \mathbb{R}$ définie par.

$$\forall \theta \in \Theta, \forall \omega \in \Omega,\ L(\theta, \omega) = f_\theta(\omega). \tag{4.14}$$

Modèle d'échantillonage. – Soit $\mathcal{M}$ un modèle statistique. Pour $N \in \mathbb{N}^*$, le produit de N modèles identiques à $\mathcal{M}$ est appelé *modèle d'échantillon empirique*, et on le note $\mathcal{M}^N$. Il correspond au cas de N épreuves indépendantes et identiques. Si $\mathcal{M}$ est dominé par une mesure μ, alors $\mathcal{M}^N$ est dominé par la mesure produit $\mu^{\otimes N}$. Dans ce cas, la fonction de vraisemblance de $\mathcal{M}^N$ s'écrit :

$$\forall \theta \in \Theta, \forall \omega_1, \ldots, \omega_N \in \Omega,\ L(\theta, \omega_1, \ldots, \omega_N) = \prod_{n=1}^{N} f_\theta(\omega_n). \tag{4.15}$$

Principe du maximum de vraisemblance. – Soit $D = (x_n)$ un échantillon de N réalisations d'une variable aléatoire X à valeurs dans un ensemble $\mathbb{X}$. Pour un modèle statistique $\mathcal{M} = \{p_\theta;\ \theta \in \Theta\}$, il est préconisé de sélectionner le paramètre θ_{opt} maximisant la vraisemblance du modèle d'échantillon empirique

$$\theta_{opt} = \arg\max_{\theta \in \Theta} L(\theta, D). \tag{4.16}$$

Lien avec l'inférence bayésienne. – Soit D des données et soit $\mathcal{H}$ un ensemble de densités de probabilité. Rappelons que réaliser une inférence bayésienne, c'est se donner une distribution de probabilité p, dite a priori, sur l'ensemble des hypothèses $\mathcal{H}$ et, pour chaque hypothèse H, une fonction q_H, dite de vraisemblance, sur les données. Dès lors, l'adoption du principe MAP et l'application de la formule de Bayes conduisent à sélectionner l'hypothèse qui maximise $p(H)q_H(D)$.

La fonction de vraisemblance L du modèle d'échantillon empirique constitue une définition possible de la fonction de vraisemblance q nécessaire à l'inférence bayésienne. Depuis les travaux de Fisher, cette définition s'est imposée et c'est ce qui constitue un avantage de l'estimaton de densités : pour mettre en place une inférence bayésienne dans ce cas, il suffit de définir une distribution a priori.

4.2.2 A priori usuels sur un ensemble de densités

La question de la construction d'une distribution a priori a plutôt tendance à diviser. Nous n'entrons pas ici dans le débat et nous contentons de rester factuel en présentant différents types d'a priori. Nous nous référons pour cela à la synthèse de Jean-Pierre Florens contenue dans [Droesbeke *et al.*, 2002]. Le but est de montrer que les approches ne sont pas adaptées au cas que nous considérons.

A priori informatif et a priori non informatif. – Un point de vue subjectiviste considère que l'a priori doit être l'expression analytique de l'opinion des experts avant observation des données : l'approche et l'a priori sont dits *informatifs*. Le second est moins subjectif et vise à produire un a priori en ne s'aidant d'aucune information sur le phénomène étudié : l'approche et l'a priori sont dits *non informatifs*.

A priori naturel conjugué. – Parmi les approches informatives, lorsqu'on accorde plus d'importance à la faisabilité des calculs qu'à la pertinence de l'a priori, on est amené à considérer des familles d'a priori dits *naturels conjugués* de la vraisemblance. L'idée sous-jacente consiste à voir l'inférence bayésienne comme une application transformant une densité a priori $p(H)$ sur les hypothèses en une densité a posteriori $p_D(H)$ (définie par la règle de Bayes) et à demander que p et p_D appartiennent à la même famille de densités.

Prenons un exemple. Les données sont N valeurs (x_n) binaires dont une proportion π est nulle, et l'ensemble des hypothèses $\mathcal{H}$ est l'ensemble des densités binomiales de paramètre π et N. Si on adopte comme distribution a priori sur les paramètres (π, N) une loi bêta de paramètres (a_0, b_0), alors la distribution a posteriori est encore une loi bêta, de paramètres $a_0 + \pi N$ et $b_0 + (1 - \pi)N$. Les lois bêta sont donc les naturelles conjuguées des lois binomiales.

Dans une approche informative, la construction de la distribution a priori est contrainte par l'expert (sa connaissance, sa volonté de préserver l'aspect calculatoire, etc). Dans le cas non informatif, on met en place des contraintes génériques, souvent mathématiques, qui n'ont aucun lien avec le phénomène étudié.

A priori invariant. – Les a priori *invariants* répondent à une contrainte d'invariance vis-à-vis d'un ensemble de transformation des hypothèses. Pour un ensemble de transformations, un travail théorique doit être effectué afin d'assurer l'unicité de l'a priori invariant par ces transformations. De plus, la construction d'un tel a priori pour utilisation effective est difficile en dehors des hypothèses paramétriques classiques (gaussiennes, par exemple).

A priori de Jeffreys. – L'*a priori de Jeffreys*, non informatif, concerne des espaces d'hypothèses qui sont des ensembles de densités f_θ paramétrées par un vecteur θ de nombres réels. Il exploite la notion de *matrice d'information de Fisher* $I(\theta)$ associée à une hypothèse f_θ. Le coefficient (i, j) de cette matrice est l'espérance du produit scalaire de $\frac{\delta \log f_\theta}{\delta \theta_i}$ et $\frac{\delta \log f_\theta}{\delta \theta_j}$. L'a priori de Jeffreys est défini à partir de la racine carrée du déterminant de la matrice de Fisher.

Cet a priori vérifie une propriété d'invariance vis-à-vis d'une statistique exhaustive de l'échantillon. Son application est limitée à des ensembles de densités paramétriques vérifiant des hypothèses fortes de régularité sur le paramétrage. En pratique, excepté pour des hypothèses simples, l'information de Fisher n'est qu'approximée.

A priori de référence. – Citons l'analyse "de référence" présentée dans [Bernardo and Smith, 2000]. L'idée consiste à mesurer l'écart à l'indépendance résultant du choix d'un a priori. Plus précisément, dès qu'on fixe la vraisemblance q_H, on associe à tout a priori p le produit $p(H)q_H(D)$, correspondant à la loi jointe, et le produit $p(H)q(D)$ (avec q la densité marginale sur les données obtenue en sommant $q_H(D)$ sur les hypothèses), correspondant à la loi jointe sous l'hypothèse d'indépendance. La divergence de Kullback permet de comparer $p(H)q_H(D)$ à $p(H)q(D)$ et l'*a priori de référence* est celui maximisant cette divergence. Ce problème de maximisation n'admet pas de solution générale.

Limite des approches classiques. – Nous avons présenté différentes catégories d'a priori adaptées au cas de l'estimation de densités. Dans la recherche d'une pénalisation de la fonction de vraisemblance, nous abandonnons ces pistes au moins pour la raison suivante : ils conduisent à des critères paramétriques. Par exemple, l'emploi d'un a priori naturel conjugué nécessite de fixer les paramètres de la loi a priori.

Nous nous orientons vers une approche différente de celles exposées ci-dessus, issue des travaux de Schwartz sur la sélection de modèles.

4.2.3 La sélection bayésienne de modèles et le critère BIC de Schwartz

L'inférence bayésienne décrite précédemment s'applique à un ensemble homogène de densités. C'est le cas par exemple des mélanges de gaussiennes. Lorsque l'on cherche à comparer des densités aux formes différentes, par exemple des mélanges de gaussiennes et des modèles markoviens, on entre dans le contexte de la sélection de modèles.

Sélection de modèles.– L'ensemble des hypothèses $\mathcal{H}$ est l'union d'un nombre fini d'ensembles de densités $\mathcal{H}_1, \ldots, \mathcal{H}_K$, appelés *modèles*. Chaque modèle correspond à une forme de densité particulière et il s'agit de déterminer le modèle le plus adapté.

Approche bayésienne [Schwartz, 1978]. – Soient D des données. Il est préconisé de sélectionner le modèle le plus probable connaissant les données :

$$k_{opt} = \arg\max_{1 \leq k \leq K} \; p_D(k). \tag{4.17}$$

C'est alors la question de la définition de la distribution de probabilité $p_D(k)$ qui est posée. La formule de Bayes permet de définir p_D de manière indirecte. Elle consiste à écrire

$$p_D(k) = \frac{p(k)p_k(D)}{p(D)}. \tag{4.18}$$

Un a priori uniforme est posé sur les modèles (*i.e.* $p(k) = 1/K$). En constatant que la distribution $p(D)$ est indépendante du modèle, on est ramené à définir le terme $p_k(D)$ et à sélectionner le modèle

$$k_{opt} = \underset{1 \leq k \leq K}{\arg\max}\ p_k(D). \tag{4.19}$$

Obtention de la densité. – Quel sens donner à $p_k(D)$? L'idée consiste à intégrer la vraisemblance sur les éléments du modèle k puis à utiliser la formule de Bayes :

$$\begin{aligned} p_k(D) &= \int_{\mathcal{H}_k} p_H(D)\ \mathrm{d}H \\ &= \int_{\mathcal{H}_k} p_{k,H}(D) p_k(H)\ \mathrm{d}H. \end{aligned} \tag{4.20}$$

Il reste alors à définir et calculer cette intégrale. En pratique, on se contente d'une approximation et du critère suivant.

Critère BIC [Schwartz, 1978]. – Les modèles sont supposés paramétriques : pour $k \in [\![1, K]\!]$, chaque élément de $\mathcal{H}_k$ est déterminé par un unique vecteur de paramètres réels de dimension d_k. Sous des hypothèses de dérivabilité de la vraisemblance vis-à-vis des paramètres, l'intégrale définissant $p_k(D)$ est approximée, et on obtient en éliminant les termes constants (*i.e.* ne dépendant pas de N)

$$\log p_k(D) = \log L(\widehat{\theta}_k, D) - \frac{d_k}{2} \log N, \tag{4.21}$$

où $\log L(\widehat{\theta}_k, D)$ est la log-vraisemblance de l'hypothèse la plus vraisemblable du modèle k, de paramètre $\widehat{\theta}_k$.

L'obtention du critère BIC repose sur la considération de modèles de densités paramétriques et résulte d'une approximation asymptotique nécessitant des hypothèses fortes de régularité vis-à-vis des paramètres. Sa popularité découle de son interprétation a posteriori, qui permet de faire de l'inférence bayésienne à peu de frais.

Interprétation a posteriori du critère. – Le critère BIC, mis en place afin d'opérer une sélection de modèle, est utilisé en pratique pour faire de l'estimation de densité. En effet, il consiste en une pénalisation de la log-vraisemblance par une quantité dépendante du nombre de paramètres du modèle. L'approche de Schwartz est souvent interprétée de la manière heuristique suivante : pénaliser la log-vraisemblance par le nombre de paramètres du modèle multiplié par $-\frac{1}{2}\log N$ et choisir l'hypothèse maximisant le critère obtenu.

- les hypothèses sont des densités de probabilité,
- la vraisemblance est définie comme la probabilité que les données soient générées par une densité,
- l'hypothèse i.i.d. est adoptée pour le calcul de la vraisemblance,
- dans le cas fini, un a priori uniforme est adopté,
- sinon, la faisabilité des calculs est privilégiée.

FIG. 4.3 – Us et coutumes de l'évaluation bayésienne pour l'estimation de densité

4.2.4 Evaluation BIC heuristique des partitions de Voronoi

Nous avons posé au chapitre 3 les grandes lignes de l'inférence bayésienne. Dans ce qui précède, nous avons présenté le contexte usuel d'application de ce principe d'inférence, celui de l'estimation de densités. Nous résumons sur la fig.4.3 les habitudes de l'inférence bayésienne dans ce contexte.

Dans le cadre supervisé, ce sont les densités conditionnelles qui nous intéressent. Pour pouvoir traiter les partitions de Voronoi dans un cadre supervisé, nous associons à chaque partition une telle densité. C'est seulement alors que nous pourrons proposer un critère d'évaluation de type BIC des partitions de Voronoi.

Notation. – Soit (X, Y) un couple de variables aléatoires, X étant à valeurs dans un ensemble $\mathbb{X}$ et Y étant à valeurs dans un ensemble fini $\mathbb{Y}$. Soit $D = (x_n, y_n)$ un échantillon de N réalisations de (X, Y) et notons $D^{(x)} = \{x_n\}$ et $D^{(y)} = \{y_n\}$.

Densité conditionnelle. – Soit f une densité de probabilité sur $\mathbb{X} \times \mathbb{Y}$. Soient $f_{\mathbb{X}}$ la densité marginale de f sur $\mathbb{X}$ (obtenue par sommation sur $\mathbb{Y}$ en tout $x \in \mathbb{X}$, d'après le théorème de Fubini). On définit la *densité conditionnelle* $f(\cdot/\cdot)$ comme l'application de $\mathbb{Y} \times \mathbb{X}$ dans $\mathbb{R}_+$ telle que :

$$f(y/x) = \frac{f(x, y)}{f_{\mathbb{X}}(x)}. \tag{4.22}$$

En pratique, la valeur $f(y/x)$ en un point $(x, y) \in \mathbb{X} \times \mathbb{Y}$ s'interprète comme la probabilité d'obtenir y en connaissance de x.

Pour proposer un critère bayésien d'évaluation des partitions de Voronoi, nous adoptons l'approche heuristique suivante, inspirée par la mise en place du critère de Schwartz : pénaliser la log-vraisemblance conditionnelle par $-\frac{1}{2} \log N$ fois le nombre de paramètres. Nous devons pour cela définir la log-vraisemblance conditionnelle, définir l'ensemble des densités conditionnelles que nous considérons, calculer la log-vraisemblance conditionnelle pour ces hypothèses et enfin déterminer le nombre de paramètres caractérisant une hypothèse.

Vraisemblance conditionnelle. – Pour $D = (x_n, y_n)$ un échantillon de N réalisations indépendantes de (X, Y) et pour une densité conditionnelle $f(\cdot/\cdot)$ sur $\mathbb{X} \times \mathbb{Y}$, on appelle

vraisemblance conditionnelle de l'échantillon D relativement à f la quantité :

$$L(f, D^{(y)}/D^{(x)}) = \prod_{n=1}^{N} f(y_n/x_n). \tag{4.23}$$

Pour faciliter les manipulations algébriques, on considère souvent la *log-vraisemblance conditionnelle* $\log L(f, D^{(y)}/D^{(x)})$.

Définition. – Soit (X, Y) un couple de variables aléatoires, X étant à valeurs dans un ensemble $\mathbb{X}$ et Y étant à valeurs dans un ensemble fini. Soit $D = (x_n, y_n)$ un échantillon de N réalisations indépendantes de (X, Y). Soit δ une métrique sur $\mathbb{X}$. A tout sous-ensemble de $D^{(x)}$ de cardinal K est associée une partition de Voronoi de $\mathbb{X}$ en K cellules, les égalités de distance étant tranchées de manière déterministe.

La densité définie à partir de cette partition est la suivante. Dans chaque cellule, on calcule les fréquences d'apparition des étiquettes sur les individus appartenant à la cellule. Pour un individu quelconque, on détermine sa cellule d'appartenance (*i.e.* on recherche le plus proche parmi les K prototypes) et on lui attribue la distribution fréquentielle des étiquettes de cette cellule. Nous notons $\mathcal{H}_K(D)$ l'ensemble des densités conditionnelles ainsi obtenues à partir d'un sous-ensemble de D de cardinal K.

Calcul de la log-vraisemblance conditionnelle. – Soit $f \in \mathcal{H}_K(D)$ une densité conditionnelle associée à une partition de Voronoi. La log-vraisemblance de f s'écrit :

$$\log L(f, D^{(y)}/D^{(x)}) = \sum_{k=1}^{K} \sum_{j=1}^{J} N_{kj} \log N_{kj} - \sum_{k=1}^{K} N_k \log N_k, \tag{4.24}$$

où J est le nombre de classes cibles, N_{kj} le nombre d'individus de l'échantillon dans la k^{eme} cellule portant la j^{eme} étiquette et N_k le nombre d'individus de l'échantillon dans la k^{eme} cellule.

Preuve. – La distribution des étiquettes est constante sur chaque cellule. Sur la k^{eme} cellule, elle vaut $\frac{1}{N_k}(N_{k1}, \ldots, N_{kJ})$. Dès lors,

$$\begin{aligned} \log L(f, D) &= \sum_{n=1}^{N} \log f(y_n/x_n) \\ &= \sum_{j=1}^{J} \sum_{k=1}^{K} N_{kj} \log \frac{N_{kj}}{N_k}, \end{aligned} \tag{4.25}$$

■

Notons qu'on ne peut appliquer un simple principe de maximisation de la vraisemblance. En effet, cela conduirait à sélectionner l'hypothèse la plus complexe, de log-vraisemblance conditionnelle nulle : 1 individu par cellule, 1 cellule pour chaque individu. Le besoin d'une pénalisation est ici impératif, le sur-apprentissage se manifestant

de manière flagrante. Nous utilisons pour cela le nombre de paramètres, par analogie avec l'interprétation courante du critère BIC.

Proposition d'un critère bayésien. – Pour une densité f_H de $\mathcal{H}_K(D)$ associée à un diagramme de Voronoi H composé de K cellules, le nombre de paramètres est $J(K-1)+1$. En effet, on dispose de K degrés de liberté pour le choix des prototypes, et de $(J-1)(K-1)$ degrés de liberté pour la spécification de la table de contingence (K distributions de J classes à spécifier, en connaissance des effectifs dans les groupes et des effectifs dans les classes). Comme $K + (J-1)(K-1) = J(K-1) - 1$, nous proposons de pénaliser la log-vraisemblance par $-\frac{1}{2}\log N$ fois ce nombre, et donc de minimiser le *critère BIC* suivant :

$$c_{\delta,Y}^{BIC}(H) = \frac{J(K-1)+1}{2}\log N - \sum_{k=1}^{K}\sum_{j=1}^{J} N_{kj}\log N_{kj} + \sum_{k=1}^{K} N_k \log N_k. \tag{4.26}$$

4.2.5 Comparaison avec le critère MDL

En nous plaçant dans le cadre de l'estimation de densités, nous avons déduit un critère d'évaluation des partitions de Voronoi. Plus précisément, nous avons obtenu de manière heuristique un critère analogue au critère de Schwartz, en pénalisant la log-vraisemblance par le nombre de degrés de liberté de la densité conditionnelle induite par une partition de Voronoi. Ce critère prévient automatiquement le sur-apprentissage, et l'évaluation respecte les deux exigences que nous nous étions imposées : absence de paramètre à ajuster et non utilisation d'un ensemble de validation.

Propriété. – D'après l'approximation de Stirling ($\log x! \approx x\log x - x + O(\log x)$), pour une partition de Voronoi $H \in \mathcal{H}_\delta(\mathcal{I})$ et lorsque N tend vers l'infini, on a l'égalité asymptotique :

$$\sum_{k=1}^{K} \log \frac{N_k!}{N_{k1}!\ldots N_{kJ}!} \approx -\sum_{k=1}^{K}\sum_{j=1}^{J} N_{kj}\log N_{kj} + \sum_{k=1}^{K} N_k \log N_k. \tag{4.27}$$

Asymptotiquement, la longueur de description entropique du critère MDL et la log-vraisemblance du critère BIC sont égales au signe près. Cela peut paraître étonnant, dans la mesure où les critères reposent sur des hypothèses d'indépendance différentes : indépendance des cellules pour le critère MDL, et indépendance des individus pour le critère BIC.

La différence entre les critères BIC et MDL se fait dans le cadre non asymptotique ainsi qu'au niveau du terme de pénalisation. Les différences entre ces deux critères ne sont pas les mêmes que celles entre le critère SRM et le critère MDL. Les jeux de données utilisés ici sont donc différents de ceux utilisés à la section précédente.

Longueur de description entropique et log-vraisemblance. – Reprenons l'expérience menée à la section 4.1 permettant d'obtenir une trajectoire d'optimisation de

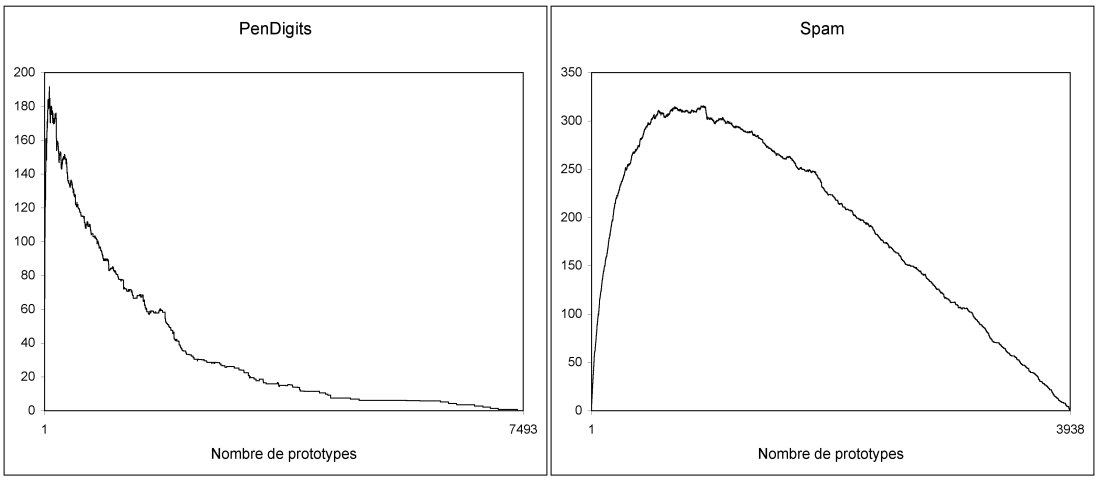

FIG. 4.4 – Différence entre l'opposé de la log-vraisemblance et la longueur de description entropique. Ces quantités sont mesurées sur une trajectoire d'optimisation pour les deux jeux de données PenDigits et Spam.

chacun des critères MDL et BIC. Nous nous intéressons ici uniquement à la partie entropique du premier et à la log-vraisemblance du second. Sur la fig.4.4, nous reportons la différence entre la trajectoire de l'opposé de la log-vraisemblance et la trajectoire de la longueur de description entropique, pour deux jeux de données de l'UCI, PenDigits et Spam.

Dans cette expérience, N est fixé. Elle montre, dans le cas non asymptotique, la différence entre l'hypothèse i.i.d. permettant le calcul de la log-vraisemblance et l'hypothèse d'indépendance des cellules conduisant à la longueur de description entropique. La première est plus forte que la seconde, ce qui se manifeste par une valeur de l'opposé de la log-vraisemblance plus élevée que celle de l'entropie.

Longueur de description structurelle et pénalisation. – Nous reportons sur la fig.4.5 les trajectoires d'optimisation des critères MDL et BIC sur le jeu de données réelles Abalone. C'est un jeu de données composé de 4177 individus et 7 variables descriptives. Chaque individu est un arbre étiqueté par son âge. La classe cible comporte 28 modalités.

On peut de nouveau noter que l'opposé de la log-vraisemblance est plus élevé que la longueur de description entropique. Mais on s'aperçoit également que la pénalisation du critère BIC est plus forte que la longueur de description structurelle du critère MDL.

Le critère BIC étant pénalisé moins finement que le critère MDL, il est moins discriminant. Sur la trajectoire considérée, le critère BIC est minimal pour une partition en 2 cellules. Le critère MDL conduit lui à sélectionner une partition en 13 cellules. Cette expérience a été menée sur d'autres jeux de données. Par exemple, sur PenDigits, le critère BIC est minimal pour une partition en 38 cellules là où le critère MDL est minimal pour 89 cellules.

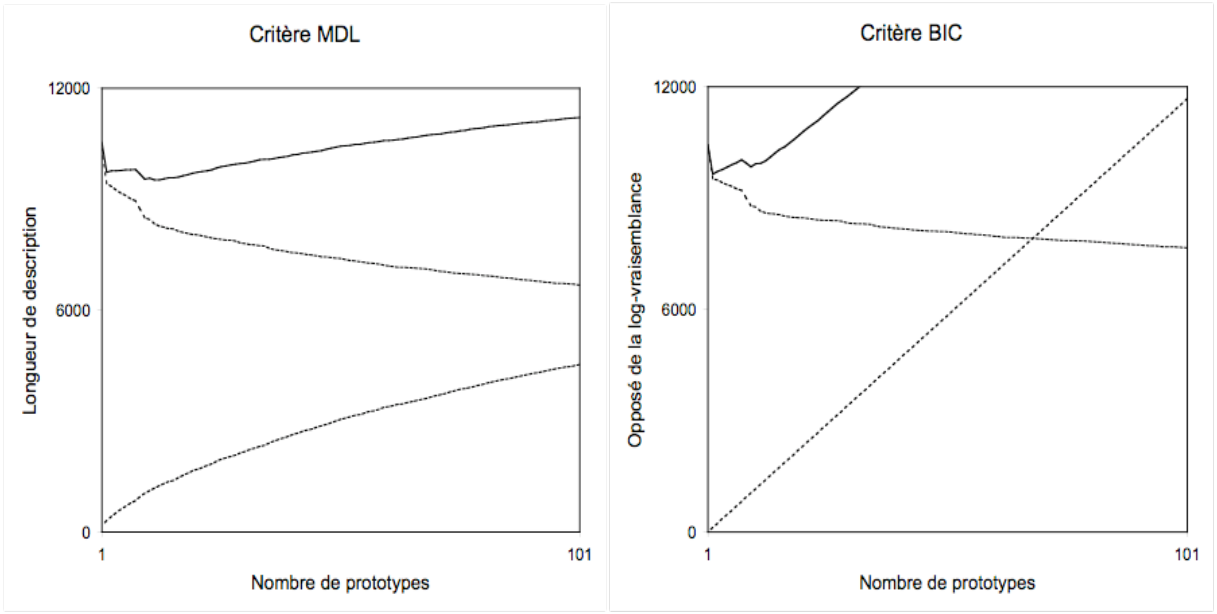

FIG. 4.5 – Trajectoires d'optimisation des critères BIC et MDL sur le jeu de données Abalone. Seules les valeurs obtenues pour un nombre de cellules inférieur à 101 sont reportées.

Le critère BIC que nous avons proposé, comme le critère MDL, tient compte des fréquences de toutes les classes. La log-vraisemblance, tout comme la longueur de description entropique, évalue la distribution des classes plus finement que ne le fait le risque empirique. Mais la pénalisation de la log-vraisemblance est heuristique et ne tient compte que du nombre de paramètres, pas de leur valeur. Cette pénalisation est donc moins fine que celle du critère MDL, qui elle intègre ègalement la valeur de ces paramètres dans sa définition.

4.3 Conclusion

En définissant un classifieur à partir de chaque partition de Voronoi, nous sommes entrés dans le formalisme de la théorie de l'apprentissage statistique pour la classification supervisée. Ce formalisme nécessite quelques aménagements afin de traiter des classifieurs dépendants des données. Nous avons proposé une régularisation non paramétrique du risque empirique et obtenu un critère c^{SRM} d'évaluation supervisée des partitions de Voronoi.

En définissant une densité de probabilité conditionnelle à partir de chaque partition de Voronoi, nous nous sommes placés dans le contexte de l'inférence bayésienne de densités de probabilité. Nous obtenons un ensemble de densités dépendant des données, qui n'entre pas dans le cadre des approches classiques. Nous avons proposé une régularisation heuristique de la vraisemblance en nous inspirant du critère de Schwarz et ainsi obtenu un critère c^{BIC} d'évaluation supervisée des partitions de Voronoi.

Nous disposons ainsi de trois critères d'évaluation supervisée de ces partitions, qui prennent en charge la gestion du sur-apprentissage et respectent nos exigences : ils sont non paramétriques et ne nécessitent pas l'emploi d'un ensemble de validation.

Le critère c^{SRM} est le moins intéressant des trois : en se basant sur l'erreur empirique et en tenant compte uniquement du caractère majoritaire d'une classe, il est le moins fin.

Le critère c^{BIC} est plus proche du critère c^{MDL}. Pour une partition fixée, l'opposé de la log-vraisemblance et la longueur de description entropique sont asymptotiquement égaux. Pour un nombre d'individus fixé, l'hypothèse i.i.d. adoptée pour calculer la log-vraisemblance fait que l'opposé de cette dernière est plus élevé que la longueur de description entropique. Cette différence n'a pas nécessairement une grande influence sur les résultats empiriques. Par contre, le caractère heuristique de l'approche adoptée pour obtenir le critère c^{BIC} conduit à une pénalisation moins fine que celle du critère MDL.

Le critère MDL est donc le plus intéressant car c'est le plus fin. Si les trois critères sont non paramétriques et assurent automatiquement la fiabilité de l'information extraite, c'est le critère MDL qui est capable d'extraire un maximum d'information.

5

Validation expérimentale

Sommaire

Eva cherche la partition de Voronoi de l'ensemble des individus la plus informative vis-à-vis d'une cible catégorielle. A partir de la partition obtenue, un classifieur est défini en attribuant à tout individu l'étiquette majoritaire sur la cellule de Voronoi à laquelle il appartient. Une classification suivant le plus proche prototype est ainsi réalisée. Eva peut être appliquée en tant que méthode de sélection des individus de l'échantillon pour classification suivant le plus proche voisin.

Dans ce chapitre, nous considérons uniquement ce mode d'application et procédons à la validation de notre méthode. Dans la section 5.1, nous décrivons les méthodes concurrentes de sélection d'individus. Dans la section 5.2, nous menons plusieurs expériences sur des données réelles et synthétiques. Nous comparons Eva aux méthodes concurrentes et faisons ressortir ses qualités.

Ce travail a fait l'objet de publications, dans les articles [Ferrandiz and Boullé, 2006c] et [Ferrandiz and Boullé, 2006a], le second étant en cours de relecture.

Définition du contexte. – Dans ce chapitre, nous considérons une variable statique multidimensionnelle numérique $X : \mathcal{I} \rightarrow \mathbb{R}^d$. Autrement dit, nous supposons les individus $\mathcal{I}$ projetés dans l'espace $\mathbb{R}^d$. Pour un individu $n \in \mathcal{I}$, notons $x_n = X(n)$. L'élément x_n est appelé *instance* de l'individu n. Par abus de langage, nous assimilons individus et instances.

Nous parlons de sélection et de construction d'instance pour qualifier les méthodes ici présentées. Les éléments servant de support à la classification sont appelés *prototypes*. Uniquement dans ce chapitre, la notion de prototype est plus générale que celle définie au chapitre 3, dans le sens où les prototypes ici considérés ne sont pas nécessairement des éléments de l'échantillon. Nous évitons ainsi de multiplier les notations et conservons le vocabulaire usuel de la sélection d'instances.

Notons de plus que la mise en œuvre de la classification par le plus proche voisin soulève une autre question fondamentale : celle de la définition d'une notion de similitude. Nous l'occultons complètement ici, puisque notre travail repose sur l'hypothèse qu'on dispose d'une telle notion. Le lecteur intéressé trouvera dans [Wilson and Martinez, 1997a] les définitions d'un grand nombre de mesures de similitude adaptées au cas statique multidimensionnel. Au moins implicitement, nous supposons donc $\mathbb{R}^d$ muni d'une mesure de similitude δ.

5.1 Construction du modèle pour la classification par le plus proche voisin

La règle de classification suivant le plus proche voisin [Fix and Hodges, 1951], [Cover and Hart, 1967] est une méthode classique de classification supervisée. Elle consiste à attribuer à une nouvelle instance l'étiquette de l'instance la plus proche parmi celles constituant l'échantillon. Ainsi, l'apprentissage du modèle ne nécessite aucun autre effort que celui de stocker les données. Si sa simplicité constitue son principal attrait et autorise une étude approfondie de ses propriétés, de nombreuses pistes de recherche ont mené à rendre la phase d'apprentissage plus consistante. La construction de prototypes et la sélection d'instances en sont des exemples.

5.1.1 Méthodes de construction de prototypes

Nous parlons de construction de prototypes pour qualifier les méthodes procédant à une modification des instances de l'échantillon initial. Nous décrivons ici quelques unes de ces méthodes.

Exemplaires généralisés. – Dans [Salzberg, 1991], la notion d'*exemplaire généralisé* est définie : un exemplaire est un parallélépipède rectangle dont les faces sont parallèles aux axes. La classification se base sur un ensemble de tels exemplaires et un algorithme incrémental est proposé afin de construire des exemplaires purs (*i.e.* dans lesquels les instances portent tous la même étiquette), se chevauchant éventuellement. Une étude comparative est menée dans [Wettschereck and Dietterich, 1995], une extension au traitement de données mixtes (numériques et catégorielles) y étant également proposée.

Algorithme de Chang. – Dans [Chang, 1991], un algorithme partant de l'échantillon initial est introduit : à chaque itération, le couple d'instances les plus proches portant une étiquette identique est remplacé par une combinaison linéaire de ces deux instances. Ce procédé de fusion est répété, sous condition de ne pas augmenter le risque empirique.

Quantification. – Les méthodes de *quantification*, regroupées dans [Kohonen, 2001], exploitent également la structure algébrique sous-jacente afin d'optimiser la position de K prototypes pré-sélectionnés. La version incrémentale sélectionne itérativement et uniformément une instance de l'échantillon initial et adapte la position du plus proche prototype selon l'étiquette qu'il porte : si l'instance et le prototype ont même étiquette, le prototype est déplacé en direction de l'instance, sinon il en est éloigné. L'ensemble de prototypes initial résulte d'une sélection aléatoire uniforme dans chaque classe ou de l'application de l'algorithme des K-means [MacQueen, 1967] séparément dans chaque classe.

Regroupement discriminant. – Le *regroupement discriminant* (de l'anglais discriminative clustering) vise à produire des groupes séparant au mieux les classes cibles [Sinkkonen *et al.*, 2002]. Les groupes sont les cellules de Voronoi associées à un ensemble de prototypes. Pour l'obtenir, un principe bayésien est appliqué et des hypothèses paramétriques sont posées. Afin d'autoriser une optimisation par descente de gradient, le critère est lissé en introduisant des fonctions d'appartenance aux cellules de Voronoi qui sont à la fois paramétriques et dérivables. Le critère obtenu ne permet pas de comparer des ensembles de prototypes de tailles différentes. Enfin, la méthode nécessite l'emploi de la métrique euclidienne.

Goulot informationnel. – Le principe du *goulot d'étranglement informationnel* (de l'anglais information bottleneck) a été appliqué au problème de la construction de prototypes [Slonim and Tishby, 2000]. L'idée consiste à extraire une représentation compacte sous forme de partition de Voronoi, sous contrainte de conservation de l'information mutuelle entre le partitionnement effectué et les classes cibles. Le problème est vu comme un problème variationnel, qu'on peut explicitement résoudre sous réserve de connaître la densité jointe génératrice des données. Le critère ne permet pas de comparer des ensembles de prototypes de tailles différentes. Une heuristique gloutonne ascendante d'optimisation construit finalement une hiérarchie de partitions.

5.1.2 Méthodes de sélection d'instances

La sélection d'instances dans l'échantillon initial est une autre voie explorée pour constituer un échantillon représentatif. Rappelons que nous distinguons les instances sélectionnées et celles qui ne le sont pas en qualifiant les premières de *prototypes*.

Règle CNN. – La méthode CNN (pour Condensed Nearest Neighbor) décrite dans [Hart, 1968] est la plus ancienne des méthodes de sélection. Toute instance mal classifiée par son plus proche voisin parmi les prototypes déjà sélectionnés est aussitôt conservée. Ce procédé incrémental est itéré tant qu'il existe des instances mal classifiées par l'ensemble de prototypes. La méthode est *consistante*, dans le sens où tout élément de l'échantillon est finalement bien classifié par son plus proche prototype. Autrement dit, le risque empirique en apprentissage est nul. L'algorithme, décrit à la fig.5.1, est de complexité un $O(N^3)$.

Une amélioration est proposée dans [Gates, 1972] : RNN (pour Reduced Nearest Neighbor). Une fois la règle CNN appliquée, toute suppression d'un prototype préservant le

risque empirique est validée. L'intérêt de cette méthode réside dans sa capacité à produire un sous-ensemble de prototypes de taille minimale relativement à la condition de consistance, sous réserve qu'un tel sous-ensemble soit inclus dans la solution proposée par CNN.

- $H \leftarrow \emptyset$
- **Faire**
  - Parcourir uniformément $\mathcal{I} \setminus H$
  - Pour chaque élément, l'ajouter à H s'il est mal étiqueté par son plus proche voisin dans H
- **Tant que** Il existe un individu mal étiqueté dans $\mathcal{I}$

FIG. 5.1 – L'algorithme CNN.

Règle ENN. – Dans [Wilson, 1972], la règle ENN, pour Edited Nearest Neighbor, est introduite dans le but d'obtenir un comportement asymptotiquement consistant. La méthode est décrémentale et une instance est éliminée si elle est mal classifiée par un vote à la majorité sur ses K plus proches voisins (typiquement, $K = 3$). Sa complexité algorithmique est un $O(KN^2)$. Notons que cette méthode peut être appliquée itérativement.

IB3. – Les précédentes méthodes soit satisfont une propriété de consistance soit possèdent un bon comportement asymptotique. A la fin des années 90, Aha a ramené la question de la réduction significative de l'ensemble des individus au centre du problème. Une série d'algorithmes, d'IB1 à IB5, est proposée dans [Aha *et al.*, 1991]. Les algorithmes IB4 et IB5 opèrent une sélection de variables conjointement à la sélection d'instances et sont ici hors-contexte. IB1 sert de référence et sélectionne toute les instances. IB2 est une version simplifiée de la règle CNN, effectuant une unique passe sur l'ensemble des instances. IB3 introduit une notion d'acceptabilité d'un prototype et une notion de pauvreté d'un prototype : une instance est conservé si elle est mal classifiée par le plus proche prototype acceptable et les prototypes pauvres sont finalement éliminés.

L'acceptabilité et la pauvreté d'un prototype sont déterminées en comparant les bornes de deux intervalles de confiance : si la borne inférieure sur le taux de bonne prédiction est supérieure à la borne supérieure sur la fréquence de son étiquette, le prototype est *acceptable* ; si la borne supérieure sur le taux de bonne prédiction est inférieure à la borne inférieure sur la fréquence de son étiquette, le prototype est *pauvre*. Le seuil de significativité est fixé à 90% pour l'acceptabilité et à 75% pour la pauvreté. Les bornes sont calculées à l'aide de la formule suivante :

$$\frac{c + \frac{z^2}{2n} \pm \sqrt{\frac{c(1-c)}{n} + \frac{z^2}{4n^2}}}{1 + \frac{z^2}{n}}. \tag{5.1}$$

Pour le taux de bonne prédiction d'un prototype, n est le nombre de fois que le prototype a été sollicité pour classifier un individu depuis son introduction dans l'ensemble

des prototypes, c est le nombre de fois qu'un individu à été bien classifié par ce prototype et z est le seuil de significativité. Pour la fréquence d'une étiquette, n est le nombre d'individus considérés jusque là, c est la proportion de ces individus qui portent cette étiquette et z est le seuil de significativité. Au final, l'algorithme, décrit à la fig.5.2, est de complexité un $O(N^2)$.

- $H \leftarrow \emptyset$
- **Pour** $n \in \mathcal{I}$ **Faire**
  - $n' \leftarrow$ le plus proche individu de n acceptable dans H
  - **Si** n et n' ne portent pas la même étiquette, ajouter n' à H
  - **Pour** k in H **Faire**
    - **Si** k est plus proche de n que ne l'est n', incrémenter l'indicateur de sollicitation de k et éliminer k s'il est pauvre
- Eliminer tous les prototypes pauvres

FIG. 5.2 – L'algorithme IB3.

DROP3. – Dans [Wilson and Martinez, 1997b], la notion d'association est utilisée. Pour $K \leq N$ fixé, si n est l'un des K plus proches voisins de n', on dit que n' est *associé* à n. Dès lors, n est éliminé si le nombre de ses associés bien classifiés ne diminue pas après sa suppression. La règle ENN est préliminairement appliquée. De plus, les instances sont considérées par ordre décroissant de distance à la plus proche instance de classe différente. La méthode obtenue, DROP3, est de complexité un $O(KN^2)$.

ICF. – La notion d'association exploitée par DROP3 est légèrement modifiée dans [Brighton and Mellish, 2002]. Au lieu de considérer les K plus proches voisins pour K fixé, il est proposé de prendre en compte les instances plus proches que la plus proche instance d'une autre classe. En dehors de cette modification dans la définition de la notion d'associé, qui rend la méthode non paramétrique, l'algorithme est identique à DROP3. La méthode obtenue est appelée ICF.

PSBoost2. – Dans [Sebban *et al.*, 2002], un test statistique évaluant l'hypothèse de non contribution d'une instance à la classification de ses associés est appliqué. Ce critère est paramétrique et son calcul nécessite l'approximation de la densité de la statistique associée. Une adaptation de l'algorithme AdaBoost permettant de traiter des classifieurs locaux (les prototypes) aboutit à une heuristique de recherche incrémentale. Dans sa version la plus rapide, l'algorithme PSBoost2 est de complexité un $O(KN^2)$.

Explore. – Dans [Cameron-Jones, 1995], la méthode Explore est développée. Elle est constituée par un critère d'évaluation de la qualité prédictive d'un ensemble de prototypes et un algorithme d'optimisation de ce critère. Le critère résulte de l'adoption d'une approche MDL et s'écrit :

$$c(K, N, E) = F(K, N) + K \log_2(J) + F(E, N - K) + E \log_2(J - 1), \qquad (5.2)$$

avec K le nombre de prototypes, N le nombre d'instances, E le nombre d'instances mal classifées par leur plus proche prototype et J le nombre de classe cibles. La quantité $F(U, V)$ mesure la longueur de description nécessaire à la spécification de U instances parmi V et est évaluée par la formule :

$$F(U, V) = \log_2^* \left(\sum_{u=0}^{U} \binom{V}{u} \right), \tag{5.3}$$

où $\log_2^*(x)$ désigne la somme des termes positifs $\log_2(x)$, $\log_2(\log_2(x))$, etc. Les termes $K \log_2(J)$ et $E \log_2(J-1)$ correspondent aux longueurs de description nécessaires à la spécification des étiquettes des K prototypes et des E exceptions respectivement.

Une heuristique est également proposée. Une première phase itérative consiste à ajouter une instance à l'ensemble des prototypes si la valeur du critère diminue. Une fois toutes les instances considérées, tout prototype dont la suppression conduit à la diminution de la valeur du critère est effectivement éliminé. Enfin, 1000 mutations sont successivement évaluées et acceptées si la valeur du critère décroît. Une mutation est soit un ajout d'une instance à l'ensemble des prototypes, soit une suppression d'un prototype, soit un échange entre une instance et un prototype. Au final, la méthode est de complexité un $O(N^3)$.

5.2 Expériences comparatives

La classification suivant le plus proche voisin se base sur un ensemble de prototypes et chaque classification d'une nouvelle instance nécessite de parcourir cet ensemble. Le déploiement d'un tel modèle est d'autant facilité que le nombre de prototypes est faible. Eva peut être appliquée en tant que méthode de sélection d'instances. Nous menons différentes expériences afin de montrer la qualité de la sélection réalisée, comparativement à plusieurs autres méthodes de sélection. Ceci permet également d'illustrer les différences de comportement entre ces méthodes.

Nous n'incluons pas les méthodes de construction de prototypes dans notre comparatif. Cela introduirait un niveau supplémentaire de discussion, ayant trait à la différence de biais entre les méthodes : en construisant des prototypes, on s'autorise à explorer un ensemble de modèles plus riche que celui considéré par la sélection d'instances. Le lecteur intéressé peut se reporter au travail décrit dans [Ferrandiz and Boullé, 2006d], où des expériences sont menées pour comparer un ancêtre d'Eva et un algorithme de quantification.

5.2.1 Qualité de la sélection d'instances

Nous comparons Eva à plusieurs méthodes de l'état de l'art sur une vingtaine de jeux de données mis à disposition par l'UCI [Blake and Merz, 1996]. Les jeux de données utilisés ainsi que leurs caractéristiques sont répertoriés dans le tableau 5.1. Les éventuelles variables catégorielles sont purement et simplement éliminées, afin d'éviter toute interférence sur les résultats liée au choix d'un procédé de gestion de la mixité. Les données sont alors numériques et nous utilisons la métrique L_1 pour mesurer les distances entre

Jeux	Taille	Variables	Classes	Classe majoritaire
Iris	150	4	3	33%
Wine	178	13	3	40%
Sonar	208	60	2	53%
Heart	270	10	2	56%
Bupa	345	6	2	58%
Ionosphere	351	33	2	64%
Australian	690	6	2	56%
Crx	690	6	2	56%
Breast	699	9	2	66%
Pima	768	8	2	65%
Vehicle	846	18	4	26%
LED	1000	7	10	11%
Yeast	1484	8	10	31%
Segmentation	2310	19	7	14%
Abalone	4177	7	28	16%
Spam	4307	57	2	65%
Waveform	5000	21	3	34%
WaveformNoise	5000	40	3	34%
Satimage	6435	36	6	24%
Thyroid	7200	21	3	93%

TAB. 5.1 – Caractéristiques des jeux de données.

les individus. Ces choix sont dictés par le fait qu'on cherche à déterminer la qualité de la sélection, pas celle de la métrique : nous choisissons donc la métrique la plus simple possible.

Critères de comparaison. – Les critères d'évaluation employés sont :
- le nombre de prototypes,
- le taux de bonne prédiction, évalué sur un ensemble de test,
- la robustesse, égale au rapport du taux de bonne prédiction évalué en test sur le taux de bonne prédiction évalué en apprentissage,
- le temps de calcul.

Un procédé de validation croisée à dix niveaux stratifiés est appliqué. Les quantités mesurées sont donc des moyennes sur dix valeurs. Un test de Student avec un niveau de confiance de 5% est appliqué afin de déterminer si les différences de performance entre les méthodes sont significatives.

La considération du nombre de prototypes, du taux de bonne prédiction et du temps de calcul autorise une analyse multi-critère de la performance des méthodes de sélection. Nous étudions ainsi le compromis entre compression et performance prédictive que réalisent ces méthodes, tout en mesurant le temps nécessaire à l'obtention d'un résultat. La robustesse est un indicateur quantifiant la stabilité de l'estimation du taux de bonne prédiction

	Eva	Explore	DROP3	IB3	ENN	CNN	PPV
Iris	3.0	4.1	10.1	17.8	126.8	19.2	132.5
Wine	4.0	4.3	26.0	54.1	125.5	53.1	160.2
Sonar	4.3	3.9	41.6	63.8	153.0	60.7	187.2
Heart	2.5	2.3	39.1	84.9	160.5	139.7	243.0
Bupa	2.5	3.0	48.1	115.7	189.5	182.0	307.2
Ionosphere	6.2	10.6	27.6	54.5	280.4	63.7	315.1
Australian	2.4	4.0	79.7	200.4	426.0	328.9	620.2
Crx	2.5	2.7	79.1	194.0	428.8	325.3	620.2
Breast	2.9	4.4	17.5	30.1	401.3	65.8	421.1
Pima	3.4	4.2	75.0	196.1	479.2	348.4	691.2
Vehicle	13.9	29.0	137.9	312.2	519.4	379.0	761.4
LED	12.1	10.5	28.0	58.8	47.1	68.1	90.7
Yeast	6.5	61.6	186.7	708.4	693.6	881.9	1310.3
Segmentation	21.8	54.0	166.0	192.6	1821.8	221.9	1897.9
Abalone	5.8	85.3	376.2	3017.6	811.0	3315.6	3759.3
Spam	4.5	2.9	474.9	762.2	2965.1	1219.9	3560.4
Waveform	15.1	85.8	583.0	1189.6	3618.2	1809.5	4500.0
WaveformNoise	17.4	106.7	669.2	1315.3	3589.0	1962.7	4500.0
Satimage	30.1	100.7	455.7	868.5	5283.1	1144.5	5791.5
Thyroid	2.6	3.9	108.6	1331.1	6039.5	1112.9	6420.0
Moyenne	8.2	29.2	181.5	538.4	1407.9	685.1	1814.5
V/D		12/1	20/0	20/0	20/0	20/0	20/0

TAB. 5.2 – Nombre de prototypes, estimé par validation croisée à 10 niveaux stratifiés. Le nombre de Victoires/Défaites significatives d'Eva est reporté.

entre l'apprentissage et le test. Il mesure, de manière totalement empirique, la fiabilité de l'estimation de ce taux.

Méthodes comparées. – Les méthodes comparées sont les suivantes, le paramétrage étant fixé suivant les recommandations des auteurs respectifs :

- la classification par le plus proche voisin sans sélection d'instances,
- CNN,
- ENN, avec un nombre de voisins égal à 3,
- IB3, avec un niveau d'acceptation égal à 0.9 et un niveau d'élimination égal à 0.75,
- DROP3, avec un nombre de voisins égal à 3 pour le calcul des associés,
- Explore, avec un nombre de mutations égal à 1000,
- Eva, avec un niveau égal à 4.

Les méthodes ICF et PSBoost2, proches de DROP3 et aux performances similaires, ne sont pas incluses dans le comparatif.

	Eva	Explore	DROP3	IB3	ENN	CNN	PPV
Iris	94	96	92	93	93	92	95
Wine	91	81	71	76	70	85	83
Sonar	71	64	79	80	77	81	85
Heart	72	67	68	63	65	59	63
Bupa	66	62	60	58	63	58	61
Ionosphere	88	88	89	88	85	89	91
Australian	73	71	69	67	68	64	68
Crx	72	71	68	66	65	64	67
Breast	96	96	96	96	95	94	96
Pima	73	75	72	70	70	65	69
Vehicle	62	61	63	63	63	65	68
LED	64	66	57	63	64	64	73
Yeast	52	57	54	49	53	51	54
Segmentation	89	92	92	95	96	96	97
Abalone	25	26	25	21	22	21	21
Spam	86	81	81	83	82	83	85
Waveform	80	82	77	73	75	72	77
WaveformNoise	79	80	75	70	71	70	76
Satimage	87	89	88	88	89	88	91
Thyroid	93	94	93	92	89	91	93
Moyenne	75.7	75.0	73.5	72.7	72.6	72.7	75.6
V/D		4/4	7/3	12/3	10/3	11/3	8/5

TAB. 5.3 – Taux de bonne classification des méthodes comparées, estimé par validation croisée à 10 niveaux stratifiés. Le nombre de Victoires/Défaites significatives d'Eva est reporté.

Réduction du nombre d'instances. – L'inspection de la taille moyenne des ensembles de prototypes construits (*c.f.* Table 5.2) conduit à classer les méthodes en trois catégories : ENN est *conservatrice*, DROP3, IB3 et CNN sont *compressives*, Eva et Explore sont *drastiques*. Cette séparation n'est pas surprenante et résulte des parti-pris régissant leur conception. Ainsi, le but d'ENN est d'éliminer les instances mal étiquetées, ce qui conduit naturellement à éliminer peu d'instances. Les méthodes compressives reposent sur l'idée que seules les instances "utiles" sont à conserver. Chacune considère une définition particulière de l'utilité, évaluant la contribution individuelle à la performance prédictive de chacune des instances. Les plus utiles sont conservées. Le caractère individuel de l'évaluation empêche la détection d'une certaine redondance et conduit à conserver plus d'instances que nécessaire. Les méthodes sont donc compressives sans que cette compression soit drastique (entre 2 et 8 fois moins de prototypes que ENN mais entre 6 et 20 fois plus que Explore).

Les méthodes drastiques travaillent pour leur part avec un critère d'évaluation mesurant la qualité d'un *ensemble* de prototypes. Elles évaluent l'information qu'un tel en-

	Eva	Explore	DROP3	IB3	ENN	CNN	PPV
Iris	96	97	98	96	96	92	95
Wine	100	94	86	82	83	85	83
Sonar	90	83	91	89	88	81	85
Heart	97	94	90	77	82	59	63
Bupa	92	88	81	72	81	59	61
Ionosphere	95	94	98	93	91	89	91
Australian	99	98	91	80	85	64	68
Crx	97	99	89	79	81	64	67
Breast	99	99	99	99	98	95	96
Pima	98	97	90	83	85	65	69
Vehicle	93	85	84	70	79	65	68
LED	97	97	97	95	100	94	93
Yeast	97	84	84	56	77	52	54
Segmentation	98	97	97	97	98	96	97
Abalone	97	75	66	22	50	21	21
Spam	100	100	93	92	90	83	85
Waveform	97	93	91	81	84	72	77
WaveformNoise	97	91	91	78	80	70	76
Satimage	98	96	96	91	94	89	91
Thyroid	100	100	99	95	94	91	93
Moyenne	96.9	93.2	90.5	81.3	85.7	74.3	76.6
V/D		6/0	14/1	16/0	14/0	18/0	17/0

TAB. 5.4 – Robustesse (en pourcentage) des méthodes comparées et nombre de Victoires/Défaites significatives d'Eva. La robustesse est un indicateur empirique qui rapporte le taux de bonne prédiction estimé en apprentissage au taux de bonne prédiction estimé en test.

semble contient relativement à l'ensemble des instances. Si Explore s'intéresse à l'information prédictive (un ensemble de prototypes est d'autant meilleur qu'il minimise le nombre d'instances mal classées), Eva s'intéresse à la distribution de la cible. Là où un prototype porte seulement les exceptions de sa cellule dans la méthode Explore, Eva fait porter à chaque prototype la distribution des étiquettes de sa cellule. Du fait que plus d'information est prise en compte dans le deuxième cas, il n'est pas étonnant qu'un nombre plus faible de prototypes soit nécessaire : en moyenne, les ensembles de prototypes construits pas Eva sont 4 fois plus petits. Ceci n'est pas négligeable dans la mesure où chaque demande de classification nécessite de parcourir l'ensemble des prototypes ; tout gain facilite un peu plus le déploiement d'un modèle de classification par le plus proche voisin.

Performance prédictive. – En réduisant le nombre de prototypes, on peut légitimement se demander si cela conduit à une perte de performance prédictive. Au regard de la Table 5.3, les méthodes se répartissent en deux groupes. Les méthodes DROP3, IB3,

	Eva	Explore	DROP3	IB3	ENN	CNN	PPV
Iris	1	1	0	0	0	0	0
Wine	1	2	0	0	0	0	0
Sonar	2	7	1	0	1	1	0
Heart	1	2	0	0	0	0	0
Bupa	2	2	0	0	0	0	0
Ionosphere	3	23	2	0	1	1	0
Australian	6	6	1	0	1	1	0
Crx	5	5	2	0	1	1	0
Breast	2	7	1	0	1	0	0
Pima	6	10	2	1	1	2	0
Vehicle	11	79	4	2	3	4	0
LED	1	1	0	0	0	0	0
Yeast	18	234	4	3	3	6	0
Segmentation	55	538	49	3	20	6	1
Abalone	107	1319	17	34	14	44	1
Spam	242	562	251	45	160	151	2
Waveform	240	5819	153	38	109	129	2
WaveformNoise	372	13980	261	86	217	269	3
Satimage	528	19209	738	63	333	216	3
Thyroid	435	373	582	59	231	126	1
Moyenne	102	2109	104	17	55	48	1
V/D		16/0	2/16	0/20	0/20	0/20	0/20

TAB. 5.5 – Temps de calcul moyen (en secondes) des méthodes comparées et nombre de Victoires/Défaites significatives de Eva.

ENN et CNN définissent chacune à leur manière une notion individuelle d'utilité prédictive d'une instance. Cette définition repose sur une intuition à laquelle certains jeux de données ne correspondent pas. En conséquence, les algorithmes DROP3, IB3, ENN et CNN perdent respectivement 2.1%, 2.9%, 3% et 2.9% de performance prédictive par rapport à la règle PPV.

Une nouvelle fois, le fait de travailler avec un critère d'évaluation globale permet aux méthodes Eva et Explore de fournir une sélection de prototype préservant la performance prédictive : 75.7% et 75% de bonnes prédictions pour Eva et Explore respectivement contre 75.6% pour la règle PPV. Notons qu'Explore a pour objectif d'obtenir un bon taux de bonne classification et qu'Eva a pour objectif de détecter des différences d'homogénéité dans la distribution des étiquettes. Le second objectif inclut le premier (si on connaît la distribution des étiquettes en un point, on connaît l'étiquette majoritaire).

Fiabilité. – Le nombre de prototypes sélectionnés est lié à la robustesse du modèle : plus une méthode conserve d'instances, plus son résultat est dépendant des données d'apprentissage et plus la performance du modèle est susceptible de s'écrouler sur de nouvelles

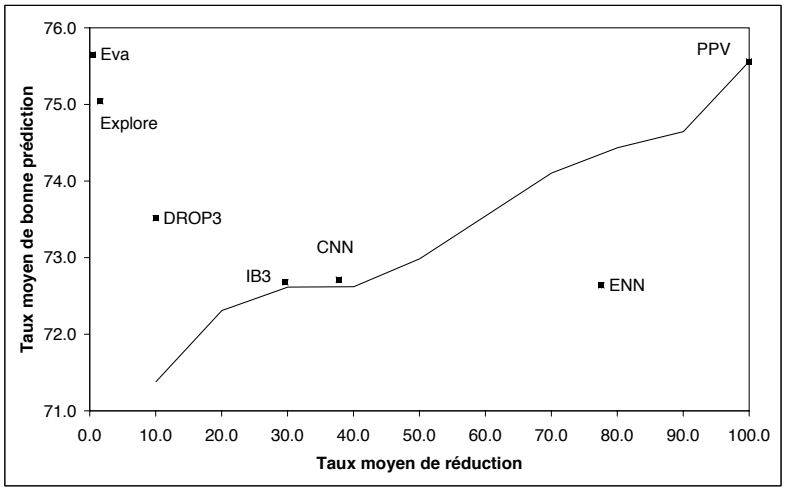

FIG. 5.3 – Evaluation bi-critère des méthodes comparées, selon leur pourcentage moyen de réduction et leur pourcentage moyen de bonne classification. Les performances de la sélection aléatoire uniforme d'instances est reportée pour différents pourcentages de réduction (de 10% à 100%).

données. Inversement, plus le nombre de prototypes est faible, plus le nombre d'instances qu'il représente croît. Dans ce sens, un prototype porte alors plus d'information. Sous réserve que cette information soit prise en compte par le critère d'évaluation, la classification d'une nouvelle instance est d'autant plus fiable.

Il n'est donc pas étonnant de constater (*c.f.* Table 5.4) qu'Eva produit les modèles les plus robustes (96.9% en moyenne). En effet, chaque prototype porte la distribution des étiquettes des instances de sa cellule de Voronoi et le critère tient compte de cette information. La robustesse est améliorée de 3.7% par rapport à Explore, qui tient seulement compte de l'étiquette majoritaire dans chaque cellule. Par rapport aux méthodes adoptant une évaluation individuelle, la robustesse est améliorée de 6% à 20%. Le résultat fourni par Eva est donc le plus fiable.

En situation réelle, le jeu de données est découpé en trois parties : l'ensemble d'apprentissage permet d'estimer les modèles, l'ensemble de validation sert à contrôler le risque de sur-apprentissage lors de l'ajustement des paramètres, et l'ensemble de test est utilisé pour évaluer indépendamment la performance prédictive du modèle final. Le critère utilisé par Eva prend automatiquement en charge la gestion du sur-apprentissage et rend inutile l'ensemble de validation. L'analyste peut alors utiliser plus de données en apprentissage. Ceci profite à la qualité du résultat.

Temps de calcul. – La Table 5.5 des temps de calcul complète l'analyse. Sur les jeux de données de petite taille (*i.e.* inférieure à 1000 instances), les différences sont statisti-

quement significatives et en défaveur de Eva (sauf vis-à-vis d'Explore). Les temps étant très faibles, ça n'a pas grande importance en pratique, d'autant plus qu'Eva peut être lancée avec un niveau inférieur à 4. Sur de plus gros jeux de données, des différences importantes commencent à apparaître. On voit ainsi qu'Explore est un algorithme beaucoup plus coûteux en temps que les autres. C'est naturel, car une évaluation globale demande plus d'effort qu'une évaluation individuelle. Bien que procédant également à une évaluation globale, les optimisations que nous avons proposées conduisent Eva à se comporter comme DROP3, qui lui se base sur une évaluation individuelle.

Nous concluons comme suit : les méthodes classiques (DROP3, IB3, ENN, CNN) construisent rapidement un gros ensemble de prototypes peu fiable et peu performant et les méthodes globales (Eva, Explore) construisent un petit ensemble de prototypes fiable et performant. Ceci est illustré par la fig.5.3, les méthodes étant soumises à une analyse bi-critère (pourcentage moyen de réduction et pourcentage moyen de bonne classification en test). Nous incluons dans cette analyse les méthodes de sélection aléatoire d'un ensemble de prototypes de taille fixée, pour référence. Notre méthode Eva se distingue d'Explore en étant encore plus rapide, plus pertinente et plus fiable.

5.2.2 Qualité de l'heuristique d'optimisation

Dans cette section, nous comparons l'algorithme RVVGlouton et l'algorithme de la méthode Explore. Nous utilisons pour cela des données synthétiques. Le critère optimisé est le critère MDL utilisé par Eva. Notons que l'expérience ici décrite a également été conduite sur une vingtaine de jeux de données de l'UCI, aboutissant aux mêmes conclusions.

Données synthétiques. – Nous considérons un problème à deux classes et générons 2500 instances uniformément sur un échiquier de taille 4 (*c.f.* fig.5.4). Une version corrompue est obtenue en inversant uniformément l'étiquette de 20% des instances.

Expérience. – Plusieurs optimisations sont lancées avec différentes valeurs de niveau pour RVVGlouton et différents nombres de mutations pour Explore. Pour chaque valeur de paramètre, les indicateurs mesurés sont le temps de calcul, la valeur du critère et le nombre de prototypes de la solution. Une validation croisée stratifiée à cinq niveaux est appliquée. Nous obtenons ainsi une courbe pour chaque algorithme et pour chaque indicateur, fonction du temps de calcul. Pour le jeu de données non perturbé, les courbes sont reportées sur la fig.5.5, et les courbes relatives au jeu de données perturbé sur la fig.5.6. A titre de référence, nous évaluons l'ensemble de 16 prototypes constitué par les instances les plus proches des centres de chacune des cases, dans la mesure où il constitue certainement une bonne solution.

Résultats dans le cas pur. – Dans le cas non corrompu, l'algorithme RVVGlouton propose une première solution en 1 seconde (RVVGlouton(0) effectue une unique passe

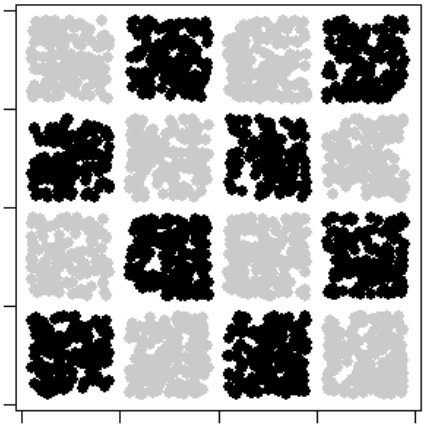

FIG. 5.4 – 2500 instances sur l'échiquier de taille 4.

gloutonne), évaluée à 362 nats. L'algorithme Explore nécessite déjà 16 secondes (Explore(0) effectue une passe incrémentale et une passe décrémentale, sans mutations). La différence s'accentue rapidement, la méta-heuristique employée par Eva conduisant rapidement à de meilleures solutions : en 22 secondes, Eva(4) propose une solution évaluée à 217 nats, alors que 2000 mutations n'ont pas permis à Explore d'améliorer significativement la solution première (la solution obtenue par Explore(2000) étant évaluée à 341 nats).

Intuitivement, la meilleure chose à faire ici est de constituer un minimum de groupes purs à l'aide d'une partition de Voronoi. C'est ce que fait la solution de référence composée de 16 prototypes. C'est donc un optimum global, évalué à 172 nats. L'algorithme RVVGlouton, pour un niveau égal à 13, aboutit à une telle solution. A temps de calcul équivalent, l'algorithme Explore construit plus de 18 groupes. Si on peut supposer que l'heuristique Explore finira par trouver une solution optimale, cette expérience montre que l'algorithme RVVGlouton construit toujours de meilleures solutions qu'Explore à temps de calcul donné ou, en d'autres termes, une solution équivalente plus rapidement.

Résultats dans le cas bruité. – Dans le cas corrompu, on ne peut garantir que la solution référence (évaluée à 1088 nats) est un optimum global. En effet, l'introduction d'un bruit conduit aléatoirement à l'obtention de formes. La présence de bruit d'étiquetage réduit également les différences relatives entre les méthodes. Cependant, l'heuristique RVVGlouton continue à fournir de meilleures solutions.

Cette expérience permet de comparer les stratégies employées pour sortir d'un optimum local. L'algorithme RVVGlouton(0) trouve une première solution évaluée à 1157 nats alors qu'Explore(0) tombe déjà dans un assez mauvais optimum local, puisque celui-ci est évalué à 1287 nats et se compose de 13 prototypes seulement. Là où l'adoption d'une méta-heuristique de recherche à voisinage variable conduit RVVGlouton à se sortir aisément d'un optimum local, la stratégie employée par Explore est plus limitée : en pratiquant des ajouts, des suppressions, des échanges un par un, 100 secondes et 16000 mutations

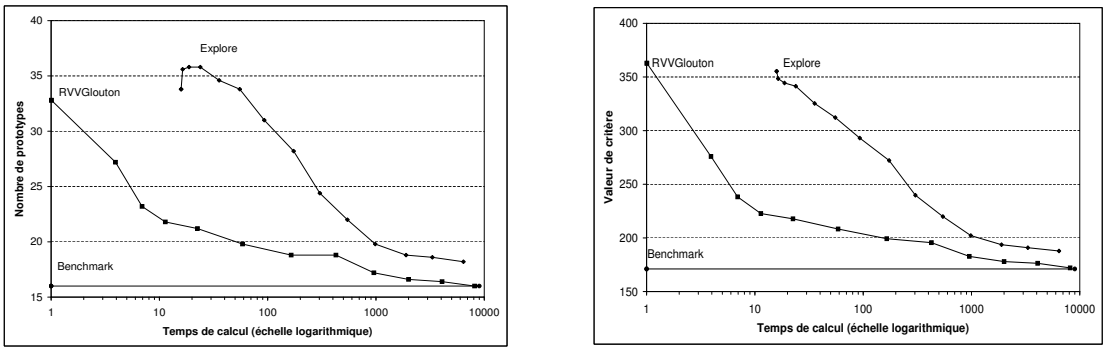

FIG. 5.5 – Comparaison des heuristiques RVVGlouton et Explore à critère fixé, sur les données "échiquier" non corrompues. La valeur du critère et le nombre de prototypes sont reportés en ordonnée et fonction du temps de calcul.

sont nécessaires pour trouver une "bonne" solution. Au final, on constate qu'Explore est incapable asymptotiquement de sortir d'un optimum local à 18 prototypes.

Cette expérience illustrative sur un jeu de données synthétiques permet de montrer les apports de l'algorithme RVVGlouton. L'heuristique gloutonne optimisée construit de bonnes solutions dans un temps court. L'emploi de la méta-heuristique à voisinage variable permet une application itérative efficace de l'heuristique gloutonne, facilitant la vie de l'algorithme lorsqu'il s'agit d'améliorer une solution. On voit de plus que le seul paramètre de l'algorithme (le niveau) quantifie le temps d'optimisation alloué, comme souhaité. Cela signifie que, du point de vue de l'utilisateur, il suffit de spécifier la quantité de temps allouée à la tâche : la qualité de l'algorithme assure que ce temps est bien exploité.

5.2.3 Illustration des différences entre les méthodes

Pour terminer, nous visualisons les différences de comportement entre les méthodes de sélection d'instance par l'intermédiaire d'une expérience sur des données synthétiques. Par extension, nous discutons les intuitions à l'origine de la définition de chacune de ces méthodes.

Première expérience

Description de l'expérience. – Nous générons uniformément 800 points dans un carré et considérons un problème à deux classes : la distribution des étiquettes est $(0.8, 0.2)$ dans les coins supérieur droit et inférieur gauche du carré, et $(0, 1)$ dans les coins restant. Les algorithmes ENN, IB3, DROP3, Explore et Eva sont appliqués avec les mêmes paramétrages que pour l'expérience sur les jeux de données de l'UCI. La règle CNN n'est pas

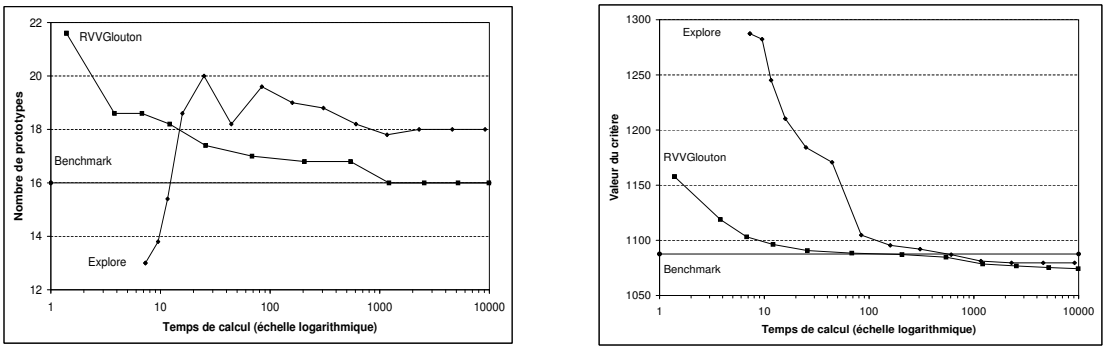

FIG. 5.6 – Comparaison des heuristiques RVVGlouton et Explore à critère fixé, sur les données "échiquier" corrompues. La valeur du critère et le nombre de prototypes sont reportés en ordonnée et fonction du temps de calcul.

considérée dans la mesure où IB3 en est un raffinement. Le jeu de données ainsi que les ensembles de prototypes sélectionnés par ces algorithmes sont reportés sur la fig.5.7.

Comportement de la règle ENN. – La règle ENN fonctionne en pratique comme un filtre : les instances mal classifiées par un vote sur leur K plus proches voisins sont considérées comme bruitées et éliminées. Le paramètre K fonctionne comme un seuil de décision au delà duquel un groupe d'instances de même classe est considéré comme significatif. Dès que K instances d'une même classe sont suffisamment proches, le groupe est conservé (ici $K = 3$). Dans ce cas, le paramètre K quantifie le niveau de sur-apprentissage qu'on s'autorise. L'ajustement de ce paramètre constitue donc une tâche sensible, d'autant plus que la méthode augmente l'importance des groupes déclarés significatifs. En effet, les instances d'une autre classe proches de ce groupe sont considérées comme du bruit et éliminées, ce qui augmente la zone d'influence du groupe.

Comportement d'IB3. – L'algorithme IB3 étant un raffinement de la règle CNN, il partage les mêmes propriétés tout en sélectionnant moins d'instances. Sur l'exemple, les instances centrales des zones pures sont éliminées : plus celles-ci sont représentées, plus la compression est élevée. L'algorithme étant incrémental stochastique, un certain nombre de ces instances est tout de même conservé. Un inconvénient de cette méthode est que toute instance mal classifiée tend à être conservée également. Cela limite d'une part la compression. D'autre part, pour les mêmes raisons que l'algorithme ENN, leur importance est surestimée au détriment de la performance prédictive.

Comportement de DROP3. – Comme pour ENN, le paramètre K de la méthode DROP3 (ici $K = 3$) contrôle la définition de la significativité d'un groupe d'instances, bien que cette définition diffère. Le challenge que constitue le réglage de ce paramètre est

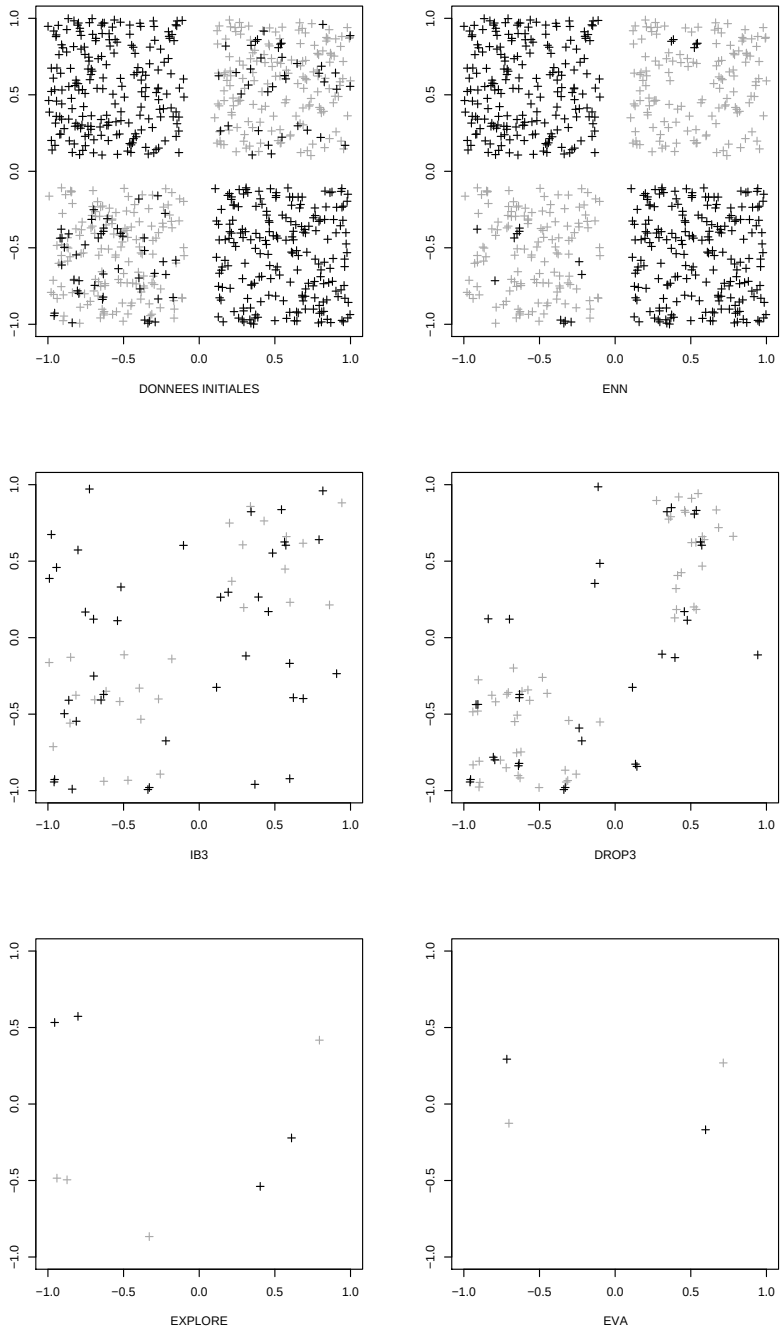

FIG. 5.7 – Illustration des différences de comportement des méthodes ENN, IB3, DROP3, Explore et Eva sur un jeu de données synthétiques. Les instances dans les coins supérieur gauche et inférieur droit sont toutes de même classe. Les instances dans les autres coins sont réparties dans les classes à raison de 20% dans la classe grise et 80% dans la classe noire.

donc le même que pour ENN. Au contraire d'IB3 et ENN, DROP3 est moins affecté par la présence de données bruitées. En effet, dès qu'un groupe significatif est détecté, toute instance proche appartenant à une autre classe tend à être conservée. Ainsi, l'algorithme tend à conserver les instances frontalières d'une zone pure. Cet aspect est renforcé par le fait que l'optimisation est décrémentale et que les suppressions sont testées par distance décroissante à la plus proche instance d'une autre classe.

Comportement d'Explore. – Le critère utilisé par Explore quantifie le compromis entre le nombre de mauvaises classifications et le nombre de prototypes. L'évaluation est globale et pénalisée. L'objectif est de préserver la qualité prédictive avec le moins de prototypes possibles. L'apport décisif du critère est son caractère non paramétrique, ce qui signifie que le seuil de significativité est fixé automatiquement.

Sur le jeu de données considéré, Explore considère donc les petits groupes situés en zone ennemie comme du bruit. En effet, isoler ces zones nécessite la considération d'un nombre trop élevé de prototypes. En raison d'un certain manque d'efficacité de l'heuristique d'optimisation, on peut noter que l'ensemble de prototypes constitué est un peu plus gros que nécessaire.

Comportement d'Eva. – Notre méthode vise à détecter des différences significatives dans la distribution conditionnelle de la variable cible. L'algorithme d'optimisation du critère étant efficace, le problème ici posé est résolu exactement : quatre zones sont détectées.

Seconde expérience

Nous discutons maintenant les limites des méthodes classiques.

Description de l'expérience. – Nous générons uniformément 2000 points dans un carré et considérons deux classes, dont la distribution conditionnelle est $(0.9, 0.1)$ dans les coins supérieur droit et inférieur gauche, $(0.6, 0.4)$ dans les autres coins. La classe majoritaire est la même en tout point du carré. Les méthodes ENN, IB3, DROP3, Explore et Eva sont appliquées, avec les mêmes paramétrages que pour l'expérience sur les jeux de données de l'UCI. Le jeu de données et les ensembles de prototypes obtenus sont reportés sur la fig.5.8.

Résultats. – Cette expérience met une nouvelle fois en lumière la séparation entre les méthodes basées sur une définition individuelle de l'utilité prédictive (CNN, ENN, IB3, DROP3 et un grand nombre d'autres méthodes non discutées ici), les méthodes basées sur une définition globale de cette notion (Explore) et les méthodes basées sur une évaluation globale de la densité conditionnelle de la variable cible (Eva). Les méthodes du premier groupe définissent chacune à leur manière la notion d'intérêt prédictif individuel d'une instance. Ces définitions sont par nature dépendantes d'un contexte (et généralement paramétriques). Lorsque le problème étudié n'est pas dans le contexte adéquat, ce qui est le cas ici, les méthodes exhibent un comportement loin d'être satisfaisant.

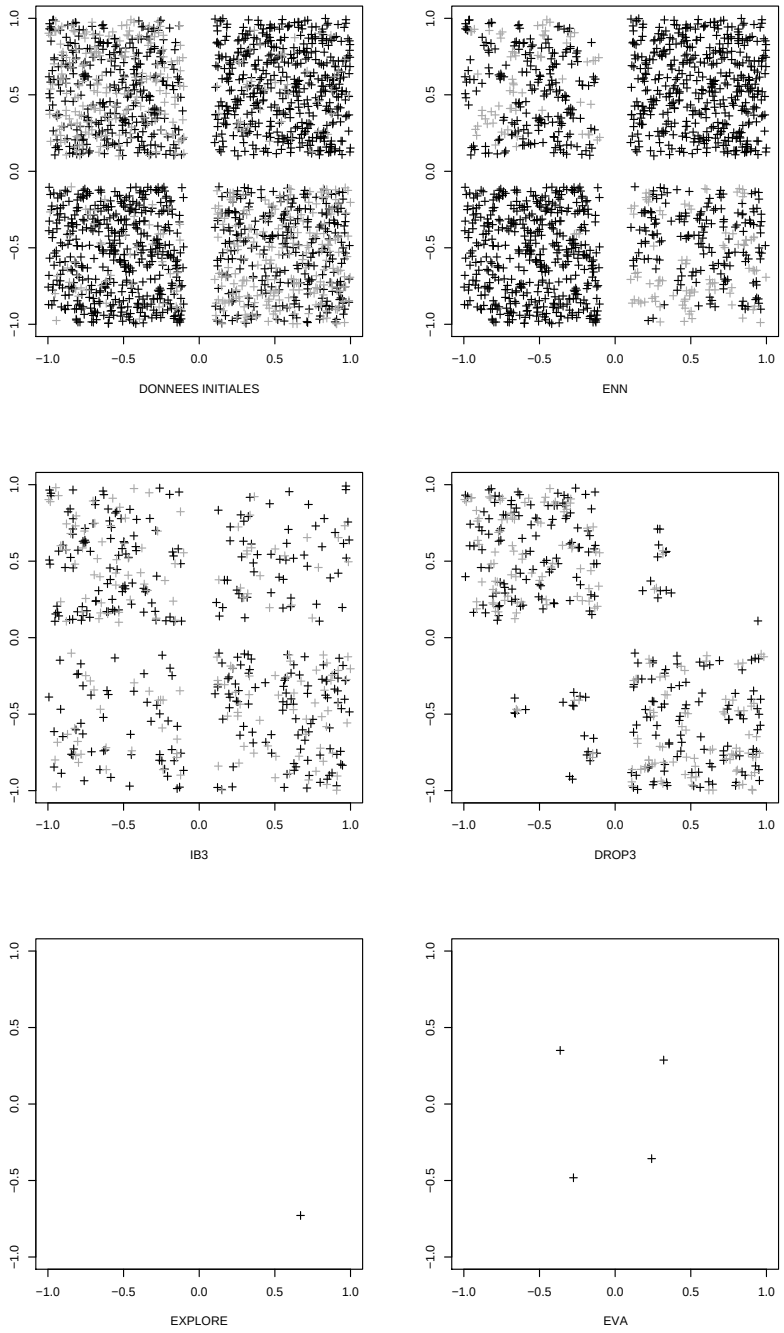

FIG. 5.8 – Illustration des limites des méthodes ENN, IB3, DROP3 et Explore comparativement à Eva sur un jeu de données synthétiques. La classe majoritaire en chaque point est la classe noire. Sa probabilité d'apparition est de 90% dans les coins supérieur droit et inférieur gauche, de 60% dans les autres coins.

En évaluant la qualité prédictive d'un ensemble de prototypes, et en pénalisant les ensembles de cardinal élevé en accord avec un principe fondé, la méthode Explore se comporte de manière satisfaisante. Ici, en terme de performance prédictive (*i.e.* en focalisant sur l'aspect majoritaire), il n'y a rien de mieux à faire que de construire un unique groupe. C'est ce que fait Explore. Mais, dans la mesure où la variable cible présente une différence significative de densité conditionnelle, il peut être souhaitable de détecter cette inhomogénéité. C'est exactement l'objectif d'Eva et elle l'atteint en constituant 4 groupes.

Pour conclure, la hiérarchie dégagée au cours des expériences est résumée à la fig.5.9.

- Niveau 1 : un prototype porte son étiquette (CNN, ENN, IB3, DROP3) et l'évaluation est individuelle,
- Niveau 2 : un prototype tient compte de la classe majoritaire dans sa cellule de Voronoi (Explore) et l'évaluation est globale,
- Niveau 3 : un prototype porte la distribution des étiquettes dans sa cellule (Eva) et l'évaluation est globale.

FIG. 5.9 – Différents niveaux de prise en compte de l'information

5.3 Conclusion

Nous avons procédé dans ce chapitre à une évaluation expérimentale de notre méthode, sous l'angle de la sélection d'instances pour la classification suivant le plus proche voisin. Ces expériences nous ont permis d'illustrer les qualités d'Eva.

Le critère d'évaluation tient compte de la distribution des étiquettes dans chaque cellule. Il est plus fin qu'un critère basé sur la performance prédictive, qui focalise sur la classe majoritaire. Le critère régularise les hypothèses plus complexes et fournit un résultat fiable. Nous avons montré qu'Eva fournit le résultat le plus fiable.

Le critère est non paramétrique. Sans paramètre à ajuster, l'utilisation d'un ensemble de validation est rendue caduque. Un plus grand nombre d'instances est alors disponible pour l'apprentissage, ce qui augmente la pertinence et la fiabilité du résultat.

L'algorithme est efficace. L'heuristique gloutonne optimisée produit rapidement une bonne solution. Chaque remise en cause opérée par la méta-heuristique améliore significativement la solution et permet d'échapper aux mauvais optimaux locaux. L'utilisateur fixe le temps alloué à la recherche et la qualité de l'algorithme assure que ce temps est bien exploité : le résultat obtenu est le meilleur possible compte tenu du délai accordé.

Conclusion

Dans cette seconde partie, nous avons décrit Eva : une méthode d'évaluation de l'intérêt d'une mesure de similitude relativement à une variable cible catégorielle. Celle-ci prend en entrée une matrice de Gram et une variable cible catégorielle, pour fournir en sortie un gain de compression et une partition de l'ensemble des individus.

Pour la développer, nous avons tout d'abord défini un ensemble d'hypothèses, chacune qualifiant à sa manière l'intérêt d'une mesure de similitude. Ces hypothèses sont les partitions de Voronoi induites par l'ensemble des individus et la matrice de Gram. Chaque partition permet de répartir les individus dans des groupes et une distribution fréquentielle des étiquettes est associée à chaque groupe. Eva prend ainsi en compte plus d'information qu'une méthode considérant uniquement la classe majoritaire dans chaque groupe.

En adaptant l'approche MDL, nous avons proposé un critère d'évaluation pénalisant les hypothèses complexes. C'est ce qui assure la fiabilité du résultat. Ce critère tient compte de l'information apportée par les distributions dans chaque groupe, ce qui rend l'évaluation plus fine qu'une évaluation basée sur la performance prédictive. Le critère est non paramétrique et évite ainsi l'utilisation d'un ensemble de validation. Plus d'individus sont disponibles pour l'apprentissage, ce qui profite à la qualité du résultat de l'évaluation.

L'heuristique d'optimisation proposée est efficace, l'utilisateur n'ayant à spécifier que le temps alloué à l'optimisation. La qualité de l'algorithme assure que le délai imparti est bien exploité.

De plus, aucune hypothèse sur la forme des données n'est nécessaire au bon fonctionnement d'Eva. Le résultat de l'évaluation dépend uniquement de la mesure de similitude et de la variable cible : l'information est extraite des données et uniquement des données. Nous avons montré que cette information est fine et fiable.

A posteriori, la méthode possède différents modes d'utilisation. Eva peut être utilisée comme une méthode de sélection d'instances pour la classification par le plus proche voisin. Nous avons montré qu'elle est très performante dans cette tâche : la sélection est drastique et la performance prédictive est préservée. Les modèles à base de plus proche voisin sont alors d'autant plus pertinents et faciles à déployer. Eva produit également une estimation de la distribution des classes cibles conditionnellement à chaque individu. Elle est ainsi utile à toute méthode utilisant une telle estimation, comme le classifieur bayésien naïf.

Troisième partie

Application : préparation de données séquentielles

Introduction

Cette troisième partie marque le retour aux données séquentielles. Nous illustrons les apports de notre méthode d'évaluation Eva à la préparation de telles données. Nous proposons également des extensions afin d'automatiser la recherche d'une représentation d'une variable séquentielle.

Dans la première partie, nous avons soulevé le besoin d'une méthode d'évaluation supervisée automatique d'une variable séquentielle. Nous avons montré que les données séquentielles sont sujettes à un travail de normalisation susceptible de les projeter dans des espaces de représentation très différents. C'est l'emploi d'une mesure de similitude qui permet d'être indépendant de la représentation. Dans la seconde partie, nous avons proposé, testé et validé une méthode d'évaluation supervisée d'une mesure de similitude : Eva.

Nous commençons dans cette partie par considérer un jeu de données relatif à un problème de classification de profils de consommation téléphonique. Dans le chapitre 6, nous menons différentes expériences à partir de ce jeu de données. La première montre les apports d'Eva à l'évaluation d'une variable séquentielle. La seconde emploie Eva dans une approche filtre de la sélection de variables séquentielles. Les dernières expériences montrent l'intérêt des autres modes d'utilisation d'Eva (en tant que règle de classification et en tant qu'estimateur de densité conditionnelle).

A titre de perspectives avancées, nous proposons au chapitre 7 deux solutions relatives à deux problèmes de représentation d'une variable séquentielle : la sélection d'un ensemble de temps de mesure et la sélection d'un fenêtrage, toujours dans le cas de la classification supervisée. Ceci permet de montrer la souplesse de notre approche de l'évaluation tout en poursuivant le but d'automatiser un peu plus la préparation des variables séquentielles.

6

Préparation de profils de consommation téléphonique

Sommaire

Nous montrons dans ce chapitre l'apport de notre méthode d'évaluation Eva à la préparation de données séquentielles. Pour cela, nous menons plusieurs expériences à partir de données de consommation téléphonique appartenant à France Télécom. Notons qu'il ne s'agit pas ici d'étudier ces données mais de les exploiter pour illustrer l'utilisation et l'intérêt de la méthode.

Eva effectuant plus qu'une évaluation, puisqu'elle partitionne également l'ensemble des individus, nous montrons dans la section 6.1 comment mettre en forme cette information supplémentaire et ainsi obtenir un support explicatif à l'évaluation. Dans la section 6.2, nous appliquons notre méthode à la sélection de variables séquentielles et de métriques. Enfin, la section 6.3 est l'occasion de montrer l'intérêt de notre méthode en tant que règle de classification puis en tant qu'estimateur de densité conditionnelle.

Description des données. – Les données exploitées dans ce chapitre sont relatives à un problème de classification de profils de consommation en téléphonie fixe. C'est un problème à 4 classes cibles A, B, C et D. Les 3516 individus sont équirépartis dans ces 4 classes. Nous disposons de 168 variables descriptives numériques, chacune mesurant la

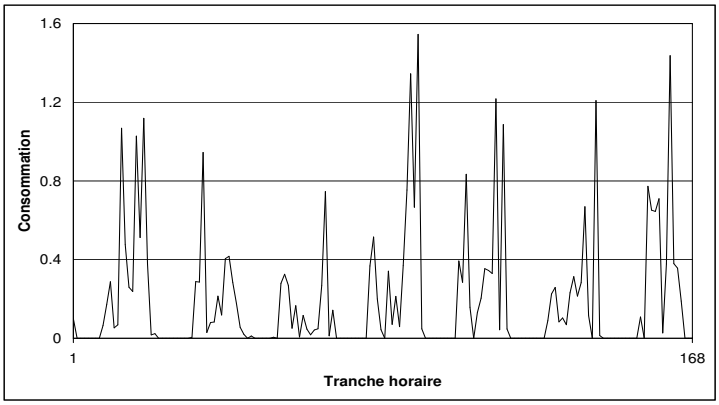

FIG. 6.1 – Exemple de profil hebdomadaire de consommation téléphonique moyenne.

consommation téléphonique sur une tranche horaire de la semaine, moyennée sur 6 mois. Nous séparons uniformément et de manière stratifiée (*i.e.* en respectant la distribution a priori des classes cibles) l'échantillon en un ensemble d'apprentissage (50% des individus), un ensemble de validation (25% des individus) et un ensemble de test (25% des individus). Nous donnons un exemple de profil de consommation sur la fig.6.1.

Les expériences ici menées ont fait l'objet de publications, dans les articles [Ferrandiz and Boullé, 2006e] et [Ferrandiz and Boullé, 2006b].

6.1 Evaluation d'une variable séquentielle

Notre méthode évalue la pertinence d'une mesure de similitude vis-à-vis d'une variable cible catégorielle. Le gain de compression quantifie cette pertinence. Il s'accompagne d'une partition de l'ensemble des individus, chaque groupe se voyant associée la distribution des fréquences des étiquettes des individus le constituant. Cette information peut être mise en forme afin de fournir un support explicatif à l'évaluation.

6.1.1 Apport explicatif d'Eva

Dans la première expérience, nous considérons la variable séquentielle "profil hebdomadaire de consommation téléphonique" composée de 168 mesures. De manière naïve, nous employons la métrique L_1 pour mesurer la distance entre deux profils. La matrice de Gram obtenue est soumise à Eva.

Résultats. – L'évaluation fournit un gain de compression de 0.052, ce qui est très faible et caractérise un fort mélange des classes cibles. Cette indicateur numérique s'accompagne d'une partition de l'ensemble d'apprentissage. La partition est constituée de 6 groupes.

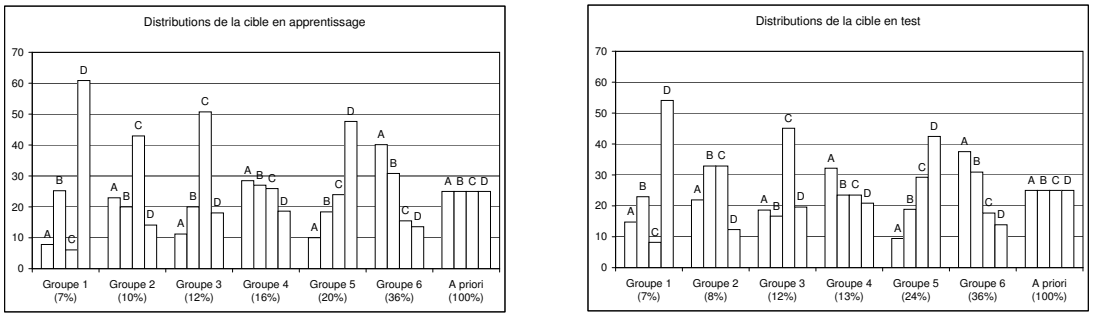

FIG. 6.2 – Distributions des étiquettes dans chaque groupe, en apprentissage et en test. Le support de chaque groupe est reporté en abscisse. Par exemple, le groupe 5 contient 20% des individus de l'ensemble d'apprentissage, 24% des individus de l'ensemble de test, et ses éléments se répartissent dans les classes cibles suivant la distribution $(10\%, 18\%, 24\%, 48\%)$ en apprentissage et $(9\%, 19\%, 29\%, 42\%)$ en test.

Les distributions relatives à chacun des groupes sont représentées par des histogrammes groupés sur la fig.6.2. Nous reportons également les distributions obtenues en test. En calculant dans chaque groupe la valeur moyenne de chacune des 168 variables, on obtient un profil de consommation moyen caractéristique de chaque groupe. Trois de ces profils sont reportés sur la fig.6.3, ainsi que le profil moyen de consommation (*i.e.* celui calculé sur tout l'ensemble d'apprentissage).

Interprétation. – En mettant en correspondance les fig.6.2 et fig.6.3, le résultat est interprétable par l'utilisateur. Par exemple, on voit que les individus du groupe 6 sont en grand nombre (36% des instances), que leur consommation moyenne est plus élevée que la moyenne globale, et que ce comportement est majoritairement caractéristique de la classe A (la répartition dans les classes cibles A, B, C, D est $(40\%, 31\%, 15\%, 14\%)$). Le groupe 1 est quant à lui plus discriminant en faveur de la classe D (la répartition dans les classes cibles est $(8\%, 25\%, 6\%, 61\%)$) avec un profil de consommation atypique (consommation élevée uniquement le soir), mais est de taille réduite (7% des instances). Le groupe 5 discrimine lui aussi la classe D, moins fortement tout de même que le groupe 1, et caractérise les consommations très faibles.

Le calcul des distributions en test permet de vérifier la fiabilité du découpage effectué. Bien que l'ensemble de test soit ici deux fois plus petit que l'ensemble d'apprentissage, nous constatons que la distribution des individus dans les groupes est stable ($(7\%, 10\%, 12\%, 16\%, 20\%, 36\%)$ en apprentissage et $(7\%, 8\%, 12\%, 13\%, 24\%, 38\%)$ en test), que les classes majoritaires dans chaque groupe en test sont les mêmes que celles observées en apprentissage.

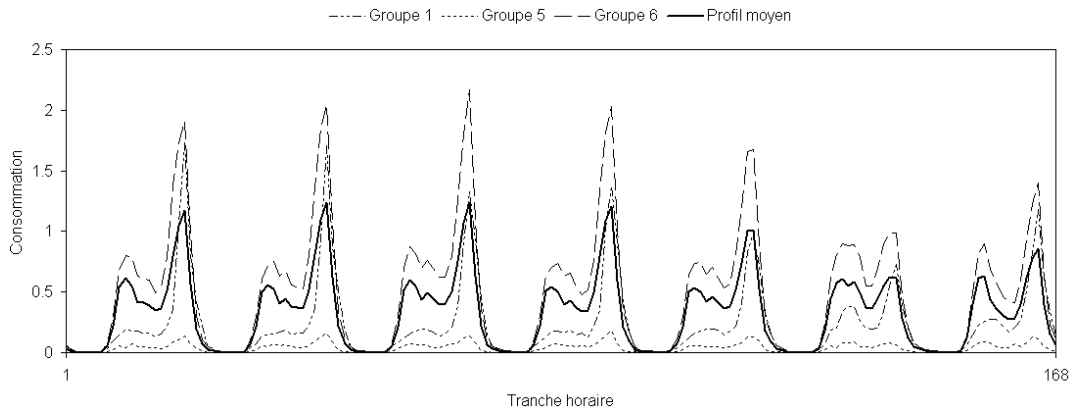

FIG. 6.3 – Consommation moyenne sur l'ensemble d'apprentissage et consommations moyennes dans chacun des groupes 1, 5 et 6. Les individus du groupe 1 ont une consommation piquée le soir et uniquement le soir. Ceux du groupe 6 ont une très faible consommation.

Le gain de compression mesure à quel point une partition est capable de faire apparaître des différences d'homogénéité dans les classes cibles. Son optimisation conduit à l'obtention d'une structure capturant un maximum de telles différences. Cette structure constitue un support explicatif pour qui veut analyser et comprendre la décision prise par Eva.

6.1.2 Discussion des alternatives

Notre méthode n'est pas la première à fournir les informations que l'on a visualisées dans la section précédente. Ainsi, l'analyse discriminante, linéaire ou quadratique, ou toute méthode supervisée construisant des prototypes, comme les méthodes de quantification, conduisent à de telles visualisations.

Analyse discriminante [Hastie *et al.*, 2001]. – L'analyse discriminante suppose les individus d'une classe tous générés par une gaussienne. Les gaussiennes sont positionnées par maximisation de la vraisemblance. Ceci se traduit en pratique par un nombre de groupes égal au nombre de classes, et les modèles considérés sont tous de même complexité. L'exemple étudié ici montre que contraindre la capacité revient à limiter la richesse de l'information extraite, voire à la biaiser (le monde n'est pas toujours gaussien).

De toute façon, même si l'on s'autorise un nombre quelconque de gaussiennes dans chaque classe (*i.e.* un mélange gaussien), la vraisemblance ne tient pas compte des différences de complexité alors introduites. Il est nécessaire de régulariser les modèles en adoptant un a priori sur l'ensemble des densités, ou d'utiliser l'échantillon de validation, ou d'appliquer un critère complémentaire. La visualisation proposée ne tient pas compte de tous ces choix et n'en est que plus inadaptée.

Quantification [Kohonen, 2001]. – Les techniques de quantification, hautement paramétriques, nécessitent entre autres de fixer a priori le nombre de prototypes. Le choix

d'un "bon" nombre de prototypes repose alors sur un critère alternatif. En pratique, on utilise le risque empirique et un ensemble de validation.

L'idée sous-jacente à la technique est de repousser les prototypes en cas de mauvais étiquetage et de les rapprocher dans le cas contraire. De nombreux prototypes sont déplacés au point de ne plus être sollicités par la suite. La présence de prototypes "morts" à la fin de l'optimisation constitue un effet secondaire peu désirable. Le résultat perd de sa pertinence et la visualisation associée est une nouvelle fois inadaptée.

Apports de notre méthode. – Le critère MDL étant non paramétrique et prenant en charge la gestion du sur-apprentissage, les données réservées pour la validation sont utilisables par notre méthode. L'augmentation du nombre de données disponibles pour l'apprentissage profite à la qualité de la partition construite. Dans le cas présent, cela permet de distinguer 7 comportements au lieu des 6 précédemment détectés.

De plus, la visualisation est adaptée à l'analyse du résultat fournit par Eva. L'information contenue dans la visualisation est exactement celle prise en compte par notre critère d'évaluation. La visualisation et l'interprétation qu'elle autorise sont donc aussi pertinentes et fiables que peut l'être Eva.

6.2 Evaluation pour sélection

Pour un problème de classification supervisé, *i.e.* pour une variable cible catégorielle donnée, notre méthode associe un gain de compression à une matrice de Gram. A travers le prisme de cette matrice, c'est la variable séquentielle qui est évaluée. Le gain de compression permet donc de comparer différentes mesures de similitude et différentes variables séquentielles.

6.2.1 Sélection de variables séquentielles

Plaçons-nous dans la situation où l'on dispose de plusieurs variables séquentielles. Le gain de compression mesuré par Eva pour chacune d'elles fournit un indicateur de comparaison, une fois fixée une mesure de similitude. Nous illustrons cette utilisation à partir du problème de classification de profils de consommation.

Expérience. – Nous considérons les 24 variables séquentielles définies par les tranches horaires, chacune composée de 7 mesures. Nous appliquons notre méthode d'évaluation (au $Niveau\ 0$) à chaque variable, en employant la métrique L_1, sur l'ensemble d'apprentissage. Le gain de compression (en apprentissage) et le taux de bonne classification en test sont mesurés. Les 24 variables étant ordonnées naturellement, nous obtenons des courbes de gain de compression et de taux de bonne classification. Elles sont reportées sur la fig.6.4.

Résultats. – Utilisé pour de la sélection, le gain de compression conduit à choisir la tranche horaire 14h-15h. A l'opposé, le gain de compression est nul pour les tranches horaires entre 1h et 8h du matin. Ceci signifie qu'un seul groupe est constitué et qu'aucune différence d'homogénéïté n'est détectée. Considérées isolément, ces variables ne sont

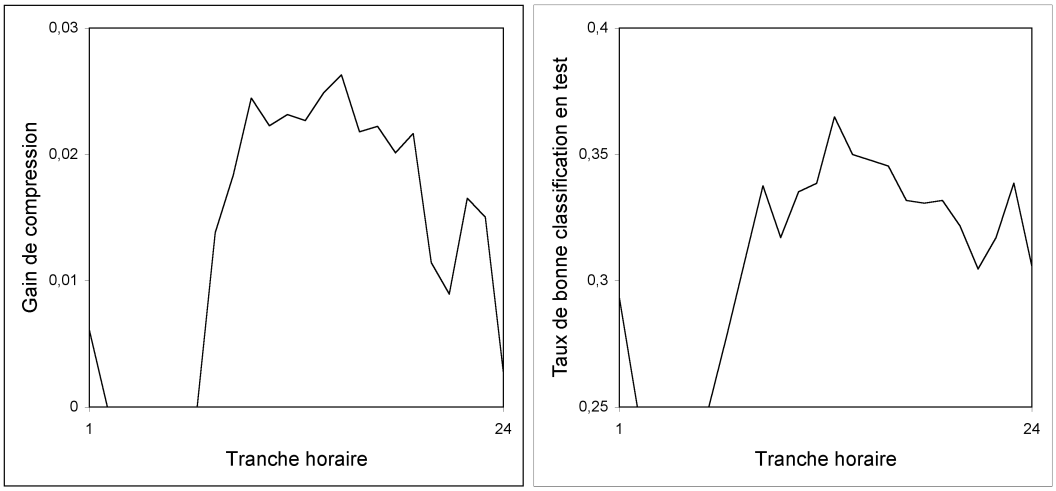

FIG. 6.4 – Gain de compression (en apprentissage) et taux de bonne prédiction en test pour chaque tranche horaire. Le gain de compression repose sur la séparation de l'espace en zones homogènes vis-à-vis de la cible, le taux de bonne prédiction mesure la séparabilité des classes. Le gain de compression et le taux de bonne classification sont corrélés.

d'aucun intérêt. C'est la régularisation, couplée avec le fait que les partitions considérées fournissent des capacités de discrimination allant d'un minimum (un seul groupe) à un maximum (autant de groupes que d'instances), qui rend possible une telle conclusion.

Les partitions construites pour chaque variable sont utiles à la sélection. Leurs caractéristiques peuvent être exploitées afin de sélectionner quelques variables supplémentaires. Par exemple, il est possible de conserver uniquement les variables pour lesquelles la partition construite est constituée de plus de 4 groupes. L'analyste, s'il a du temps, peut s'aider des visualisations pour faire "manuellement" son choix. Il peut tout aussi bien fixer a priori le nombre de variables à conserver, ou un seuil que le gain de compression doit ou ne doit pas dépasser.

6.2.2 Sélection d'une métrique

Jusqu'ici, nous avons utilisé la métrique L_1 pour mesurer la distance séparant deux profils. Eva permet de comparer différentes mesures de similitude.

Expérience. – Nous reproduisons l'expérience précédente en appliquant Eva à chacune des 24 tranches horaires et nous considérons deux métriques supplémentaires : la métrique euclidienne et un noyau gaussien (définissant une métrique euclidienne dans un espace implicite). Les courbes de gain de compression sont reportées sur la fig.6.5.

Résultats. – Certains comportements des profils sont analogues. Par exemple, quelle que soit la métrique ici considérée, les tranches horaires de fin de nuit sont déclarées non pertinentes relativement à la cible considérée. Mais c'est l'utilisation de la métrique L_1 qui

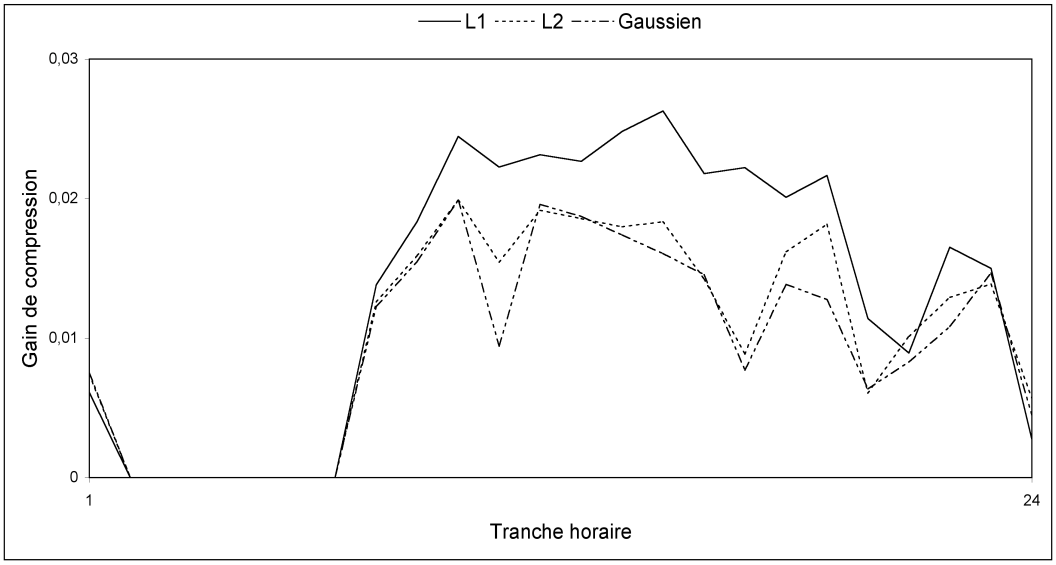

FIG. 6.5 – Gain de compression pour chaque tranche horaire et pour trois métriques. La métrique L_1 est la plus pertinente des trois.

conduit aux meilleurs gains de compression, quasiment pour toutes les tranches horaires. Pour ces variables séquentielles et cette cible, l'analyste est conduit, automatiquement et de manière fiable à choisir cette métrique au détriment des deux autres. S'il dispose de temps, il peut même s'aider de la visualisation proposée à la section précédente pour expliquer les différences de comportement sur chaque tranche horaire.

6.3 Le classifieur et l'estimateur de densité

Eva n'est pas seulement une méthode d'évaluation et possède d'autres modes d'utilisation. La partition obtenue induit un classifieur et une densité conditionnelle. Eva définit donc une règle de classification et un estimateur de densité conditionnelle. Nous illustrons maintenant ces deux aspects, toujours à partir d'expériences sur les données de consommation téléphonique.

6.3.1 Comparaison avec d'autres méthodes de classification

A partir de la partition construite par Eva, nous définissons un classifieur en procédant à un vote majoritaire dans chaque cellule. Autrement dit, nous appliquons Eva en tant que règle de classification.

Méthodes de classification. – Nous comparons notre règle de classification à d'autres règles de classification. Ces règles, décrites par exemple dans [Hastie *et al.*, 2001], sont :

- l'analyse discriminante linéaire (ADL),
- l'analyse discriminante quadratique (ADQ),

- l'analyse discriminante de mélange (ADM),
- l'analyse discriminante flexible (MARS),
- la régression logistique (Rlog),
- la classification par le plus proche voisin (PPV),
- la méthode LVQ,
- les séparateurs à vastes marges linéaire (SVMLin) et à noyau gaussien (SVMGauss),
- la classification bayésienne naïve (CBN),
- l'algorithme CART.

Expérience. – Le paramétrage de ces méthodes est celui proposé par défaut. La question du paramétrage concerne essentiellement la méthode LVQ, l'analyse discriminante de mélange et, dans une moindre mesure, la classification par le plus proche voisin, le SVM à noyau gaussien et le classifieur naïf. Le *Niveau* de RVVGlouton est fixé à 6.

Nous lançons ces méthodes sur les données d'apprentissage relatives à la consommation hebdomadaire. Nous considérons donc les 168 variables descriptives. Nous calculons sur l'échantillon de test le pourcentage de mauvaise classification du classifieur obtenu pour chaque méthode. Ces pourcentages sont reportés dans le Tableau 6.1. De plus, nous effectuons un test de Mac Nemar à 5% de significativité de la différence entre le taux d'erreur en test d'Eva et celui de chaque classifieur concurrent.

MARS	**SVMGauss**	**CART**	**Eva**	**Rlog**	**ADL**
57.3	58.1	59.7	60.4	62.6	63.1
ADM	**SVMLin**	**PPV**	**ADQ**	**LVQ**	**CBN**
63.5	*64.1*	*67.5*	*68.6*	*70.2*	*70.7*

TAB. 6.1 – Taux d'erreur en test des classifieurs construits par plusieurs règles de classification à partir des données d'apprentissage. Les taux en italique sont ceux pour lesquels la différence avec le taux d'erreur d'Eva est significative à 5% suivant le test de proportion de Mac Nemar.

Discussion des résultats. – Nous constituons trois groupes à partir des résultats obtenus. Le premier contient les méthodes dont le biais n'est pas adapté au jeu de données considéré : le classifieur naïf bayésien, l'algorithme LVQ de Kohonen, l'analyse discriminante quadratique et la règle du plus proche voisin. Leurs performance vont de 70.7% à 67.5%. Par exemple, le classifieur bayésien naïf (70.7%) travaille avec une hypothèse d'indépendance entre les variables descriptives et une hypothèse gaussienne sur chacune d'entre elles. Toutes les deux sont ici peu réalistes (les variables sont corrélées par le temps et sont plutôt poissoniennes).

Un deuxième groupe de méthodes contient : le SVM linéaire, l'analyse discriminante de mélange, l'analyse discriminante linéaire et la régression logistique. Les performances de ces méthodes vont de 63.8% à 62.6%. Ces méthodes cherchent à séparer linéairement les classes, sans avoir de latitude sur le nombre de séparations autorisées. L'analyse discriminante linéaire est par exemple conçue pour construire un groupe par classe. Ces méthodes sont donc contraintes et produisent des classifieurs aux performances similaires.

Un troisième groupe contient les méthodes qui sont des raffinements de certaines des méthodes précédentes : Eva, l'arbre de décision CART, le SVM à noyau gaussien et l'analyse discriminante flexible. Ces méthodes, qui en ajustant des séparatrices non linéaires, qui en choisissant un bon nombre de groupes, améliorent de beaucoup les résultats : l'emploi d'un noyau gaussien fait passer le pourcentage d'erreur du SVM de 63.8% à 58.2%, l'utilisation de splines fait tomber ce pourcentage de 63.1% à 57.3% pour l'analyse discriminante.

La sélection d'instances effectuée par Eva fait passer de 67.5% à 60.4% le pourcentage d'erreur de la classification par le plus proche voisin. Cela confirme qu'une bonne sélection d'instances est la clé permettant de déployer un modèle à base de plus proche voisin, notamment lorsqu'on sélectionne 6 instances sur 1768 comme c'est la cas de notre méthode. C'est également l'occasion de rappeler que notre méthode fournit un support visuel explicatif pertinent du modèle construit (*c.f.* section 6.1), ce qui n'est pas le cas de toutes les méthodes utilisées ici (et des perceptrons multi-couches, par exemple).

6.3.2 Exploitation de l'estimation de densité

A partir de la partition construite par Eva, une densité conditionnelle est définie en considérant la distribution fréquentielle des étiquettes dans chaque cellule. La méthode peut donc être employée comme estimateur de densité conditionnelle. Pour illustrer l'intérêt d'une telle estimation, nous procédons à l'expérience suivante.

Expérience. – Sur le problème de classification de profils de consommation, considérons les 7 jours de la semaine indépendamment. Nous disposons alors de 7 variables séquentielles $X_1, \ldots, X_7$, chacune de longueur 24. L'application de notre méthode à chacune d'entre elles (avec la métrique L_1 et le $Niveau$ à 4) fournit une estimation des probabilités conditionnelles $p(Y/X_l)$ de la variable cible Y.

Construction d'un classifieur bayésien. – Le principe de classification du Maximum A Posteriori (analogue au principe MAP de modélisation) préconise, pour une réalisation $(x_1, \ldots, x_7)$ de $(X_1, \ldots, X_7)$, d'attribuer l'étiquette j maximisant la probabilité $p(Y = j/X_1 = x_1, \ldots, X_7 = x_7)$. En adoptant l'*hypothèse naïve* d'indépendance des X_l conditionnellement à la cible, cela revient à maximiser la probabilité $p(Y = j)p(Y = j/X_1 = x_1) \ldots p(Y = j/X_7 = x_7)$. A partir des estimations $p(Y/X_l)$, on obtient ainsi un classifieur tenant compte de chacune des variables.

Discussion des résultats. – Le pourcentage de mauvaise classification de ce classifieur sur l'échantillon test est de 58.2%. Ce pourcentage d'erreur est à comparer avec celui du classifieur bayésien naïf, estimé précédemment à 70.7%. Ce classifieur faisait l'hypothèse d'indépendance des 168 tranches horaires. Nous avons pour notre part seulement supposé l'indépendance des 7 jours. Cette hypothèse étant moins forte, il n'est pas étonnant d'améliorer la performance.

De plus, le classifieur bayésien naïf utilisé fait l'hypothèse que chacune des 168 variables est gaussienne, ce qui est inadapté (une hypothèse poissonnienne serait certainement plus adaptée) et contribue à sa mauvaise performance. Pour notre part, nous ne faisons aucune

hypothèse sur la forme des données. Nos hypothèses, les partitions de Voronoi, sont non paramétriques au sens statistique et ne contraignent pas l'évaluation.

L'hypothèse d'indépendance conditionnelle entre les variables est un moyen simple d'obtenir un classifieur à partir de variables dont on est capable d'estimer les probabilités conditionnelles. Le classifieur obtenu ainsi à peu de frais constitue une référence pour la modélisation : toute construction plus élaborée d'un classifieur doit conduire à un meilleur résultat si elle se veut utile. Un intérêt de notre méthode est de permettre à l'analyste d'intégrer des variables séquentielles directement dans ce modèle naïf, à côté de variables statiques.

6.4 Conclusion

Nous avons illustré les apports de notre méthode d'évaluation supervisée à l'aide d'un problème de classification de profils de consommation téléphonique.

Tout d'abord, la méthode est constructive et l'évaluation par le gain de compression s'accompagne d'un support explicatif : une partition des individus. Grâce à cette partition, l'utilisateur qui le souhaite peut visualiser la décision qui a été prise.

Nous avons ensuite employé la méthode dans le but pour lequel elle a été construite : la sélection de variable séquentielle. Le gain de compression constitue un indicateur suivant lequel comparer différentes variables séquentielles. Eva est capable de détecter de manière fiable les variables non pertinentes pour le problème traité. Dans ce cas, elle construit un unique groupe et renvoie un gain de compression nul. Nous avons atteint l'objectif défini initialement : intégrer les variables séquentielles dans un traitement filtre univarié de la sélection.

Nous avons appliqué Eva en tant que méthode de sélection d'instances. Il ressort que la sélection d'instances opérée améliore la performance de la règle de classification par le plus proche voisin de plus de 8%, pour 6 individus conservés sur 1768. De plus, la performance de ce classifieur est une des meilleures parmi celles des modèles testés.

Enfin, nous avons vu comment l'estimation de densité conditionnelle induite par Eva permet d'inclure les variables séquentielles dans un modèle naïf. Il fallait auparavant soit construire des indicateurs statiques à partir de chaque variable séquentielle soit supposer l'indépendance des mesures entre chaque temps. En le rendant ainsi moins naïf, nous avons produit un classifieur bayésien dont la performance le place en seconde position parmi la douzaine de classifieurs testés.

7

Automatisation de la recherche de représentations

Sommaire

Nous illustrons dans ce chapitre la souplesse et la généricité de notre approche de l'évaluation en traitant deux problèmes de représentation d'une variable séquentielle, toujours dans le cadre de la classification supervisé.

Dans la section 7.1, nous nous intéressons à la question de la sélection d'un ensemble de temps de mesure. Il s'agit, pour une variable séquentielle composée de F temps de mesure, de sélectionner le sous-ensemble de temps de mesure le plus pertinent vis-à-vis de la variable cible. Ceci permet de réduire la dimensionalité de la variable séquentielle dans les cas où F est grand.

Dans la section 7.2, nous traitons le problème du fenêtrage d'une variable séquentielle. Pour telle variable, il s'agit de partitionner le domaine temporel en intervalles successifs et de considérer la moyenne des mesures sur chaque intervalle. Là encore, cela permet de réduire la dimensionalité de la variable séquentielle tout en synthétisant l'information.

Le travail décrit ici l'est au titre de perspectives avancées. Il n'a pas donné lieu à une implantation ou été soumis à des expérimentations. L'objectif est de montrer que notre démarche est générique et se prête au traitement d'autres problèmes d'évaluation pour lesquels l'ensemble des hypothèses est dépendant des données.

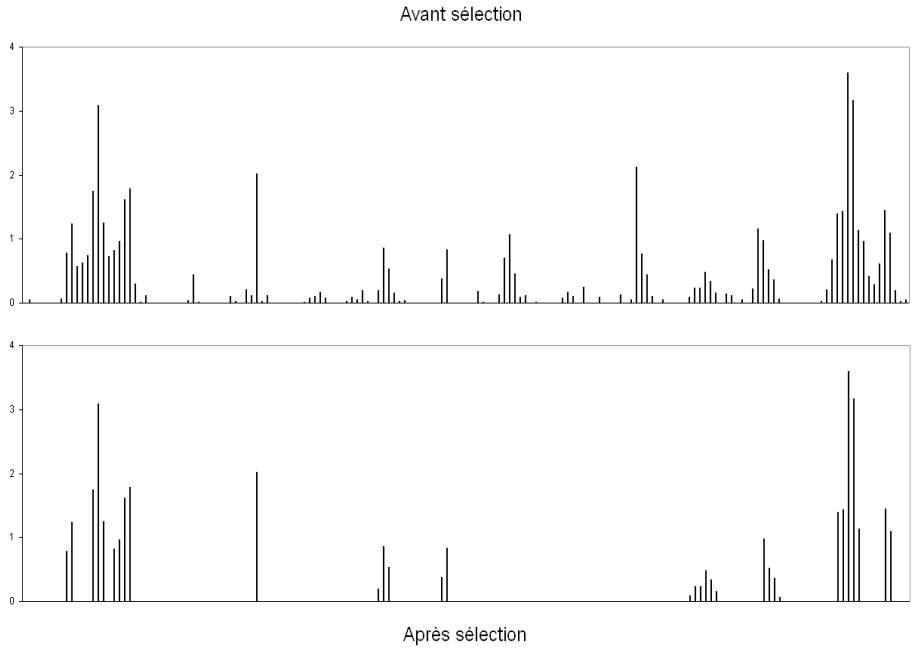

FIG. 7.1 – Exemple de sélection des temps de mesure sur un profil hebdomadaire de consommation éléphonique.

7.1 Sélection supervisée des temps de mesure

Soit $S : \mathcal{I} \to \mathbb{X}$ une variable séquentielle. Nous nous plaçons dans le cas suivant : l'espace de représentation $\mathbb{X}$ est l'ensemble des suites finies de longueur F à valeurs dans un ensemble $\mathbb{X}'$. Nous pouvons alors écrire $S = (S_f)_{1 \leq f \leq F}$, avec $S_f : \mathcal{I} \to \mathbb{X}'$ la variable correspondant au f^{eme} temps de mesure.

Nous considérons ici le problème de la sélection des temps de mesure (*c.f.* fig.7.1), dans le cadre de la classification supervisée. Nous proposons d'automatiser cette sélection en proposant un critère d'évaluation et une heuristique d'optimisation.

Hypothèse de travail. – Pour tout $E \leq F$, nous supposons l'ensemble des suites finies de longueur E muni d'une mesure de similitude symétrique δ_E.

7.1.1 Critère d'évaluation

Nous définissons dans un premier temps les hypothèses à considérer et proposons dans un second temps un protocole de description conduisant à la spécification d'une distribution a priori et d'une fonction de vraisemblance, en accord avec la démarche adoptée pour développer Eva au chapitre 3.

Ensemble des hypothèses. – Notons $\mathcal{P}_F$ l'ensemble des parties de l'intervalle entier $[\![1, F]\!]$. Pour chaque ensemble de temps de mesure $P \in \mathcal{P}_F$, un espace de représentation

$\mathbb{X}(P)$ est défini à partir de $\mathbb{X}$ en ne considérant que les temps de mesure contenus dans P.

Soit $P \in \mathcal{P}_F$ de cardinal E. D'après notre hypothèse de travail, nous disposons d'une mesure de similitude symétrique δ_E sur l'ensemble des suites indicées par P. Nous définissons une matrice de Gram δ_P sur l'ensemble des individus par transfert des propriétés.

L'ensemble des hypothèses $\mathcal{H}$ que nous considérons est l'ensemble des couples (P, H) où P est un ensemble de temps de mesures inclus dans $\mathcal{P}_F$ et H une partition de Voronoi induite par δ_P, *i.e.* un élément de $\mathcal{H}_{\delta_P}(\mathcal{I})$.

Si l'ensemble des temps de mesure sélectionnés est vide, l'ensemble des partitions de Voronoi est vide. Cette hypothèse constitue l'hypothèse nulle.

Nous avons étendu l'ensemble des hypothèses considéré au chapitre 3 en ajoutant une étape préliminaire de sélection des temps de mesure. Un critère d'évaluation MDL de ces hypothèses est alors obtenu par extension du critère MDL utilisé par Eva. Nous commençons par proposer un protocole de description.

Protocole de description. – Soit (P, H) une hypothèse. Notons E le cardinal de P et reprenons les notations relatives aux partitions de Voronoi. Nous proposons un protocole de description d'une hypothèse (P, H) et de la variable cible, décrit à la fig.7.2. Plus précisément, nous complétons le protocole de description d'un ensemble de prototypes et d'une variable cible proposé au chapitre 3 en ajoutant une étape préliminaire de description des temps de mesure sélectionnés.

- description du nombre E de temps de mesure,
- description des E temps de mesure,
- description du nombre K de cellules de la partition,
- description des K individus définissant la partition,
- pour chaque groupe, description des J fréquences des étiquettes,
- pour chaque groupe, description de l'étiquetage des individus.

FIG. 7.2 – Protocole de description pour un problème de sélection des temps de mesures.

Critère d'évaluation. – Soit (P, H) une hypothèse et notons E le cardinal de P. Précisons la distribution a priori et la fonction de vraisemblance. Nous considérons le nombre E de temps de mesure comme un élément de l'ensemble $[\![0, F]\!]$. Nous choisissons la distribution uniforme sur cet ensemble, ce qui conduit à une longueur de description de $\log F + 1$ nats d'après la correspondance de Shannon.

Si E est nul, il n'y a ni temps de mesure ni partition de Voronoi à décrire. On passe alors directement à la description de la variable cible. Si $E > 0$, nous considérons le sous-ensemble des E temps de mesure comme un élément de l'ensemble des E-combinaisons avec répétition dans un ensemble à F éléments. C'est un ensemble de cardinal $\binom{F+E-1}{E}$.

Nous adoptons un a priori uniforme sur cet ensemble et obtenons une longueur de description égale à $\log \binom{F+E-1}{E}$ nats.

Le reste de la description correspond exactement au cas traité au chapitre 3. La distribution a priori et la longueur de description fixées sur l'ensemble des partitions de Voronoi sont inchangées et la fonction de vraisemblance sur les données pour chaque hypothèse est identique. Le critère d'évaluation d'une hypothèse (P, H) s'écrit :

$$\begin{aligned} c(P,H) &= \log(F+1) + \log \binom{F+E-1}{E} + \log N + \log \binom{N+K-1}{K} \\ &\quad + \sum_{k=1}^{K} \log \binom{N_k+J-1}{J-1} + \sum_{k=1}^{K} \log \frac{N_k!}{N_{k1}! \dots N_{kJ}!}, \end{aligned} \tag{7.1}$$

lorsque P est non vide, et l'hypothèse nulle est évaluée par la formule

$$c^0 = \log(F+1) + \log \binom{N+J-1}{J-1} + \log \frac{N!}{N_{\cdot 1}! \dots N_{\cdot J}!}, \tag{7.2}$$

où $N_{\cdot j}$ est le nombre d'individus portant l'étiquette j.

7.1.2 Heuristique d'optimisation

Nous avons fixé un ensemble d'hypothèses relatives au problème de sélection des temps de mesure et proposé un critère d'évaluation de ces hypothèses. Nous proposons une heuristique d'optimisation.

Heuristique d'optimisation. – Nous envisageons une heuristique de parcours incrémental uniforme des variables $S_1, \dots, S_F$. Un ensemble de prototypes est initialement constitué, de taille fixée K. A chaque étape, un temps de mesure est ajouté à l'ensemble P des temps de mesure déjà sélectionnés et l'algorithme GLOUTON est appliqué à l'ensemble de prototypes pour la mesure de similitude δ_P. Si on aboutit à une meilleure solution, l'ensemble de prototypes obtenu est complété uniformément pour atteindre la taille fixée initialement et le temps de mesure est conservé. Sinon, on recommence avec un autre temps de mesure. L'algorithme est décrit à la fig.7.3.

L'intérêt d'un tel parcours est d'assurer l'évaluation de l'hypothèse nulle et, surtout, de permettre une implantation optimisée. Lorsque la mesure de similitude s'écrit comme une somme sur les variables, ce qui est le cas des métriques höldériennes (la distance euclidienne, par exemple), il suffit à chaque étape d'ajouter la contribution de la variable considérée.

Optimisation et complexité. – Supposons que la mesure de similitude se décompose en une somme sur les variables. En plus des N listes triées de longueur K mises en place pour optimiser le GLOUTON, on conserve les NK distances. A chaque itération, les NK distances sont mises à jour à coût constant en additionnant la contribution de la

- $P \leftarrow \emptyset$
- $BestCost \leftarrow$ évaluer l'hypothèse nulle
- $(S_{(1)}, \ldots, S_{(F)}) \leftarrow$ réordonner uniformément $S_1, \ldots, S_F$
- $H_0 \leftarrow$ sélectionner uniformément un ensemble de K prototypes
- **Pour** $f = 1$ à F **Faire**
  - $P' \leftarrow P \bigcup \{(f)\}$
  - $H' \leftarrow$ GLOUTON(H_0) avec la mesure de similitude associée à P'
  - $Cost \leftarrow$ coût de l'hypothèse (P', H')
  - **Si** $Cost < BestCost$
    - $BestCost \leftarrow Cost$
    - $(P, H) \leftarrow (P', H')$
    - $H_0 \leftarrow$ ajouter des prototypes à H à hauteur de K
- **Retourner** (P, H)

FIG. 7.3 – Heuristique de sélection des temps de mesure.

variable dont l'ajout est considéré lors de cette itération. Puis les N listes de longueur K sont triées. Le GLOUTON est alors de complexité temporelle un $O(NK \log K)$, au lieu d'un $O(NK(F + \log F))$ si on recalculait entièrement les distances à chaque étape. Ainsi, l'algorithme proposé est de complexité temporelle un $O(FNK \log K)$, de complexité spatiale un $O(NK)$.

Notons que cet algorithme peut être itéré et complété. Par exemple, il est possible d'intégrer l'optimisation incrémentale dans une boucle et de la répéter jusqu'à ce qu'aucune variable ne soit plus ajoutée. La complexité temporelle serait alors un $O(F^2NK \log K)$. On peut également envisager l'ajout d'une passe décrémentale sur l'ensemble des variables ; ou encore, effectuer une passe finale de RVVGlouton avec un niveau d'optimisation non nul pour améliorer un peu plus l'ensemble des prototypes.

Un travail d'optimisation, de développement et de validation est nécessaire afin de limiter la complexité de la méthode finale et d'assurer que le temps passé à optimiser est bien exploité par l'algorithme.

7.2 Selection supervisée d'un fenêtrage

Soit $S : \mathcal{I} \rightarrow \mathbb{X}$ une variable séquentielle. Nous nous plaçons dans le cas suivant : l'espace de représentation $\mathbb{X}$ est l'ensemble des suites finies de longueur F à valeurs dans un ensemble $\mathbb{R}$ et nous écrivons $S = (S_f)_{1 \leq f \leq F}$, avec $S_f : \mathcal{I} \rightarrow \mathbb{R}$ la variable correspondant au f^{eme} temps de mesure.

Nous considérons ici le problème du fenêtrage du domaine des temps de mesure (*c.f.* fig.7.4). Nous automatisons cette sélection en proposant un critère d'évaluation et une heuristique d'optimisation.

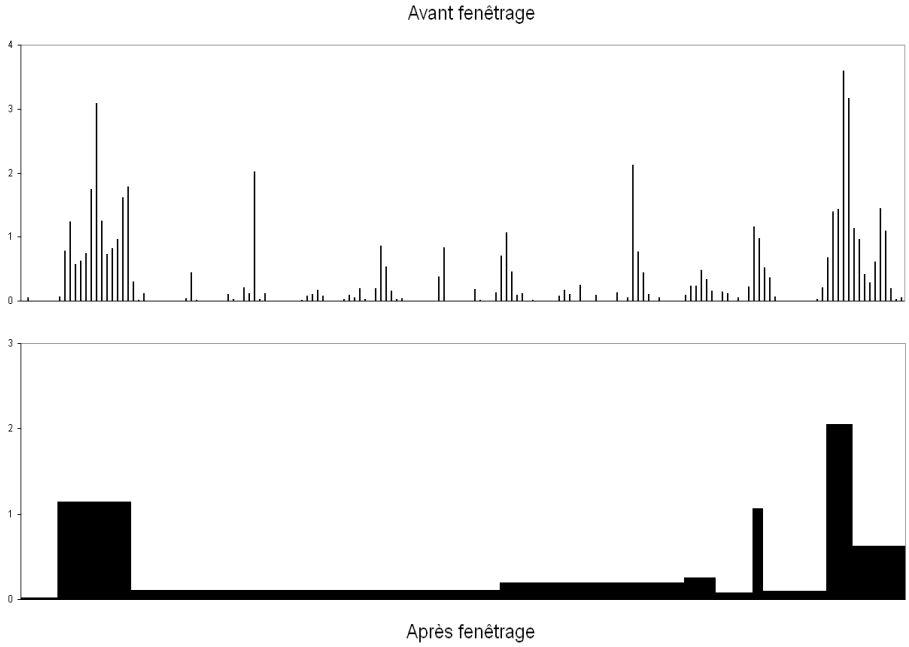

FIG. 7.4 – Exemple de fenêtrage sur un profil hebdomadaire de consommation téléphonique.

Hypothèse de travail. – Pour tout $f \in \mathbb{N}$, nous supposons l'ensemble des suites finies à valeurs réelles de longueur f muni d'une mesure de similitude symétrique δ_f.

7.2.1 Critère d'évaluation

Nous définissons les hypothèses à considérer et proposons un protocole de description conduisant à la spécification d'une distribution a priori et d'une fonction de vraisemblance, en accord avec la démarche adoptée pour développer Eva au chapitre 3.

Ensemble des hypothèses. – Notons π_F l'ensemble des partitions en intervalles entiers de l'intervalle entier $[\![1, F]\!]$. Pour chaque partition $\pi \in \pi_F$, une variable $S(\pi) : \mathcal{I} \to \mathbb{R}^E$ est définie en calculant la valeur moyenne de la suite de mesures sur chaque intervalle de π. Autrement dit, à tout individu n, on associe le E-uplet des moyennes des mesures sur chaque intervalle.

Soit $\pi \in \pi_F$ de cardinal E. D'après notre hypothèse de travail, nous disposons d'une mesure de similitude symétrique δ_E sur l'espace de représentation de $S(\pi)$. Nous obtenons une matrice de Gram δ_π sur l'ensemble des individus par transfert des propriétés.

L'ensemble des hypothèses $\mathcal{H}$ que nous considérons est l'ensemble des couples (π, H) où π est un élément de π_F et H une partition de Voronoi induite par δ_π, *i.e.* un élément de $\mathcal{H}_{\delta_\pi}(\mathcal{I})$.

Nous avons étendu l'ensemble des hypothèses considéré au chapitre 3 en ajoutant une

étape préliminaire de sélection d'un fenêtrage. Nous proposons un critère d'évaluation MDL de ces hypothèses. Nous proposons pour commencer un protocole de description.

Protocole de description. – Soit (π, H) une hypothèse. Notons E le cardinal de π et reprenons les notations relatives aux partitions de Voronoi. Le protocole de description d'une hypothèse (π, V) et de la variable cible que nous proposons est décrit à la fig.7.5. Nous complétons le protocole de description d'un ensemble de prototypes et d'une variable cible proposé au chapitre 3 en ajoutant une étape préliminaire de description du fenêtrage.

- description du nombre E d'intervalles,
- description de la partition en E intervalles,
- description du nombre K de cellules de la partition,
- description des K individus définissant la partition,
- pour chaque groupe, description des J fréquences des étiquettes,
- pour chaque groupe, description de l'étiquetage des individus.

FIG. 7.5 – Protocole de description pour un problème de fenêtrage d'une variable séquentielle.

Critère d'évaluation. – Soit (π, H) une hypothèse et notons E le cardinal de π. Nous proposons maintenant une distribution a priori et une fonction de vraisemblance afin de définir les longueurs de description.

Nous considérons le nombre E de fenêtres comme un élément de l'ensemble $[\![1, F]\!]$. Nous choisissons la distribution uniforme sur cet ensemble, ce qui conduit à une longueur de description de $\log F$ nats d'après la correspondance de Shannon.

Nous considérons la partition π comme un élément de l'ensemble des partitions en E intervalles éventuellement vides. Le cardinal de cet ensemble est égal à $\binom{F+E-1}{E-1}$. Nous adoptons un a priori uniforme sur cet ensemble et obtenons une longueur de description égale à $\log \binom{F+E-1}{E-1}$ nats.

Le reste de la description correspond exactement au cas traité au chapitre 3. La distribution a priori et la longueur de description fixées sur l'ensemble des partitions de Voronoi sont inchangées et la fonction de vraisemblance sur les données pour chaque hypothèse est identique. Le critère d'évaluation d'une hypothèse (π, H) s'écrit :

$$\begin{aligned} c(\pi, H) &= \log F + \log \binom{F+E-1}{E-1} + \log N + \log \binom{N+K-1}{K} \\ &\quad + \sum_{k=1}^{K} \log \binom{N_k + J - 1}{J-1} + \sum_{k=1}^{K} \log \frac{N_k!}{N_{k1}! \dots N_{kJ}!}. \end{aligned} \tag{7.3}$$

7.2.2 Heuristique d'optimisation

Nous avons fixé un ensemble d'hypothèses relatives au problème du fenêtrage d'une variable séquentielle et proposé un critère d'évaluation de ces hypothèses. Nous proposons ici une heuristique d'optimisation.

Heuristique d'optimisation. – Nous proposons une heuristique fonctionnant en deux temps : les prototypes sont fixés et le fenêtrage optimisé, puis le meilleur fenêtrage obtenu est fixé et l'ensemble de prototypes optimisé.

Soit K le nombre de prototypes à considérer. Les prototypes sont sélectionnés uniformément. Nous proposons une heuristique gloutonne agrégative de parcours des partitions de l'intervalle de mesure. A chaque étape, toutes les fusions possibles de deux intervalles adjacents sont évaluées tour à tour. La meilleure partition obtenue est conservée et l'optimisation gloutonne est réitérée. La partition initiale est la partition la plus fine (chaque intervalle contient un unique temps de mesure) et la partition finale est la partition grossière (les temps de mesure sont tous regroupés dans un unique intervalle).

Le meilleur fenêtrage obtenu au cours de cette recherche est ensuite fixé. Nous proposons d'appliquer alors l'algorithme RVVGlouton pour optimiser l'ensemble des prototypes fixé initialement. L'algorithme final est décrit à la fig.7.6.

- $\pi_{opt} \leftarrow$ la partition la plus fine de l'intervalle des temps de mesure
- $H_{opt} \leftarrow$ sélectionner aléatoirement un ensemble de K prototypes
- $\pi_0 \leftarrow \pi_{opt}$
- $H_0 \leftarrow H_{opt}$
- $BestCost \leftarrow$ évaluer (π, H_0)
- **Pour** $f = F - 1$ à 1 **Faire**
  - $LocallyBestCost \leftarrow +\infty$
  - $\pi \leftarrow \pi_0$
  - **Pour** tout couple (π_i, π_j) d'intervalles adjacents de π_0 **Faire**
    - $\pi' \leftarrow$ fusionner π_i et π_j
    - $Cost \leftarrow$ coût de l'hypothèse (π', H_0)
    - **Si** $Cost < LocallyBestCost$
      - $LocallyBestCost \leftarrow Cost$
      - $\pi \leftarrow \pi'$
  - **Si** $LocallyBestCost < BestCost$
    - $BestCost \leftarrow LocallyBestCost$
    - $\pi_{opt} \leftarrow \pi$
- $H_{opt} \leftarrow$ RVVGlouton$(H_0, Niveau)$, avec la mesure de distance induite par π_{opt}
- **Retourner** (π_{opt}, H_{opt})

FIG. 7.6 – Heuristique de sélection d'un fenêtrage.

Complexité. – Une implantation directe de la première partie de cette heuristique

conduit à une complexité temporelle un $O(F^3KN)$. En effet, au cours de chacune des $F-1$ étapes, au plus F fusions sont envisagées, et chaque fusion nécessite le calcul de la partition de Voronoi de l'ensemble des individus, en $O(FKN)$. La complexité de la seconde étape est celle de RVVGlouton.

Notons la présence d'un terme cubique en le nombre de temps de mesure dans la complexité de cette heuristique. Ceci contraint son application à des données séquentielles de faible longueur. Cette heuristique doit donc être optimisée ou adaptée. Ceci constitue un travail en soi, qui dépasse le cadre du simple propos illustratif que nous nous sommes fixé dans ce chapitre.

7.3 Conclusion

Nous avons ici mis en avant la démarche proposée au chapitre 3, en l'appliquant à deux problèmes de représentation. Nous avons ainsi proposé une méthode de sélection supervisée des temps de mesures et une méthode de fenêtrage supervisée. Nous avons ainsi, entre autres, montré la capacité de l'approche descriptive à fournir des critères prévenant le risque de sur-apprentissage, non paramétriques et se passant d'ensemble de validation. Notre démarche, résumée à la fig.7.7, assure que le résultat de ces méthodes est pertinent et fiable.

1. Formuler des hypothèses, en nombre fini et dépendantes des données.
2. Adopter une approche descriptive de l'évaluation.
3. Développer une heuristique d'optimisation combinatoire efficace.

FIG. 7.7 – Notre démarche conduisant à l'obtention de méthodes d'évaluation non paramétriques pertinentes et fiables.

Plus prosaïquement, nous avons proposé deux méthodes permettant d'aller plus loin en préparation de données séquentielles : en plus de fournir une évaluation de la variable séquentielle, nous automatisons la recherche d'un bon espace de représentation. Si les critères sont ici bien posés, il nous reste à mener un travail de développement et de validation non négligeable.

Conclusion

Dans cette troisième partie, nous avons mené des expériences sur des données réelles afin d'illustrer l'apport et les différents modes d'utilisation de notre méthode, Eva. Nous avons également montré la souplesse de notre approche de l'évaluation en automatisant la recherche d'un bon espace de représentation.

Dans le chapitre 6, nous avons montré que le gain de compression calculé par Eva s'accompagne d'un support explicatif visualisable. Ceci est utile, car l'analyste peut ainsi comprendre et expliquer son résultat. Nous avons appliqué notre méthode en tant que règle de sélection d'instances pour la classification par le plus proche voisin. La performance du classifieur construit s'est avérée bonne et, surtout, très au-dessus de celle de la classification par le plus proche voisin exploitant tous les individus. Dans la mesure où nous n'avons conservé que 6 individus, ce résultat milite en faveur du déploiement de tels modèles.

Nous avons également montré comment l'estimation des probabilités conditionnelles induite par notre méthode pouvait être exploitée sous une hypothèse naïve. Toute variable à partir de laquelle on est capable de construire une mesure de similitude (les variables séquentielles en sont un exemple) peut alors être intégrée naïvement et à peu de frais dans un classifieur. Dans une étude, ce modèle peut servir de référence : sa performance est la performance minimale que doit atteindre un modèle plus élaboré. Notons que l'expérience menée a produit un classifieur naïf dont la performance est supérieure à celles de tous les classifieurs testés sauf un.

Enfin, dans le chapitre 7, notre intérêt s'est porté sur l'automatisation de deux problèmes de représentation de données séquentielles : la sélection supervisée d'un ensemble de temps de mesure et le fenêtrage supervisé. Dans ces deux cas, nous avons proposé un ensemble d'hypothèses et appliqué notre démarche de l'évaluation afin d'obtenir un critère d'évaluation supervisé. Nous avons également proposé des heuristiques d'optimisation. Ces méthodes nécessitent un travail de développement, de test et de validation. Le but est de montrer que notre approche de l'évaluation est générique et conduit à l'obtention de méthodes d'évaluation automatiques, fines et fiables.

Bilan et perspectives

Bilan

L'objectif informel du travail reporté dans ce document est de répondre à la question : comment éviter le saucissonage d'une variable séquentielle en préparation de données dans le cas de la classification supervisée ?

Processus de résolution

Pour atteindre cet objectif, nous avons tout d'abord explicité la question posée. Nous avons placé la phase de préparation de données dans son contexte, celui d'un processus de fouille de données, et avons décrit l'approche filtre univariée de la préparation des variables. Nous avons précisé la notion de variable séquentielle et avons montré qu'en l'état actuel des choses une telle variable doit être saucissonée pour être traitée dans une approche filtre.

Nous avons alors précisé un objectif formel : disposer d'une méthode d'évaluation supervisée d'une variable séquentielle. Les données sont d'autant plus soumises à un travail de normalisation qu'elles sont séquentielles. En parcourant la littérature, il est apparu que ces données sont susceptibles de prendre différentes formes. Quelle que soit cette forme, nous avons constaté qu'il est toujours possible de définir une mesure de similitude entre les individus à partir d'une représentation.

Nous avons donc abstrait la question de l'évaluation d'une variable séquentielle en un problème d'évaluation d'une mesure de similitude, que nous avons résolu en trois temps. Nous avons commencé par formuler le problème en un problème de recherche de l'hypothèse la plus informative dans un ensemble d'hypothèses que nous avons fixé. Puis nous avons proposé un critère d'évaluation supervisée de ces hypothèses. Enfin, nous avons développé un algorithme de recherche de la meilleure hypothèse. Eva, la méthode ainsi obtenue, évalue la pertinence d'une mesure de similitude relativement à une variable cible catégorielle.

Propriétés de la solution

Les hypothèses que nous considérons ne sont ni des classifieurs ni des densités de probabilité et sont dépendantes des données. Elles n'entrent pas dans le cadre des approches statistiques classiques de l'évaluation. Nous nous sommes tournés vers un principe générique d'évaluation, le principe de minimisation des longueurs de description (principe

MDL).

Pour instancier ce principe et obtenir un critère effectif, nous avons montré qu'il suffit de proposer une distribution a priori sur les hypothèses et une fonction de vraisemblance de la variable cible en fonction de chaque hypothèse. C'est ce que nous avons fait et nous avons obtenu un critère non paramétrique prévenant automatiquement le sur-apprentissage. Avec un tel critère, la fiabilité de la décision est assurée sans recourir à un ensemble de validation.

Si les hypothèses considérées ne sont pas originellement des classifieurs ou des densités de probabilité conditionnelle, il est néanmoins possible d'associer un classifieur et une densité à chacune d'entre elles. Nous avons dès lors appliqué la théorie de l'apprentissage statistique et l'inférence bayésienne de densités, et nous avons proposé deux critères non paramétriques prévenant le risque de sur-apprentissage. Nous avons montré que le critère MDL est plus fin que ces deux derniers.

L'ensemble des hypothèses étant de taille exponentielle en le nombre d'individus, la recherche de la meilleure hypothèse nécessite l'emploi d'une heuristique. Nous avons adopté une heuristique gloutonne que nous avons optimisée et encapsulée dans une méta-heuristique de recherche à voisinage variable. L'utilisateur contrôle le temps de calcul alloué à l'optimisation à l'aide d'un unique paramètre. Nous avons montré que l'algorithme exploite efficacement le temps de calcul qui lui est accordé.

Validation de la solution

Eva, notre méthode d'évaluation supervisée d'une mesure de similitude, a été appliquée sur de nombreux jeux de données réelles et synthétiques. Nous avons ainsi montré sa performance et discuté les apports du critère d'évaluation et de l'algorithme d'optimisation. En tant que méthode de sélection d'instance pour la classification par le plus proche voisin, Eva réduit considérablement le nombre d'individus nécessaires tout en conservant la performance prédictive. Nous avons également constaté expérimentalement la fiabilité de la décision prise par Eva.

Nous avons montré comment l'utilisation d'Eva conduit à faire sauter un verrou et permet de faire entrer directement les variables séquentielles dans l'approche filtre univariée de la préparation de variables. Pour cela, nous avons mené des expériences sur des données de consommation téléphonique. Nous avons vu que la partition de l'ensemble des individus qui accompagne la simple évaluation numérique permet une interprétation visuelle du résultat.

Perspectives

Nous envisageons deux types de prolongement du travail reporté dans ce document. Le premier, plutôt applicatif, concerne l'exploitation des sorties et les différents modes d'utilisation de la méthode d'évaluation Eva. Le second, plus prospectif, consiste à résoudre d'autres problèmes d'évaluation.

Exploitation des sorties de la méthode d'évaluation

Eva peut être utilisée en tant que méthode de sélection d'individus pour la classification suivant le plus proche voisin. La validation expérimentale que nous avons menée adopte d'ailleurs ce point de vue. Eva constitue donc une méthode de classification à part entière, pertinente et fiable, concurrente des techniques classiques de modélisation.

Eva est également un estimateur de densité conditionnelle. Pour chaque variable séquentielle, Eva fournit une densité de la variable cible conditionnellement à la variable séquentielle. Ceci permet d'intégrer chacune de ces variables dans un classifieur bayésien, certes naïvement, mais surtout concomitamment à des variables de format quelconque. L'expérience réalisée sur des données réelles a validé cette approche, dans la mesure où la performance du classifieur obtenu est très satisfaisante.

Résolution d'autres problèmes d'évaluation

Nous avons montré comment le critère d'évaluation peut être étendu et l'algorithme de recherche adapté afin d'automatiser la sélection d'une représentation d'une variable séquentielle. Nous avons traité la question de la sélection supervisée des temps de mesure d'une variable séquentielle. Nous avons également considéré le problème du fenêtrage supervisé d'une telle variable. Les méthodes proposées doivent être implantées et validées.

Eva est une méthode de sélection d'individus pour la classification suivant le plus proche voisin. Le critère et l'algorithme peuvent être étendus et adaptés afin d'obtenir une méthode sélectionnant également les variables. La méthode de sélection d'instances et de variables résultante sera toujours non paramétrique, fine et fiable.

Eva réalise une évaluation de la pertinence d'une mesure de similitude vis-à-vis d'une variable cible catégorielle. Suite aux travaux de Boullé et Hue [Boullé and Hue, 2006], il est possible d'obtenir une méthode d'évaluation analogue pour une variable cible numérique. Autrement dit, il est possible de passer de la classification supervisée à la régression. Il s'agirait ensuite de passer au cas non supervisé. Même si le contexte semble éloigné de celui décrit dans ce document, les idées introduites ici peuvent être exploitées. Mais ceci est une autre histoire.

Bibliographie

[Abraham *et al.*, 2003] C. Abraham, P. Cornillon, E. Matzner-Lober, and N. Molinari. Unsupervised curve clustering using B-splines. *Scandinavian journal of statistics*, 30(3) :581–595, 2003.

[Agrawal and Srikant, 1995] R. Agrawal and R. Srikant. Mining sequential patterns. In P.S. Yu and A.S.P. Chen, editors, *Eleventh international conference on data engineering*, pages 3–14, Taipei, Taiwan, 1995. IEEE Computer Society Press.

[Agrawal *et al.*, 1993a] R. Agrawal, C. Faloutsos, and A.N. Swami. Efficient similarity search in sequence databases. In D. Lomet, editor, *Proceedings of the 4th international conference of foundations of data organization and algorithms (FODO)*, pages 69–84, Chicago, Illinois, 1993. Springer Verlag.

[Agrawal *et al.*, 1993b] R. Agrawal, T. Imielinski, and A.N. Swami. Mining association rules between sets of items in large databases. In *Proceedings of the ACM SIGMOD conference on management of data*, pages 207–216, Washington, D.C., 1993.

[Agrawal *et al.*, 1995a] R. Agrawal, K.I. Lin, H.S. Sawhney, and K. Shim. Fast similarity search in the presence of noise, scaling, and translation in time-series databases. In *Twenty-first international conference on very large data bases*, pages 490–501, Zurich, Switzerland, 1995. Morgan Kaufmann.

[Agrawal *et al.*, 1995b] R. Agrawal, G. Psaila, E.L. Wimmers, and M. Zaït. Querying shapes of histories. In U. Dayal, P.M.D. Gray, and S. Nishio, editors, *Twenty-first international conference on very large databases (VLDB '95)*, pages 502–514, Zurich, Switzerland, 1995. Morgan Kaufmann Publishers, Inc. San Francisco, USA.

[Aha *et al.*, 1991] D.W. Aha, D. Kibler, and M.K. Albert. Instance-based learning algorithms. *Machine learning*, 6 :37–66, 1991.

[Baxter and Oliver, 1994] R. Baxter and J.J. Oliver. MDL and MML : similarities and differences (introduction to minimum encoding inference - part iii). Technical Report 207, Department of computer science, Monash university, 1994.

[Bernardo and Smith, 2000] J.M. Bernardo and A.F.M. Smith. *Bayesian theory.* John Wiley & sons, 2000.

[Berndt and Clifford, 1996] D.J. Berndt and J. Clifford. Finding patterns in time series : a dynamic programming approach. Technical report, Advances Knowledge Discovery Data Mining, 1996.

[Bicego *et al.*, 2003] M. Bicego, V. Murino, and M. Figueiredo. Similarity-based clustering of sequences using hidden markov models. In P. Perner and A. Rosenfeld, editors,

Machine learning and data mining in pattern recognition, pages 86–97, Berlin, Germany, 2003. Springer Verlag.

[Blake and Merz, 1996] C.L. Blake and C.J. Merz. UCI repository of machine learning databases, 1996. http ://www.ics.uci.edu/mlearn/MLRepository.html.

[Blum and Langley, 1997] A. Blum and P. Langley. Selection of relevant features and examples in machine learning. *Artificial intelligence*, 97(1-2) :245–271, Décembre 1997.

[Boullé and Hue, 2006] M. Boullé and C. Hue. Optimal bayesian 2D-discretization for variable ranking in regression. In *Ninth international conference on discovery science (DS 2006)*, pages 53–64, 2006.

[Boullé, 2005] M. Boullé. A bayes optimal approach for partitioning the values of categorical attributes. *Journal of machine learning research*, 6 :1431–1452, 2005.

[Boullé, 2006] M. Boullé. MODL : a bayes optimal discretization method for continuous attributes. *Machine learning*, 2006.

[Box and Jenkins, 1994] G.E.P. Box and G. Jenkins. *Time series analysis, forecasting and control*. Prentice-Hall, 3ème edition, 1994.

[Brighton and Mellish, 2002] H. Brighton and C. Mellish. Advances in instance selection for instance-based learning algorithms. *Data mining and knowledge discovery*, 6(2) :153–172, 2002.

[Cadez *et al.*, 2000] I.V. Cadez, S. Gaffney, and P. Smyth. A general probabilistic framework for clustering individuals and objects. In *Knowledge discovery and data mining*, pages 140–149, 2000.

[Cameron-Jones, 1995] R.M. Cameron-Jones. Instance selection by encoding length heuristic with random mutation hill climbing. In *Proceedings of the eighth australian joint conference on artificial intelligence*, pages 99–106, 1995.

[Cannon *et al.*, 2002] A. Cannon, J. Ettinger, D. Hush, and C. Scovel. Machine learning with data dependent hypothesis classes. *Journal of machine learning research*, 2 :335–358, 2002.

[Chan and Fu, 1999] K.P. Chan and A.W.C. Fu. Efficient time series matching by wavelets. In *ICDE*, pages 126–133, 1999.

[Chang, 1991] C.L. Chang. Finding prototypes for nearest neighbor classifiers. *IEEE Transactions on Computers*, 23(11) :1179–1184, Novembre 1991.

[Chapman *et al.*, 2000] P. Chapman, J. Clinton, R. Kerber, T. Khabaza, T. Reinartz, C. Shearer, and R. Wirth. *CRISP-DM 1.0 : step-by-step data mining guide*, 2000.

[Conan-Guez, 2003] B. Conan-Guez. *Modélisation supervisée de données fonctionnelles par perceptron muli-couches*. PhD thesis, Université de Paris-Dauphine, 2003.

[Cover and Hart, 1967] T.M. Cover and P.E. Hart. Nearest neighbor pattern classification. *Institute of electrical and electronics engineers transactions on information theory*, 13 :21–27, 1967.

[Das *et al.*, 1998] G. Das, K.I. Lin, H. Mannila, G. Renganathan, and P. Smyth. Rule discovery from time series. In *Knowledge Discovery and Data Mining*, pages 16–22, 1998.

[Devroye *et al.*, 1996] L. Devroye, L. Györfi, and G. Lugosi. *A probabilistic theory of pattern recognition*. Springer, 1996.

[Dougherty *et al.*, 1995] J. Dougherty, R. Kohavi, and M. Sahami. Supervised and unsupervised discretization of continuous features. In *International conference on machine learning*, pages 194–202, 1995.

[Droesbeke *et al.*, 2002] J.-J. Droesbeke, J. Fine, and G. Saporta. *Méthodes bayésiennes en statistiques*. Technip, France, 2002.

[Fayyad *et al.*, 1996] U.M. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. Knowledge discovery and data mining : towards a unifying framework. In *KDD*, pages 82–88, 1996.

[Ferrandiz and Boullé, 2004] S. Ferrandiz and M. Boullé. Utilisation des graphes de proximité dans le cadre de l'apprentissage basé sur les voisins. In D.A. Zighed and G. Venturini, editors, *Actes des 4èmes journées francophones extraction et gestion des connaissances (EGC'04)*, pages 355–366. Cépaduès-Editions, 2004.

[Ferrandiz and Boullé, 2005a] S. Ferrandiz and M. Boullé. Multivariate discretization by recursive supervised bipartition of graph. In P. Perner and A. Imiya, editors, *Machine learning and data mining in pattern recognition*, pages 253–264. Springer, 2005.

[Ferrandiz and Boullé, 2005b] S. Ferrandiz and M. Boullé. Supervised evaluation of dataset partitions : advantages and practice. In P. Perner and A. Imiya, editors, *Machine learning and data mining in pattern recognition*, pages 600–609. Springer, 2005.

[Ferrandiz and Boullé, 2006a] S. Ferrandiz and M. Boullé. Efficient instance selection for the nearest neighbor rule. *Machine learning*, 2006. en cours de relecture.

[Ferrandiz and Boullé, 2006b] S. Ferrandiz and M. Boullé. Illustration d'une méthode d'évaluation supervisée par un problème de classification de courbes. In *Actes des 13èmes rencontres de la société francophone de classification*, 2006.

[Ferrandiz and Boullé, 2006c] S. Ferrandiz and M. Boullé. Sélection supervisée d'instances : une approche descriptive. In D.A. Zighed and G. Venturini, editors, *Actes des 6èmes journées francophones extraction et gestion des connaissances (EGC'06)*, pages 421–432. Cépaduès-Editions, 2006.

[Ferrandiz and Boullé, 2006d] S. Ferrandiz and M. Boullé. Supervised evaluation of voronoi partitions. *Journal of intelligent data analysis*, 10(3) :269–284, 2006.

[Ferrandiz and Boullé, 2006e] S. Ferrandiz and M. Boullé. Supervised selection of dynamic features, with an application to telecommunication data preparation. In P. Perner and A. Ahlemeyer-Stubbe, editors, *Proceedings of the* 6^{th} *industrial conference on data mining*. Springer Verlag, 2006.

[Fisher, 1936] R.A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of eugenics*, 7 :179–188, 1936.

[Fix and Hodges, 1951] E. Fix and J. Hodges. Discriminatory analysis. nonparametric discrimination : Consistency properties. *Technical Report 4, Project Number 21-49-004, USAF School of Aviation Medicine, Randolph Field, TX*, 1951.

[Foata and Fuchs, 2003] D. Foata and A. Fuchs. *Calcul des probabilités*. Dunod, 2003.

[Gammerman and Vovk, 1999] A. Gammerman and V. Vovk. Special issue on Kolmogorov complexity. *Computer journal*, 42(4), 1999.

[Gates, 1972] G.W. Gates. The reduced nearest neighbor rule. *IEEE transactions on information theory*, 18(3) :431–433, 1972.

[Gavrilov *et al.*, 2000] M. Gavrilov, D. Anguelov, P. Indyk, and R. Motwani. Mining the stock market : which measure is best ? In *Knowledge discovery in databases*, pages 487–496, 2000.

[Ge and Smyth, 2000] X. Ge and P. Smyth. Deformable markov model templates for time-series pattern matching. In *Knowledge discovery and data mining*, pages 81–90, 2000.

[Giles *et al.*, 2001] C.L. Giles, S. Lawrence, and A.C. Tsoi. Noisy time series prediction using a recurrent neural network and grammatical inference. *Machine Learning*, 44(1/2) :161–183, 2001.

[Gouriéroux and Monfort, 1995] C. Gouriéroux and A. Monfort. *Séries temporelles et modèles dynamiques*. Economica, 1995.

[Grünwald *et al.*, 2005] P.D. Grünwald, I.J. Myung, and M.A. Pitt. *Advances in minimum description length : theory and applications*. MIT Press, 2005.

[Guralnik and Srivastava, 1999] V. Guralnik and J. Srivastava. Event detection from time series data. In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 33–42. ACM Press, 1999.

[Guyon and Elisseeff, 2003] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of machine learning research*, 3 :1157–1182, 2003.

[Guyon *et al.*, 2006] I. Guyon, S. Gunn, M. Nikravesh, and L.A. Zadeh. *Features extraction : foundations and applications*. Springer-Verlag, 2006.

[Hand, 1998] D.J. Hand. Data mining : statistics and more ? *The American Statistician*, 52(2) :112–118, 1998.

[Hansen and Mladenovic, 2001] P. Hansen and N. Mladenovic. Variable neighborhood search : principles and applications. *European Journal of Operational Research*, 130 :449–467, 2001.

[Hansen and Yu, 2001] M.H. Hansen and B. Yu. Model selection and the principle of minimum description length. *J. American Statistical Association*, 96 :746–774, 2001.

[Hart, 1968] P.E. Hart. The condensed nearest neighbor rule. *IEEE transactions on information theory*, 14 :515–516, 1968.

[Hastie *et al.*, 2001] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning*. Springer, 2001.

[Hugueney, 2003] B. Hugueney. *Représentation symbolique de longues séries temporelles*. PhD thesis, Université Paris 6, 2003.

[James, 2002] G.M. James. Generalized linear models with functional predictors. *Journal of the royal statistical society, Series B*, 64(3) :411–432, 2002.

[Jaromczyk and Toussaint, 1992] J.W. Jaromczyk and G.T. Toussaint. Relative neighborhood graphs and their relatives. *Proceedings IEEE*, 80(9) :1502–1517, 1992.

[Kaddous and Sammut, 2005] M.W. Kaddous and C. Sammut. Classification of multivariate time series and structured data using constructive induction. *Machine learning*, 58(2-3) :179–216, 2005.

[Keogh and Pazzani, 1998] E. Keogh and M. Pazzani. An enhanced representation of time series which allows fast and accurate classification, clustering and relevance feedback. In R. Agrawal, P. Stolorz, and G. Piatetsky-Shapiro, editors, *Fourth international conference on knowledge discovery and data mining (KDD'98)*, pages 239–241, New York City, NY, 1998. ACM Press.

[Keogh and Pazzani, 1999] E. Keogh and M. Pazzani. Scaling up dynamic time warping to massive datasets. In J.M. Zytkow and J. Rauch, editors, *3rd European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD'99)*, volume 1704, pages 1–11, Prague, Czech Republic, 1999. Springer.

[Keogh and Smyth, 1997] E. Keogh and P. Smyth. A probabilistic approach to fast pattern matching in time series databases. In D. Heckerman, H. Mannila, D. Pregibon, and R. Uthurusamy, editors, *Third international conference on knowledge discovery and data mining*, pages 24–30, Newport Beach, CA, USA, 1997. AAAI Press, Menlo Park, California.

[Kohavi and John, 1997] R. Kohavi and G. John. Wrappers for feature selection. *Artificial intelligence*, 97(1-2) :273–324, Décembre 1997.

[Kohavi and Sahami, 1996] R. Kohavi and M. Sahami. Error-based and entropy-based discretization of continuous features. In *KDD*, pages 114–119, 1996.

[Kohonen, 2001] T. Kohonen. *Self-organizing maps*. Springer, third edition edition, 2001.

[Lanterman, 2001] A.D. Lanterman. Schwarz, Wallace, and Rissanen : intertwining themes in theories of model selection. *International statistical review*, 69 :185–212, 2001.

[Li and Vitanyi, 1997] M. Li and P.M.B. Vitanyi. *An introduction to Kolmogorov complexity and its applications*. Springer-Verlag, Berlin, 1997.

[Li, 2000] C. Li. *A bayesian approach to temporal data clustering using the hidden markov model methodology*. PhD thesis, Vanderbilt University, 2000.

[MacQueen, 1967] J. MacQueen. Some methods for classification and analysis of multivariate observations. In Le Cam and Neyman, editors, *Fifth Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297, 1967.

[Oliver and Baxter, 1994] J.J. Oliver and R. Baxter. MML and bayesianism : similarities and differences (introduction to minimum encoding inference - part ii). Technical Report 206, Department of computer science, Monash university, 1994.

[Oliver and Hand, 1994] J.J. Oliver and D.J. Hand. Introduction to minimum encoding inference. Technical Report 205, Department of computer science, Monash university, 1994.

[Pavlidis, 1974] T. Pavlidis. Waveform segmentation through functional approximation. *IEEE Trans. Comp.*, C-22(7) :689–697, 1974.

[Preparata and Shamos, 1986] F.P. Preparata and M.I. Shamos. *Computational geometry : an introduction*. Springer, 1986.

[Qu *et al.*, 1998] Y. Qu, C. Wang, and X.S. Wang. Supporting fast search in time series for movement patterns in multiple scales. In *CIKM*, pages 251–258, 1998.

[Quinlan and Rivest, 1989] J.R. Quinlan and R.L. Rivest. Inferring decision trees using the minimum description length principle. *Inf. Comput.*, 80(3) :227–248, 1989.

[Rabiner, 1989] L.R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2) :257–285, 1989.

[Ramsay and Silverman, 1997] J. Ramsay and B. Silverman. *Functional data analysis*. Springer-Verlag, 1997.

[Rissanen, 1978] J. Rissanen. Modeling by shortest data description. *Automatica*, 14 :465–471, 1978.

[Rissanen, 1989] J. Rissanen. *Stochastic complexity in statistical inquiry*, volume 15. World Scientific Publishing, River Edge, 1989.

[Rossi *et al.*, 2004] Fabrice Rossi, Brieuc Conan-Guez, and A. El Golli. Clustering functional data with the SOM algorithm. In *Proceedings of the ESANN*, pages 305–312, Avril 2004.

[Salzberg, 1991] S. Salzberg. A nearest hyperrectangle learning method. *Machine Learning*, 6 :277–309, 1991.

[Saporta, 1990] G. Saporta. *Probabilités et analyse des données statistiques*. Technip, 1990.

[Scholkopf and Smola, 2001] B. Scholkopf and A.J. Smola. *Learning with kernels : support vector machines, regularization, optimization, and beyond*. MIT Press, Cambridge, MA, USA, 2001.

[Schwartz, 1978] G. Schwartz. Estimating the dimension of a model. *Annals of statistics*, 6(2) :461–464, 1978.

[Sebban *et al.*, 2002] M. Sebban, R. Nock, and S. Lallich. Stopping criterion for boosting-based data reduction techniques : from binary to multiclass problem. *Journal of machine learning research*, 3 :863–885, 2002.

[Shannon, 1948] C.E. Shannon. A mathematical theory of communication. Technical report, Bell systems technical journal, 1948.

[Siebes *et al.*, 2006] A. Siebes, J. Vreeken, and M. van Leeuwen. Item sets that compress. In *SIAM Conference on data mining*, pages 393–404, 2006.

[Sinkkonen *et al.*, 2002] J. Sinkkonen, S. Kaski, and J. Nikkilä. Discriminative clustering : optimal contingency tables by learning metrics. In *Proceedings of the* 13^{th} *european conference on machine learning*, pages 418–430, 2002.

[Slonim and Tishby, 2000] N. Slonim and N. Tishby. Agglomerative information bottleneck. In *Proceedings of neural information processing system*, volume 12, pages 317–623, 2000.

[Smyth, 1997] P. Smyth. Clustering sequences with hidden markov models. In M.C. Mozer, M.I. Jordan, and T. Petsche, editors, *Advances in neural information processing systems*, volume 9, pages 648–654. The MIT Press, 1997.

[Smyth, 1999] P. Smyth. Probabilistic model-based clustering of multivariate and sequential data. *Proceedings of artificial intelligence and statistics*, pages 299–304, 1999.

[Solomonoff, 1964] R. Solomonoff. A formal theory of inductive inference, I and II. *Information and control*, 7 :1–22 and 224–254, 1964.

[Tikhonov, 1963] A.N. Tikhonov. On solving ill-posed problem and method of regularization. *Doklady Akademii Nauk USSR*, 153 :501–504, 1963.

[Uderzo and Goscinny, 1990] A. Uderzo and R. Goscinny. *Le grand fossé*. Editions Albert René, 1990.

[Vapnik, 1996] V. Vapnik. *The nature of statistical learning theory*. Springer-Verlag, New-York, 1996.

[Vitanyi and Li, 2000] P.M.B. Vitanyi and M. Li. Minimum description length induction, bayesianism, and Kolmogorov complexity. *IEEE Transactions on information theory*, 46 :446–464, 2000.

[Wallace and Boulton, 1968] C.S. Wallace and D.M. Boulton. An information measure for classification. *Computer Journal*, 11(2) :185–194, 1968.

[Wettschereck and Dietterich, 1995] D. Wettschereck and T.G. Dietterich. An experimental comparison of the nearest neighbor and nearest hyperrectangle algorithms. *Machine Learning*, 19(1) :5–27, 1995.

[Wilson and Martinez, 1997a] D.R. Wilson and T.R. Martinez. Improved heterogeneous distance functions. *Journal of artificial intelligence research*, 6(1) :1–34, 1997.

[Wilson and Martinez, 1997b] D.R. Wilson and T.R. Martinez. Instance pruning techniques. In D. Fisher, editor, *Proceedings of the* 14^{th} *international conference on machine learning*, pages 403–411, San Francisco, 1997. Morgan Kaufmann.

[Wilson, 1972] D.L. Wilson. Asymptotic properties of nearest neighbor rules using edited data. *IEEE Transactions on systems, man and cybernetics*, 2 :408–421, 1972.

[Yamanishi, 1998] K. Yamanishi. A decision-theoretic extension of stochastic complexity and its applications to learning. *IEEE Transactions on Information Theory*, 44(4) :1424–1439, 1998.

[Yi *et al.*, 1998] B.K. Yi, H.V. Jagadish, and C. Faloutsos. Efficient retrieval of similar time sequences under time warping. In *ICDE*, pages 201–208, 1998.

Résumé. – En phase de préparation d'un processus de fouille de données, une part importante du travail est consacrée à la construction et à la sélection des variables descriptives. L'approche filtre univariée usuellement adoptée nécessite l'emploi d'une méthode d'évaluation d'une variable. Nous considérons la question de l'évaluation supervisée d'une variable séquentielle. Pour résoudre ce problème, nous montrons qu'il suffit de résoudre un problème plus général : celui de l'évaluation supervisée d'une mesure de similitude.

Nous proposons une telle méthode d'évaluation. Pour l'obtenir, nous formulons le problème en un problème de recherche d'une partition de Voronoi informative. Nous proposons un nouveau critère d'évaluation supervisée de ces partitions et une nouvelle heuristique de recherche optimisée. Le critère prévient automatiquement le risque de surapprentissage et l'heuristique trouve rapidement une bonne solution. Au final, la méthode réalise une estimation non paramétrique robuste de la densité d'une variable cible catégorielle conditionnellement à une mesure de similitude définie à partir d'une variable descriptive.

La méthode a été testée sur de nombreux jeux de données. Son utilisation permet de répondre à des questions comme : quel jour de la semaine ou quelle tranche horaire sur la semaine discrimine le mieux le segment auquel appartient un foyer à partir de sa consommation téléphonique fixe ? Quelle série de mesures permet de quantifier au mieux l'appétence à un nouveau service ?
Mots clés. – Apprentissage, Exploration de données, Statistique bayésienne, Analyse discriminante.

Title

Supervised learning from sequential data.

Abstract. – In the data mining process, the main part of the data preparation step is devoted to feature construction and selection. The filter approach usually adopted requires evaluation methods for any kind of feature. We address the problem of the supervised evaluation of a sequential feature. We show that this problem is solved if a more general problem is tackled : that of the supervised evaluation of a similarity measure.

We provide such an evaluation method. We first turn the problem into the search of a discriminating Voronoi partition. Then, we define a new supervised criterion evaluating such partitions and design a new optimised algorithm. The criterion automatically prevents from overfitting the data and the algorithm quickly provides a good solution. In the end, the method can be interpreted as a robust non parametric method for estimating the conditional density of a categorical target feature given a similarity measure defined from a descriptive feature.

The method is experimented on many datasets. It is useful for answering questions like : which day of the week or which hourly time segment is the most relevant to discriminate customers from their call detailed records ? Which series allows to better estimate the customer need for a new service ?

Discipline : Sciences et Technologies de l'Information

Laboratoire : Groupe de Recherche en Informatique, Image, Automatique et Instrumentation de Caen (UMR 6072), Université de Caen Basse-Normandie, France.