Table des matières

Liste des tableaux

Liste des figures

À la mémoire de Thérèse Bergeron,
Jean-Marie Brochu et Roger Hudon.

Remerciements

Premièrement, je veux remercier les Professeurs Jacques Corbeil et François Laviolette de m'avoir donné la chance de travailler dans leurs laboratoires et aidé dans mon développement scientifique et professionnel.

Je remercie aussi les partenaires qui ont travaillé avec nos laboratoires pour le projet décrit en ce mémoire, soit Héma-Québec, Phytronix et Water Corporation. Leur collaboration fut d'une aide inestimable dans nos travaux.

Je tiens aussi à remercier les autres étudiants du laboratoire Corbeil qui ont également travaillé sur ce projet, soit Pier-Luc Plante, Frédéric Raymond et Maxime Déraspe. Un merci spécial aussi à Lynda Robitaille, dont les efforts permettent au laboratoire de bien fonctionner. Je remercie aussi les étudiants du GRAAL, groupe de recherche des professeurs Laviolette et Marchand, qui m'ont aidé dans mon apprentissage au domaine de l'apprentissage automatique. Alors merci à Alexandre Drouin, Pascal Germain, Jean-Francis Roy, Amélie Rolland, Hana Ajakan, Prudencio Tossou et Mazid Osseni. Merci aussi aux stagiaires qui ont contribué aux travaux, surtout au niveau du noyau à boîtes chevauchantes, Louis Fortier-Dubois et Louis-Émile Robitaille.

Finalement, je veux remercier les organismes subventionnaires, soit le Conseil de Recherches en Sciences Naturelles et en Génie (CRSNG) le Fonds de Recherche du Québec - nature et Technologies (FRQNT), l'organisme MITACS et la faculté de Médecine de l'université Laval.

Introduction

De nos jours, l'informatique prend de plus en plus de place dans la vie quotidienne. L'informatique prend une place de plus en plus grande dans la plupart des champs de recherche en particulier. Un de ces champs où il y a eu de grandes avancées est la bio-informatique, où l'informatique est appliquée à la recherche en biologie. La bio-informatique a principalement pris sa place en recherche dans les domaines de l'étude des protéines et leurs structures, de la biologie des systèmes, des sciences "omiques" (génomique, transcriptomique, métagénomique, protéomique), etc.. Une de ces sciences "omique" est la métabolomique, qui consiste en l'étude de l'ensemble des métabolites d'un individu ou échantillon. Mesurer le niveau d'un grand nombre de métabolites est donc aussi un exercice très intéressant, permettant d'établir des liens entre différents aspects du phénotype exprimé par l'individu source, voire établir des liens entre le métabolisme et la génétique d'un individu. Guo et collab. (2015) Une des méthodes les plus répandues afin de mesurer les métabolites dans un échantillon donné est la spectrométrie de masse. Les multiples aspects de ce projet seront introduits séparément ci-dessous.

0.1 Spectrométrie de masse et métabolomique

La spectrométrie de masse est une méthode de mesure bien établie et connue en biologie. Le principe de base de cette technologie est qu'un échantillon est exposé à une source d'ionisation afin d'en analyser les molécules composantes. La première étape de l'acquisition du spectre de masse d'un échantillon donné est l'ionisation. Les sources varient selon l'application, mais le principe reste le même. Les molécules de l'échantillon sont exposées à cette source, qui va ioniser les molécules entières et parfois causer la fragmentation des molécules et permettre l'ionisation de ses fragments. Les ions ainsi formés sont ensuite transportés par un champ électrique. On pratique ensuite une sélection des ions. Ce procédé est généralement fait par des quadripôles électrostatiques qui vont éjecter tous les ions circulant à travers le champ magnétique qui ont un ratio de masse sur charge (m/z) inférieur ou supérieur à la fenêtre de masses désirée. Le quadripôle va aussi sélectionner si l'on mesure seulement les ions positivement chargés ou négativement chargés. Les ions restants sont ensuite acheminés vers le détecteur. Le détecteur enregistre ensuite les ions qui frappent sa surface. Cela permet de mesurer le ratio de masse sur charge ainsi que le nombre d'ions frappant le détecteur. Un schéma du fonctionnement interne de base d'un spectromètre de masse est présenté à la figure 0.1.

La spectrométrie de masse est utilisée dans une grande variété d'applications dans les domaines scien-

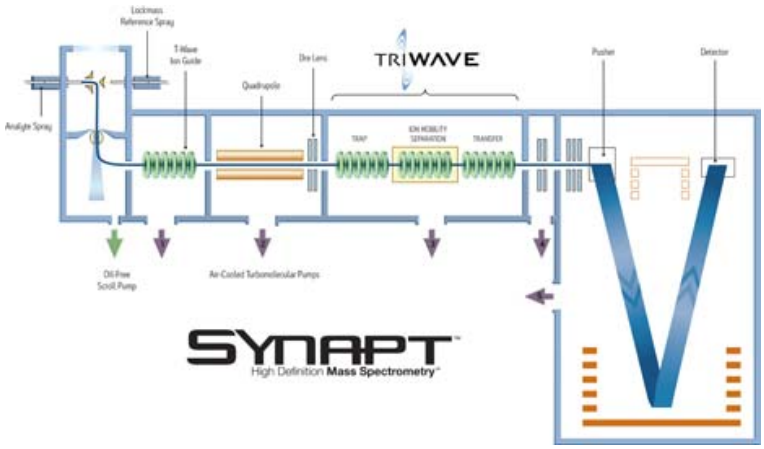

FIGURE 0.1 – **Schéma du fonctionnement interne d'un spectromètre de masse Synapt-G2 Si de Waters Corporation**

tifiques. Entre autres, on retrouve des applications telles que la datation au carbone 14, le contrôle de qualité de l'eau en vérifiant la présence ou l'absence de molécules contaminantes et plusieurs applications en biologie. Ces applications biologiques peuvent être cliniques, telles que la détection de drogue dans un échantillon sanguin ou d'urine ou bien à détecter des biomarqueurs spécifiques. D'autres applications sont utilisées en recherche. Parmi ses applications, notons plusieurs utilisations en protéomique qui permettent de vérifier l'interaction entre une protéine et d'autres molécules, l'identification de protéines par la masse de leurs fragments ou bien le suivi de réactions enzymatiques.

Dans le projet décrit ici, l'instrument utilisé pour faire l'acquisition des spectres de masse est le Synapt G2 Si de Waters Corporation. Le mode d'acquisition pour les expériences décrites dans ce mémoire était en mode temps de vol (*Time of Flight*, TOF). Ce mode signifie que le ratio masse sur charge des ions est calculé à partir de leur temps de vol dans l'instrument. Ce calcul est fait selon le principe que le champ électrique dans l'instrument est constant et le même pour tous les ions. Ainsi, plus une molécule est légère, plus elle va voyager rapidement à travers le champ. Aussi, plus une molécule est chargée, plus elle va voyager rapidement. Ainsi, le ratio de masse sur charge est proportionnel au temps que l'ion prend pour parcourir la distance connue de l'instrument.

En général, la spectrométrie de masse vise à identifier la présence, et si possible la quantité, d'un métabolite connu en analysant l'intensité d'un pic qui lui est associé dans le spectre. La technique du MS/MS consiste à sélectionner un delta masse, en uma ou en Dalton, autour d'un pic associé au métabolite d'intérêt, et ensuite de fragmenter une seconde fois cette molécule afin de pouvoir observer les différents fragments de la molécule d'intérêt. Ces fragments se nomment les ions filles. Cette technique aide grandement à identifier dont on connait le pic associé dans le spectre de masse. Au contraire, l'approche utilisée dans ce projet est beaucoup plus large bande, c'est-à-dire que notre objectif ne sera pas d'identifier la présence ou l'absence d'une molécule particulière, mais de chercher des signatures dans le spectre associé à un échantillon prélevé chez un individu. Ces signatures

permettraient d'identifier la présence ou non d'une pathologie ou de toute autre forme de condition.

La notion de résolution d'un spectromètre de masse entre également en compte. L'instrument utilisé au cours des travaux présentés dans ce mémoire est un instrument à haute résolution, et a la possibilité d'être utilisé en mode *sensitivity*, *resolution* et *high resolution*. Une résolution signifie que le spectromètre peut détecter un pic relié à une molécule plus précisément sur l'axe m/z. Cela peut être évalué par la mesure de *Full Width at Half Maximum* (FWHM), qui consiste à évaluer la largeur d'un pic à la moitié de son intensité maximale. On recherche donc une valeur maximale de résolution (donc un FWHM élevé) afin de résoudre les pics le plus précisément possible. Par contre, en spectrométrie de masse, il faut considérer que plus la résolution d'un spectromètre est élevée, plus sa sensibilité, c'est-à-dire sa capacité à résoudre des pics de faible intensité, sera faible.

Les travaux présentés ici traitent principalement d'échantillons de produits sanguins. Le sang est formé principalement de deux parties : soit la partie cellulaire, contenant les plaquettes, les globules rouges (érythrocytes) et les globules blancs ; et le plasma, qui est la partie liquide du sang. Le plasma forme environ 50% du sang par volume. La séparation du sang en plasma et en contenu cellulaire se fait avec une simple centrifugation. Le plasma est riche en plusieurs types de molécules. On y retrouve plusieurs types de protéines et de peptides, des nutriments et de nombreuses autres petites molécules en solution. Une étude sur le métabolome du sang place le nombre de métabolites différents à environ 4200 molécules différentes Psychogios et collab. (2011).

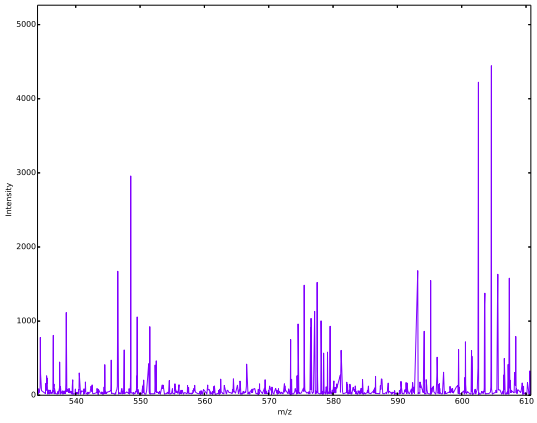

FIGURE 0.2 – **Exemple de spectre de masse**

La figure 0.2 montre un exemple d'un spectre de masse acquis lors des travaux présentés dans ce mémoire. L'abscisse est ici l'axe du ratio de masse sur charge m/z, en unités arbitraires de masses (uma) ou Daltons (Da). L'ordonnée est l'axe correspondant à l'intensité des pics, en unités arbitraires.

0.2 Apprentissage automatique

Dans le projet présent, la recherche de signatures dans les spectres se fera par apprentissage automatique. Le chercheur Arthur Samuel a défini ce domaine, en 1959, comme le champ d'études donnant aux ordinateurs la capacité d'apprendre sans être explicitement programmés. En termes simples, cela signifie que c'est un champ de recherche dont l'objectif est de rendre un ordinateur capable d'apprendre à effectuer une tâche en ne lui fournissant que des exemples pour qu'il puisse s'entraîner. En contraste, prenons l'exemple d'un programme qui accomplit une tâche en étant explicitement programmé. Pour faire un tel programme, il faudrait une connaissance avancée de la tâche à accomplir et généralement l'intervention d'un expert. Si l'on considère l'exemple d'un programme dont la tâche est de faire le diagnostic d'un patient dans un hôpital, il faudrait certainement l'intervention de médecins et experts de la santé. De plus, le programme aurait une structure rigide et inflexible. Au contraire, l'application de l'apprentissage automatique pour faire cette même tâche consisterait à montrer des centaines, voire milliers, de dossiers médicaux à un algorithme qui pourrait ensuite prédire le diagnostique. Cette approche a aussi l'avantage d'être plus flexible et facilement modifiable dans le temps, puisqu'on a qu'à montrer de nouveaux exemples au programme.

Il existe plusieurs branches à l'apprentissage automatique. Celle qui concerne le présent projet est la branche de l'apprentissage supervisé. Dans ce cas, les exemples disponibles sont séparés en au moins deux classes. Chacune de ces classes est représentée par une *étiquette*. Chaque exemple est décrit par une série d'informations, qui peuvent être quantitatives ou qualitatives, que l'on nomme les *caractéristiques* des exemples. On fournit ensuite à l'algorithme une liste de tels exemples ainsi que leurs étiquettes respectives afin qu'il puisse apprendre en considérant les caractéristiques d'exemples à prédire les étiquettes. L'objectif est que le prédicteur appris soit non seulement performant sur les exemples de la liste fournie, mais surtout sur tout exemple non étiqueté.

Le type d'apprentissage utilisé au courant de ce projet est exclusivement de l'apprentissage inductif. Dans ce type d'apprentissage, on suppose l'existence d'une distribution de probabilité (inconnue) qui est la source des exemples utilisés pour l'entraînement et qui sera la source des exemples à étiqueter à l'avenir. L'existence de cette distribution garantie que tous les exemples sont pigés indépendamment et sont identiquement distribués (*i.i.d.*). On applique ensuite l'algorithme à l'ensemble d'apprentissage. De cette manière, on obtient un *prédicteur*, également appelé classificateur. Ce classificateur peut alors être utilisé sur de nouveaux exemples inconnus pour déterminer leurs classes. Si ces exemples viennent de la distribution originale, la théorie de l'apprentissage automatique nous donne des garanties quant à la performance du classificateur appris sur ces nouveaux exemples non étiquetés.

En effet, un élément important de l'apprentissage automatique est qu'il faut des garanties de généralisation. Ces garanties sont généralement des bornes statistiques qui encadrent le taux d'erreur sur les exemples à venir du classificateur. On appelle ce taux d'erreur le risque. Il faut calculer ce risque et les garanties qui y sont associés de la bonne manière afin d'avoir un classificateur performant qui soit bien capable de généraliser ce qui a été appris sur toutes autres données provenant de la distribution

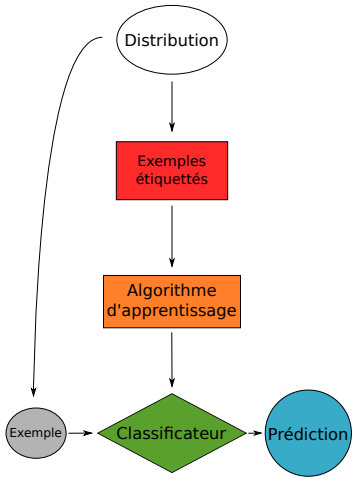

FIGURE 0.3 – **Une représentation graphique du principe de l'apprentissage supervisé par induction.** Une distribution inconnue génère des exemples. On prend un ensemble d'exemples étiquetés que l'on fournit à un algorithme d'apprentissage. On obtient de cette manière un classificateur. Le classificateur peut ensuite classifier des exemples venant de la même distribution.

des exemples. Un effet qu'il est important de contrôler en apprentissage automatique est le surapprentissage (*overfitting* en anglais). Ce phénomène est lorsqu'un classificateur apprend trop des données sur lesquelles il est entraîné et qui a une moins bonne capacité à classifier sur de nouvelles données inconnues.

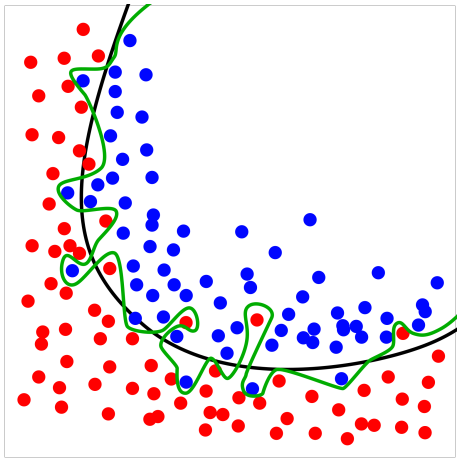

FIGURE 0.4 – **Exemple de surapprentissage**. Les lignes vertes et noires représentent des frontières de décision de différents classificateurs. Le classificateur vert surapprend, alors que le classificateur noir ne le fait pas. Source : Image sous licence Creative Commons. Auteur : Ignacio Icke. Trouvé sur http ://commons.wikimedia.org/

La figure 0.4 démontre un bon exemple de surapprentissage. La ligne verte représente la frontière de décision d'un classificateur qui a surappris les données qui lui ont été montrées. Ce classificateur ne

fera aucune erreur sur les données sur lesquelles il a été entraîné. La ligne noire représente la frontière de décision d'un classificateur qui n'a pas surappris. Alors que ce second classificateur fait quelques erreurs sur les données, présentées ici comme les points rouges et bleus, on peut constater qu'il va très probablement mieux performer sur de nouvelles données. Éviter le surapprentissage aide entre autres à avoir des classificateurs plus résistants au bruit et aux données aberrantes. Plusieurs techniques sont appliquées pour éviter cet effet.

Une technique utilisée pour évaluer si un classificateur donné généralise bien est la séparation de l'ensemble de données. Cette technique consiste à séparer aléatoirement l'ensemble de données en deux ensembles, que l'on nomme *l'ensemble d'entraînement* et *l'ensemble de test*. On utilise l'ensemble d'entraînement pour entraîner le classificateur. Si l'on évalue le risque du classificateur sur l'ensemble d'entraînement, on obtient le *risque empirique*. L'ensemble de test est gardé séparément. Une fois le classificateur entraîné, on lui demande de prédire la classe de chacun des exemples en voyant seulement les caractéristiques de ces exemples, et non les étiquettes. On compare ensuite les prédictions du classificateur avec les vraies étiquettes des exemples de test. On obtient ainsi le *risque de test*, soit la proportion d'erreurs commises par le classificateur par rapport à une estimation non biaisée de la réalité. C'est sur le risque de test que l'on peut calculer des bornes du vrai risque (inconnu) sur la distribution source. Par contre, cette méthode nous permet seulement de constater s'il y a eu surapprentissage (ce qui correspond au cas où le risque empirique est bas et le risque de test est élevé) et non pas l'éviter.

Une méthode couramment utilisée pour contrôler le surapprentissage est la validation croisée (*cross-validation*). Une visualisation de cette méthode est montrée à la figure 0.5. Pour utiliser cette méthode, on sépare aléatoirement l'ensemble d'entraînement en plusieurs sous-ensembles, ou plis (*fold*). La figure 0.5 présente un exemple avec 5 plis. On prend ensuite tous les plis sauf un et l'on considère l'ensemble ainsi obtenu comme nouvel ensemble d'entraînement. On donne ce nouvel ensemble d'entraînement à l'algorithme d'apprentissage et on teste le classificateur qui en résulte sur le dernier pli sur lequel le classificateur n'a pas été entraîné. Ces étapes sont répétées afin de tester chaque pli une fois. On prend finalement la moyenne des risques obtenus sur chacun des plis pour obtenir le risque de validation croisée. Ce risque est un bon indicateur pour savoir si un classificateur entraîné sous certains paramètres va surapprendre ou pas. On peut ainsi choisir quels paramètres semblent optimaux pour l'algorithme et construire un classificateur sur l'ensemble d'entraînement complet et finalement le tester sur l'ensemble de test, qui est resté jusque là et caché à l'algorithme d'apprentissage.

Les paramètres choisis par validation croisée sont les **hyperparamètres** de l'algorithme. Ce sont des paramètres que l'on doit fixer à l'algorithme afin qu'il puisse déterminer ses paramètres internes. Certains sont présents pour agir comme régulation du surapprentissage, alors que d'autres influencent la manière dont l'algorithme va apprendre son modèle. On peut aussi considérer des paramètres sur le traitement des données, tels que des paramètres reliés au prétraitement des spectres de masse, comme des hyperparamètres, car ils vont affecter l'apprentissage sur les données même s'ils ne font pas partie intégrale de l'algorithme.

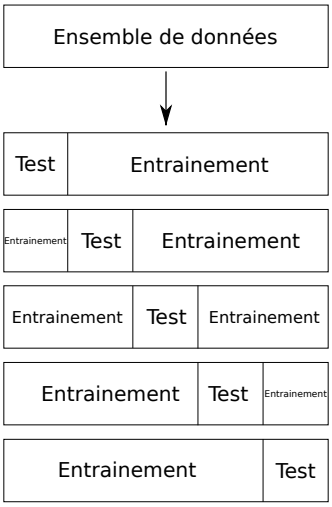

FIGURE 0.5 – **Un exemple de validation croisée**. Dans cet exemple, on fait une validation croisée à 5 plis. On itère sur chacun des 5 ensembles de données, entraînant le classificateur sur les 80% du sous-ensemble réservé à cet effet et l'on teste sur le 20% restant. On aura ainsi utilisé toutes les données pour s'entraîner et pour tester.

0.3 Contexte des travaux et hypothèses

Les travaux décrits ici ont été faits en collaboration avec Héma-Québec. Héma-Québec est un organisme sans but lucratif qui a l'objectif de fournir aux besoins du Québec en termes de produits sanguins, de tissus humains, de sang de cordon, de lait maternel et de produits cellulaires. Héma-Québec, dans l'année 2014-2015, a reçu des dons d'environ 325 000 donneurs et a livré environ 484 850 produits sanguins labiles aux centres hospitaliers de la province. Ces produits sanguins sont des culots globulaires, des plaquettes et du plasma. Ces produits sont générés à partir du sang donné par fractionnement. Héma-Québec (2015)

Un don de sang total permet de séparer trois produits sanguins importants. Une première centrifugation du sang total fait précipiter les globules rouges. Le surnageant produit à cette étape est constitué de plasma avec globules blancs et plaquettes en suspension. Le plasma et les globules rouges sont récupérés séparément. Puis, une solution nourricière est ajoutée aux globules rouges. Le produit en résultant est le culot globulaire. Le plasma est ensuite filtré afin d'en enlever les globules blancs. Cette étape est la déleucocytation et elle est pratiquée afin d'éviter des réactions immunitaires lors de la transfusion. Une seconde centrifugation est alors faite sur le plasma. Cette centrifugation va permettre de séparer les plaquettes et le plasma.

Héma-Québec rapporte 34 cas d'accidents reliés aux produits sanguins en 2014-2015. Une grande partie de la mission d'Héma-Québec consiste à s'assurer de la qualité des produits sanguins fournis aux centres hospitaliers et éviter de permettre à du sang contaminé d'être livré. Plusieurs dispositifs sont en place à cet effet. Il y a premièrement le questionnaire adressé aux donneurs, visant à trier les

donneurs afin d'éviter ceux à risque. De plus, des tests diagnostiques de certaines pathologies sont effectués sur les dons. Héma-Québec rapporte plus d'un million de ces tests par année. Si un de ces tests est positif, le donneur est avisé et le don est détruit. Les pathogènes ou maladies testées sont le cytomégalovirus, l'hépatite B, l'hépatite C, la syphilis, le virus du Nil oriental, le virus HTLV et le VIH. De plus, des normes strictes sont appliquées sur la conservation des produits sanguins traités. Héma-Québec (2015)

Le projet décrit ici a comme objectif de mettre au point une nouvelle méthode de contrôle de qualité qui pourrait être ajoutée au fonctionnement d'Héma-Québec. L'objectif principal est donc de fournir un classificateur informatique entraîné sur des données de spectrométrie de masse permettant de faire la différence entre des échantillons de produits sanguins dégradés, ou périmés, et des échantillons de produits sanguins frais et pouvant être utilisés pour des transfusions. Un objectif secondaire est que ce classificateur comporte un modèle interprétable et parcimonieux afin de pouvoir identifier des biomarqueurs potentiels à partir des pics (et donc molécules) utilisés pour la classification. Une telle méthode comporterait de nombreux avantages au niveau du coût et du temps, grâce à l'utilisation de technologies à haut débit de spectrométrie de masse, tel que la technologie LDTD dont nous parlerons plus loin.

Il existe déjà dans la littérature plusieurs exemples d'application d'apprentissage automatique à la métabolomique en vue de faire du diagnostic. Une revue de littérature par Swan *et al* démontre que l'apprentissage automatique et la spectrométrie de masse sont deux champs de recherche qui peuvent être compatibles en plusieurs applications Swan et collab. (2013). Par contre, cette revue est plutôt générale et considère plus la spectrométrie de masse appliquée à la protéomique.

On retrouve aussi plusieurs articles concernant le diagnostic à partir de spectres de masse acquis sur des produits sanguins. Les travaux de Ge *et al* sont un exemple où l'apprentissage automatique fut appliqué aux spectres de masse d'échantillons de sérum sanguin afin de faire le diagnostic du cancer du pancréas Ge et Wong (2008). De plus, les travaux de Ge *et al* favorisaient l'algorithme de l'arbre décisionnel afin d'avoir des modèles interprétables et pouvoir détecter des biomarqueurs.

Un article de Shin *et al* amène plus de support, faisant le diagnostic de cancer à partir de spectres de masse de plasma sanguin Shin et Markey (2006). L'approche d'apprentissage automatique utilisée dans ces travaux est par contre non parcimonieuse. On note aussi que ce projet utilise le MALDI-TOF (*matrix-assisted laser desorption/ionisation - time of flight*) comme technique de spectrométrie de masse. Ce sont donc principalement les protéines qui sont ciblées dans ces travaux.

On a aussi un article de Zang *et al* concernant des spectres de masse de sérum sanguin et la détection du cancer de la prostate Zang et collab. (2014). Un élément intéressant de ces travaux est que les spectres ont été acquis par LC-MS (*liquid chromatography mass spectrometry*), ce qui en fait donc une expérience de métabolomique au lieu d'une expérience de seule protéomique. Cet article établit donc un précédent qu'il est possible de faire du diagnostic à partir de données métaboliques acquises en spectrométrie de masse.

Un autre article apportant du support à l'approche que nous voulons mettre de l'avant dans ce projet est celui de West *et al* West et collab. (2014). Les travaux décrits dans cet article concernent le diagnostic à partir d'échantillons de plasma. Les spectres furent acquis en plusieurs conditions en LC-MS et GC-MS (*Gas Chromatrography Mass Spectrometry*) afin d'avoir une grande couverture des petits métabolites. Une machine à vecteur de support (voir chapitre 2 pour plus de détails) fut utilisée pour construire un classificateur, qui sera donc non-parcimonieux. Le diagnostic voulu dans cette étude était un diagnostic de désordres sur le spectre de l'autisme à partir des métabolites.

Ces multiples articles supportent l'hypothèse qu'il est possible d'appliquer l'apprentissage automatique aux données de spectrométrie de masse. Il est à noter que la plupart des articles mentionnés utilisaient des méthodes LCMS ou MALDI-TOF. Le projet décrit ici utilise une méthode différente et qui ne permet pas de mesurer les protéines des échantillons. Tout de même, les articles de Zang *et al* et West *et al* supportent qu'il soit possible de faire du diagnostic avec les petits métabolites. L'article de Ge *et al* supporte aussi l'utilisation d'algorithmes parcimonieux, ainsi que la revue de Swan *et al*.

L'approche utilisée dans les travaux décrits dans le présent mémoire diffère en quelques points de ceux décrits dans la revue de littérature et d'avec les travaux plus classiques en spectrométrie de masse. Au lieu d'utiliser une approche très spécifique, ciblant une petite région de ratios de masse sur charge, les spectres de masse sont acquis sur une large bande, soit d'un ratio de masse sur charge de 50 Da jusqu'à 2000 Da. De plus, le projet utilise une technologie récente permettant l'acquisition de spectres à haut-débit. Cette technologie sera décrite en plus de détails au chapitre 1. L'application de cette technologie comporte de nombreux avantages pour le projet. Vu que ce système permet d'acquérir le spectre de masse d'un échantillon en aussi peu que 10 secondes, comparé à des temps d'acquisition de 10 à 30 minutes avec des techniques plus classiques, on peut acquérir un beaucoup plus grand nombre d'échantillons. De plus, cette technologie rend moins cher le coût de l'acquisition de ces spectres. Ces deux avantages font que l'on peut acquérir un très grand nombre d'échantillons en peu de temps et à coûts faibles. Comme l'apprentissage automatique nécessite un grand nombre d'exemples, cette technologie est encore plus avantageuse pour l'approche proposée.

Ce mémoire sera séparé en deux parties distinctes. La première de ces parties concernera les nouvelles méthodes de traitement des données de spectrométrie de masse mise au point au cours du projet. Vu l'approche de ces travaux au problème utilisant de nouvelles technologies et une vision non orthodoxe du problème, plusieurs des méthodes de traitement des données de spectrométrie de masses classiques se sont révélées incompatibles avec les données générées dans ces travaux. Il a donc fallu produire de nouveaux algorithmes afin de traiter les données convenablement et de pouvoir les utiliser pour apprendre.

La seconde partie sera dédiée aux résultats obtenus par nos méthodes d'apprentissage automatique et aux améliorations que nous avons eu à apporter à ces algorithmes. Notamment, l'élaboration d'un nouveau noyau que nous avons inventé afin d'avoir une méthode plus adaptée au travail avec les spectres de masse. Cette seconde partie contiendra aussi des résultats de plusieurs types d'algorithmes

déjà connus, lorsqu'appliqués à des spectres de masse.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

Chapitre 1

Méthodes de traitement des données de spectrométrie de masse

1.1 Introduction

Ce chapitre sera consacré aux travaux effectués sur l'amélioration des méthodes de traitements de données qui ont été faits afin de rendre les données acquises prêtes pour l'apprentissage automatique.

Comme mentionné dans l'introduction, l'utilisation de la spectrométrie de masse dans ce projet n'est pas entièrement orthodoxe. En plus d'une approche de balayage à large bande, une nouvelle technologie permettant l'analyse à haut débit est utilisée dans ce projet.

Cette nouvelle technologie se trouve à être la *Laser Diode Thermal Desorption* (LDTD) de la compagnie Phytronix. Une source LDTD utilise un laser à forte puissance afin de désorber un échantillon d'une matrice, soit une plaque LazWell. Un échantillon est placé dans un puit sur la plaque LazWell. Par la suite, le laser est envoyé sur le dos du puits, où il va chauffer l'échantillon et vaporiser les molécules. Les molécules vaporisées sont ensuite amenées à une chambre d'ionisation chimique à pression atmosphérique (APCI) pour permettre l'ionisation des molécules de l'échantillon. Finalement, les ions sont acheminés vers le spectromètre de masse.

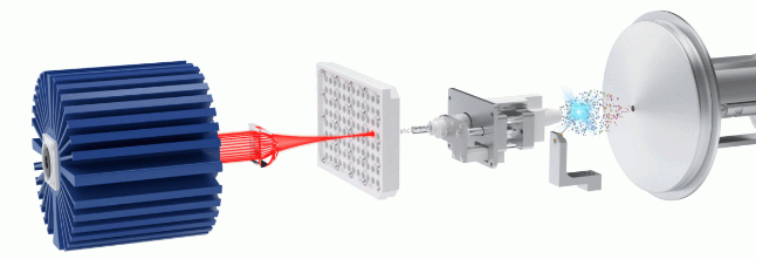

FIGURE 1.1 – **Système LDTD**

Ce procédé a plusieurs avantages. L'avantage majeur est sa rapidité par rapport aux autres méthodes d'ionisation et de spectrométrie de masse, puisqu'un échantillon peut être analysé en environ 10 secondes par LDTD. Cela permet donc l'utilisation à haut débit du spectromètre. Par contre, cette approche comporte aussi l'inconvénient qu'il n'est pas possible de mesurer les protéines et peptides par LDTD. Le chauffage par laser ne vaporise pas les protéines, mais les fait brûler dans le puits de la plaque. Il faut donc un traitement des échantillons afin d'isoler les peptides et protéines.

Prétraitement des données

La première étape de prétraitement est la centroïdation des pics. La centroïdation consiste à faire une moyenne pondérée des points formant un pic afin d'en trouver le centre de masse. C'est une technique standard dans le domaine du traitement de signal, utilisée entre autres dans le traitement de signal sonore. Il est utile de trouver les centroïdes d'un pic pour ensuite pouvoir traiter le pic comme étant un point unique sur l'axe m/z. Cette étape est exécutée par le logiciel MassLynx de Waters Corporation.

Une autre étape de traitement des données importante qui fut découverte au courant du projet est l'élimination des pics de faible intensité. Sur chaque spectre de masse, on obtient quelques milliers de pics à très faible intensité. La présence de ces pics a plusieurs explications possibles. Une possibilité est que des effets quantiques et des erreurs sur le détecteur causent la détection de faibles impacts à des endroits où aucun ion ne se retrouve réellement. Une autre explication pour certains de ces pics est qu'ils sont causés par des ions très rares ou bien très peu volatiles. Dans les deux cas, il est mieux d'ignorer ces pics afin d'éviter d'établir à tort des corrélations entre ces pics à très faible intensité et les classes d'exemples. Certains des résultats préliminaires d'apprentissage automatique tentaient, par erreur, de prédire les classes à partir de pics extrêmement faibles. Cette erreur était due au fait que ces pics étaient corellés à la plaque et au jour où les échantillons étaient mesurés.

Malheureusement, aux fins de ce projet et de l'apprentissage automatique, certaines propriétés de la spectrométrie de masse et des étapes de prétraitement décrites ci-dessus introduisent certaines incertitudes auxquelles il faut pallier. Le type principal d'incertitude qu'il nous faut nécessairement corriger est une variation au niveau des ratios de masse sur charge. Puisque nous considérons les différentes masses détectées comme étant des caractéristiques pour l'apprentissage automatique, les algorithmes ne peuvent pas naturellement compenser ces différences, contrairement à des variations en intensité pour une caractéristique donnée qui pourraient être contrebalancées.

Il existe deux types de variations à solutionner. La première est une variation aléatoire de faible grandeur. Ce type de variation est causé par de légères incertitudes de mesures aléatoires et des effets quantiques dans le détecteur de masse du spectromètre, et donc un effet instrumental. La centroïdation des pics contribue aussi à ce type de variation, puisque c'est une procédure basée sur le centre de masse d'un pic. Un petit changement dans la distribution d'un pic peut ainsi changer la masse qui sera attribuée au centre du pic, même si ce n'est que de quelques décimales. Ce problème n'est pas aidé par l'utilisation de la technologie LDTD, puisqu'il n'est pas possible de séparer les ions selon une

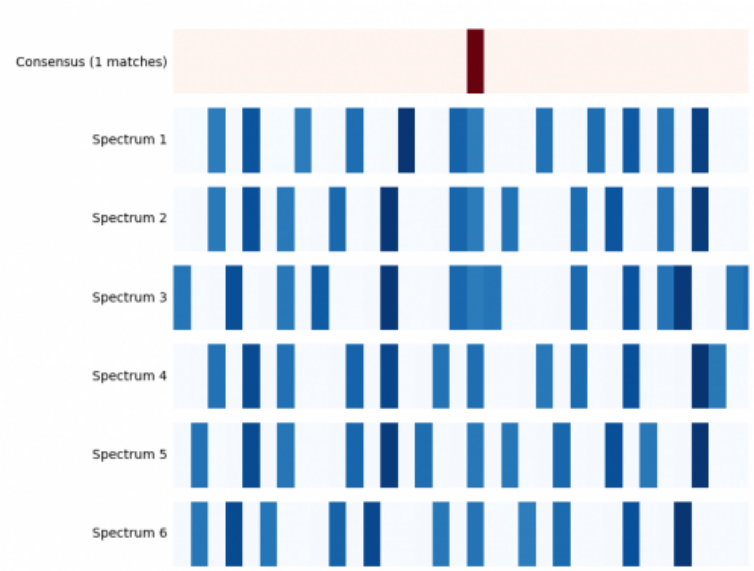

FIGURE 1.2 – **Exemple du désalignement des pics dans six spectres d'échantillons différents.** La première ligne contient les pics consensus, donc présents dans tous les spectres. Dans les autres lignes, dénotées "Spectrums", on a une ligne bleue aux endroits où l'on retrouve un pic. L'intensité de la couleur de la barre est proportionnelle à l'intensité du pic. On remarque que, sur six spectres, on ne retrouve qu'un seul pic consensus. Pour la plupart des autres pics, on remarque un léger désalignement entre chacun des spectres, ou du moins entre une partie des spectres et les autres.

affinité pour un milieu ou une solubilité comme dans une technique de chromatographie. On risque alors d'avoir de multiples molécules et fragments de molécules qui ont des ratios de masse sur charge très similaire, ce qui peut entrainer un chevauchement entre les pics qui sont associés à ces molécules. Les variations de ce type sont de l'ordre de quelques parties par million (ppm) de la masse mesurée, généralement dans la région de 5 à 10 ppm. Par exemple, une erreur de 5 ppm sur une masse mesurée de 500.0000 Da est de 0.0025 Da. On peut voir un exemple de ce type de désalignement à la figure 1.2.

Ce problème est déjà connu en spectrométrie de masse, puisque les effets de chevauchement entre ions et de légères incertitudes de mesure sont universels à cette technologie. Par contre, les algorithmes pour réaligner les pics de différents échantillons existant dans la littérature sont conçus pour les techniques de spectrométrie de masse par chromatographie liquide (LC-MS) et utilisent les données chromatographiques pour aligner les pics. Ils sont donc incompatibles avec nos données obtenues avec la technologie LDTD, puisque l'acquisition ne contient pas de dimension de temps de rétention nécessaire à ces algorithmes. Pour solutionner ce problème, nous avons donc mis au point un algorithme d'alignement entre les pics et qui n'utilise pas de données de chromatographie. Cet algorithme et ses résultats forment une section de ce chapitre.

Le deuxième type de variation est une variation systématique des masses entre des spectres acquis des jours différents. Expérimentalement, nous avons aussi constaté que ce type de variation était claire-

ment présent entre des plaques d'échantillons mesurés sur des calibrations différentes du spectromètre, ce qui en ferait un effet instrumental. La calibration du spectromètre a comme objectif de compenser pour les changements atmosphériques et tout changement interne de la machine et ainsi placer la même molécule à la même masse de calibration en calibration. Cela est fait en envoyant un mélange connu de molécules donnant un spectre qui est déjà connu. La machine va ainsi replacer les masses aux bons endroits du spectre.

Encore une fois, le fait que l'application de la spectrométrie de masse dans ce projet soit appliquée de manière moins classique fait que la technique existante ne solutionne pas entièrement le problème dans notre cas. Une application plus classique va donc utiliser les masses de verrouillage (*lock mass* en anglais) pour replacer exactement une petite section du spectre où l'on sait qu'on veut doser une molécule précise. Dans notre application, il faut un plus grand nombre de masses de verrouillage qui sont réparties sur l'entièreté ou presque du spectre. Ceci est nécessaire car on requiert une correction la plus exacte possible et qui minimise au maximum les variations de masses des pics sur le spectre à large bande complet. Nous avons donc mis au point une méthode afin de corriger les déviations sur le long du spectre avec une plus large bande que l'approche par masses de verrouillage classique. Cette méthode forme une autre section de ce chapitre.

1.2 Alignement des pics

Le premier de ces désalignements que nous avons tenté de solutionner est celui du désalignement aléatoire. Ce fut pour plusieurs raisons. Une première est que ce désalignement fut détecté et quantifié en premier. De plus, c'est un désalignement particulièrement important à régler pour la suite du projet. Puisque nous considérons chaque position sur l'axe m/z où l'on retrouve un pic dans au moins un échantillon, il nous faut un alignement pour ces variances de masses. De plus, les algorithmes d'apprentissage ont en général des moyens d'être robustes au bruit dans les données. Cette robustesse s'exprime généralement en tant que résistance au bruit sur les valeurs dans une caractéristique d'exemple en exemple, soit à l'intensité des pics dans notre cas. Par contre, il est beaucoup plus complexe d'avoir une résistance à un bruit sur les caractéristiques.

1.2.1 Évaluation du problème

Jeu de données d'évaluation de reproductibilité Les variances en intensité des masses et des intensités de pics ont été évaluées sur un jeu de données fait pour mesurer la reproductibilité des données acquises sur une plaque par LDTD. Les données utilisées pour cet exercice étaient 8 échantillons de plasma provenant d'une même plaque d'échantillons. Chacun a été placé 12 fois sur la plaque. Nous avons donc 96 échantillons, soit 12 réplicats de 8 échantillons différents. Les échantillons ont subi une extraction à l'acétonitrile (ACN) et au méthanol (MeOH) dans une proportion de 75 :25. La proportion de l'extraction était de 10 μL d'échantillon pour 90 μL de solution ACN :MeOH. Cette étape est un *crash* à l'acétonitrile, méthode fait précipiter les protéines et peptides en solution Xu et collab.

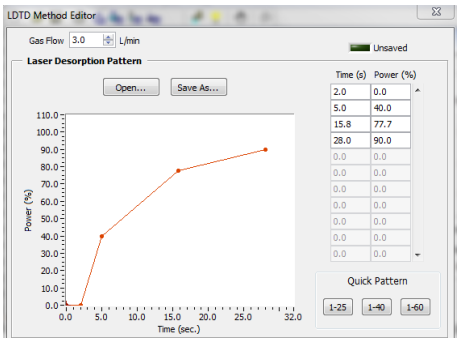

FIGURE 1.3 – **Patron laser utilisé en LDTD**

(2005). Par la suite, chaque échantillon de plasma a subi une sonication pour une période de 5 minutes, afin de s'assurer de la fragmentation des biomolécules. On fait ensuite une centrifugation de 5 minutes à 5000 RPM afin de concentrer les protéines précipitées au fond du tube et l'on récupère le surnageant. Les échantillons ont ensuite été dilués dans une proportion 1 :10 afin d'éviter la suppression ionique, qui peut arriver si l'on a trop d'ions dans l'échantillon acquis. Du clomiphène fut ajouté dans chaque échantillon. Par la suite, 2 μL de solution furent pipettés dans chaque puit de la plaque LazWell utilisée.

Les données furent ensuite acquises par LDTD et avec un spectromètre Synapt G2 de Waters Corporation. La figure 1.3 montre le patron laser utilisé par le LDTD pour cette expérience. Le laser reste fermé 2 secondes. Il s'allume et monte à 40% de sa force maximale en 3 secondes. Le laser continue ensuite de chauffer en montant à 77% de sa force maximale les 10 secondes suivantes. Le laser termine ensuite en chauffant graduellement jusqu'à 90% de sa force maximale durant les 13 secondes suivantes, avant de se fermer. L'acquisition sur le spectromètre Synapt G2 se fait ensuite en mode *high resolution*. Ce mode a été choisi puisqu'il est plus spécifique sur les masses, ayant une erreur moindre, et qu'il est moins sensible. Cela représente un avantage dans la situation puisqu'un grand nombre d'ions frappe le détecteur en peu de temps vu la technologie LDTD. Cette diminution de sensibilité va donc aider à prévenir la suppression du signal au détecteur. Le spectre est aussi acquis en mode MS simple, avec une rampe d'énergie de collision. L'énergie de collision est une énergie ajoutée dans la cellule de collision afin de favoriser la fragmentation des molécules. Dans cette expérience, le gradient d'énergie était de 0V à 35V sur les 28 secondes d'acquisition.

Le clomiphène fut ajouté dans cette expérience pour fournir un pic dans chaque échantillon qui représente une molécule connue. Le clomiphène est une molécule ayant une masse connue de 406.19322 Da. C'est cette molécule qui nous a servi, à cette étape du projet, de masse de verrouillage. Cette molécule a été choisie pour l'expérience de reproductibilité vu qu'aucun autre pic de grande intensité n'est près de la région où cette molécule paraît. Ainsi, en ayant un seul pic de haute intensité dans cette région, nous sommes certaines de comparer le pic de la même molécule dans tous nos échantillons. De plus, le fait de rajouter une quantité égale de cette molécule dans chaque échantillon permet d'évaluer la variance en intensité du pic. On peut aussi se fier que cette molécule ne sera pas présente à

l'avance dans les échantillons, puisque le clomiphène est une molécule pharmaceutique qui n'est pas disponible au Canada.

Une portion de l'évaluation a aussi été effectuée sur un second pic dans les spectres. C'est un pic dans la région de 369 Da. Ce pic est un pic candidat pour le cholestérol (masse de 386.65 Da pour la molécule complète). Ce pic a aussi été choisi puisqu'il a une forte intensité et est isolé d'autres pics d'intensité similaire. S'il s'agit effectivement d'un pic formé par un fragment de cholestérol, on peut aussi considérer ce pic afin d'évaluer la variance d'intensité des pics, puisque l'homéostasie du sang devrait oeuvrer pour garder un niveau très similaire de cholestérol dans le plasma. Cette mesure sera plus certaine entre les réplicats d'un certain échantillon, alors qu'on s'attend que la quantité de clomiphène soit constante dans tous les échantillons et réplicats.

TABLE 1.1 – **Variance du pic de clomiphène**. Expérience de reproductibilité, en termes de masse et d'intensité.

Masse moyenne (en Da)	406.2001
Écart type de la masse (en Da)	0.0003
Écart type de la masse (en ppm)	0.7386 ppm
Intensité moyenne	36889
Écart type de l'intensité	7242
Écart type de l'intensité (en %)	19.63%

On remarque dans la table 1.1 que la variance est relativement faible au niveau de la masse du pic de clomiphène, avec une variance en ppm de moins de 1 ppm. Par contre, même avec cette erreur relativement faible, on remarque qu'en termes de Daltons, l'écart type est quand même équivalent à $\pm$ 0.0003 Da. Vu que le spectromètre utilisé est précis sur la position des pics jusqu'à 0.0001 Da, il y a clairement un changement de position des pics entre les échantillons qui sera visible dans nos données. Les données statistiques sur l'intensité ont aussi été incluses dans ce tableau. On remarque qu'avec nos instruments et méthodes expérimentales, on observe un écart type d'environ 20% de la valeur moyenne d'intensité à travers les réplicats, alors qu'une quantité égale de clomiphène a été ajoutée à chacun des échantillons.

TABLE 1.2 – **Variance du pic de cholestérol.** Expérience de reproductibilité, en termes de masse et d'intensité

Masse moyenne (en Da)	369.3577
Écart type de la masse (en Da)	0.0002
Écart type de la masse (en ppm)	0.5415 ppm
Intensité moyenne	309373
Écart type de l'intensité	67035
Écart type de l'intensité (en %)	21.67%

Considérons maintenant les résultats de l'expérience de reproductibilité pour les pics de cholestérol, présentés à la table 1.2. La masse moyenne à laquelle le pic candidat est retrouvé dans les 96 réplicats

est de 369.3577 Da. L'écart type autour de cette moyenne est de $\pm$ 0.0002 Da. Encore une fois, cette déviation est très faible et revient à être moins de 1 ppm. Au niveau de l'intensité, on remarque que l'écart type relatif de l'intensité de ce pic est supérieur à l'écart type de l'intensité du pic de clomiphène. Par contre, la même quantité de clomiphène a été ajoutée à chaque puits, alors qu'on a une quantité variable de cholestérol pour chacun des 8 individus. Il est donc plus significatif de considérer les comparaisons entre réplicats du même individu pour le cholestérol. Ces données sont montrées dans la table 1.3.

TABLE 1.3 – **Variance de l'intensité du pic de cholestérol par individu.** Expérience de reproductibilité, 12 réplicats par individu.

Individu	Moyenne d'intensité	Écart type	Écart type (en %)
1	319424	54557	17.07%
2	238315	84702	35.54%
3	302878	52292	17.26%
4	322434	36216	11.23%
5	366026	51748	14.13%
6	318783	52221	16.38%
7	263575	46200	17.53%
8	343550	47826	13.92%

On voit clairement dans la table 1.3 que l'écart type est, en général, beaucoup plus bas à l'intérieur des réplicats d'un même individu. L'exception ici est le second individu qui a un écart type beaucoup plus élevé. Cela est probablement dû à un réplicat aberrant. Si l'on considère les moyennes et écarts types pour les pics de clomiphène (présentées à la table A.1 en annexe), on remarque que l'individu 2 a également un écart type d'environ 30% alors que les autres en ont un de moins de 20%.

1.2.2 Première version de l'algorithme

Le problème de désalignement aléatoire présenté dans la section précédente et à la figure 1.2 motive donc un nouvel algorithme d'alignement des pics. Contrairement aux algorithmes classiques adaptés à la spectrométrie de masse de types LC-MS, qui utilisent les données de masses et de chromatographie, il nous faut un algorithme utilisant seulement les masses des spectres. Il faut aussi que cet algorithme possède un paramètre permettant de régler à quelle distance il peut aligner des pics.

Le premier algorithme d'alignement mis au point durant ces travaux repose sur l'idée que les pics désalignés qui représentent la même molécule se retrouveront à une faible distance les uns des autres, et se retrouveront donc dans une fenêtre de faible taille. Cet algorithme présume aussi que les pics représentant des molécules différentes dans différents spectres seront distants. Il utilise aussi un paramètre de distance maximale d'alignement. Ce paramètre est considéré en unité partie par million (ppm), afin d'avoir une distance qui grandit lorsqu'on considère de plus grandes masses. Cette unité est donc relative à la masse sur laquelle elle est appliquée. La formule pour calculer une distance en ppm est la suivante :

$$\frac{(\mu_2 - \mu_1)}{\mu_1} * 10^6 = x \text{ ppm}$$

Une incertitude de 0.0010 Da sur une masse de 100.0000 Da correspond donc à 10 ppm, autrement dit ($\frac{0.0010}{100.0000} \times 10^6 = 10ppm$). À titre d'exemple, une même incertitude de 0.0010 Da appliquée sur une masse de 1500.0000 Da correspond à 0.666 ppm. La décision d'utiliser cette unité est supportée par le fait que l'incertitude sur les pics dépend de la masse observée, tout comme les ppm.

L'algorithme d'alignement se fait en deux temps. En premier lieu, on construit un spectre de référence, ou le spectre repère (*landmark spectrum* en anglais). Ce spectre repère est une liste de masses placées en ordre croissant. C'est sur cette référence que les spectres du jeu de données seront alignés. C'est cette étape qui est la plus importante et coûteuse computationellement pour un alignement de spectres. Le premier algorithme de construction de spectres de référence est présenté à l'algorithme 1, CONSTRUCTION REPÈRE.

Expliquons ici l'algorithme 1, CONSTRUCTION REPÈRE. On considère l'ensemble des spectres. Chaque spectre est composé d'une liste de paires de masses et intensités. On considère maintenant un nouveau spectre que l'on appelle le spectre cumulé, qui contient toutes les masses présentes dans au moins un spectre du jeu de données. Dans ce spectre, l'intensité est remplacée par le nombre d'occurrences des masses dans le jeu de données. Si l'on a, par exemple, quatre spectres avec des pics aux masses, soit 100.0000 Da, 100.0002 Da, 100.0001 Da et 100.0001 Da ; le spectre cumulé aura trois pics : un pic de masse 100.0000 Da et d'intensité 1, un pic de masse 00.0001 Da et d'intensité 2 et un pic de masse 100.0002 Da et d'intensité 1.

La deuxième partie de l'algorithme consiste à parcourir de "gauche à droite", soit des plus petites masses vers les plus grandes masses, le spectre cumulé. À chaque nouveau pic, on regarde si l'on retrouve d'autres pics dans une fenêtre en ppm, notée d, commençant à ce pic. On a alors une fenêtre f_i dont les limites sont définies pour la masse μ_i par :

$$f_i = [\mu_i, \mu_i + \mu_i \cdot \frac{d}{10^6}]$$

On répertorie alors tous les pics se retrouvant dans la fenêtre f_i. On calcule ensuite la masse moyenne pondérée par l'intensité (le nombre d'occurrences) de ces pics. Cette masse moyenne sera ajoutée au spectre de repère. On recommence ensuite à parcourir le spectre cumulé vers des masses plus grandes, à partir de la première masse qui n'était pas dans la fenêtre que l'on vient de considérer. Ainsi, on parcourt le spectre sommé des petites masses vers les plus grandes, sans revenir en arrière, et l'on ajoute ainsi des points au spectre de repère. Chaque masse présente dans le spectre cumulé ne peut être que dans un seul point de repère et chaque masse sera utilisée pour construire le spectre de référence.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

Algorithm 1 CONSTRUCTION REPÈRE, première implémentation

Require: Un ensemble de m spectres. Chaque spectre comprend une liste de pics, avec une masse et une intensité. $S =<< \mu_1, \iota_1 >, ..., < \mu_m, \iota_m >>$
Require: Un paramètre de distance maximale d'alignement à considérer en ppm, d
  for all spectres **do**
    Mettre toutes les valeurs d'intensité à 1 pour les pics existants.
  end for
  Sommer les spectres binaires en un spectre cumulé.
  Créer un spectre de repère vide, S_R
  while $i < m$ **do**
    Trouver les masse présentes dans la fenêtre $[\mu_i, \mu_i + \mu_i \cdot \frac{\frac{d}{2}}{10^6}]$
    Faire la moyenne pondérée des masses dans la fenêtre et l'ajouter à la fin du spectre de repère
    $i =$ index de la masse maximale dans la fenêtre $+1$
  end while
  return S_R

La deuxième étape du processus consiste à aligner les spectres du jeu de données sur une référence fournie. Cette partie de l'algorithme est présentée à l'algorithme 2 (algorithme ALIGNE).

Algorithm 2 ALIGNE

Require: Le spectre de repère comprenant n pics, $S_R =< \overline{P_1} =< \overline{\mu_1}, 1 >, ..., \overline{P_n} =< \overline{\mu_n}, 1 >>$
Require: Un spectre à aligner comprenant o pics, $S =< P_1 =< \mu_1, \iota_1 >, ..., P_n =< \mu_o, \iota_o >>$
Require: la distance maximale en ppm à considérer, d
  On crée S_a, le spectre aligné
  $\widehat{\mu_0} = \mu_1$
  for $i = 1$ to o **do**
    $f_i = [\mu_i - \mu_i \cdot \frac{\frac{d}{2}}{10^6}, \mu_i + \mu_i \cdot \frac{\frac{d}{2}}{10^6}]$
    On trouve $j \in 1, ..., n$ tel que $|\mu_i - \overline{\mu_j}|$ est minimal
    if $\overline{\mu_j} \in f_i$ **then**
      $\widehat{\mu_i} = \overline{\mu_j}$
    else
      $\widehat{\mu_i} = \max(\mu_i, \widehat{\mu_{i-1}})$
    end if
    On ajoute $< \widehat{\mu_i}, \iota_i >$ à la fin de S_a
  end forreturn le spectre aligné

Détaillons l'algorithme 2, l'algorithme ALIGNE, qui aligne un spectre au spectre repère. Le principe est similaire à celui de la construction du spectre de repère S_R puisqu'il opère sur le principe de parcourir le spectre des plus petites masses aux plus grandes. On parcourt donc tous les pics du spectre à aligner. On calcule à chaque cas la fenêtre de masse f_i correspondant à la masse et plus ou moins la distance maximale en ppm à considérer.

On trouve ensuite le pic du spectre de repère $\overline{P_j}$ dont la masse $\overline{\mu_j}$ est la plus près de celle du pic $P_i =< \mu_i, \iota_i >$. Si cette masse $\overline{\mu_j}$ ce retrouve dans la fenêtre f_i autour de μ_i, on assigne $\overline{\mu_j}$ comme valeur à $\widehat{\mu_i}$. Si $\overline{\mu_j}$ ne se retrouve pas dans f_i, alors on assigne à $\widehat{\mu_i}$ la valeur maximale entre $\widehat{\mu_{i-1}}$ et μ_i.

On ajoute donc le pic $< \widehat{\mu}_i, \imath_i >$ à la fin du spectre à aligner S_a.

On applique généralement l'algorithme d'alignement différemment aux ensembles d'entraînement et de test. Dans le contexte d'une expérience d'apprentissage automatique, on ne veut pas prendre de décisions basées sur de l'information prise de l'ensemble de test. Construire un spectre repère à partir des spectres de l'ensemble de test violerait cette contrainte. La procédure utilisée au courant de ce projet est que l'on construit le repère sur les spectres de l'ensemble d'entraînement seulement. On aligne chaque spectre d'entraînement au spectre de repère. On aligne ensuite les spectres de l'ensemble de test au spectre de repère. Ainsi, les spectres d'entraînement et ceux de tests subissent le même traitement sans que l'on retire de l'information de l'ensemble de test.

Dans une situation où l'on aurait un très grand nombre de spectres, on peut ne travailler que sur un sous-ensemble de spectres de l'ensemble d'entraînement afin de construire le spectre de repère. En effet, la construction de ce spectre est l'étape la plus intense computationellement, en termes de temps de calcul et de mémoire vive requise. En comparaison, aligner les spectres individuels utilise largement moins de ressources. Par ailleurs, l'utilisation d'un sous-ensemble peut être suffisante pour construire une bonne référence d'alignement. Si l'on sélectionne de manière indépendante et identiquement distribuée (*i.i.d.*) dans l'ensemble des spectres disponibles, on peut avoir un échantillon qui est statistiquement représentatif de l'ensemble.

1.2.3 Résultats

Les données utilisées pour générer ces résultats proviennent de huit échantillons de plasma. Les échantillons proviennent de la même plaque d'échantillons et ont été acquis sur la même plaque LazWell. Les échantillons ont d'abord été décongelés. Ils ont subi une extraction à l'ACN :MeOH 75 :25 par la suite. On a extrait 10 μL d'échantillon avec 90 μL de solution ACN :MeOH. Cette extraction précipite les protéines. Ils ont subi une sonication pendant 5 minutes. Ils ont ensuite été centrifugés à 5000 RPM pendant 5 minutes afin de sédimenter les protéines. Finalement, les échantillons ont été dilués dans une proportion 1 :10 afin d'éviter la suppression ionique. Ils ont alors été acquis par LDTD et dans un spectromètre Synapt G2 de Waters Corporation.

Le patron laser utilisé par le LDTD est le même que décrit plus haut et présenté à la figure 1.3. C'est donc 28 secondes d'acquisition par échantillon, pendant lequel le laser monte jusqu'à 90% de sa force maximale. Les spectres sont alors acquis sur le Synapt G2 en mode MS avec rampe d'énergie de collision avec un gradient de 0V à 35V. Les spectres ont également été acquis en mode *high resolution* sur le Synapt G2 afin d'avoir une meilleure précision sur la masse des pics.

Les données ont subi un traitement automatique par le logiciel MassLynx de Waters corporation, faisant entre autres la centroïdation des pics. Pour ces résultats, un seuil d'intensité de 500 fut appliqué sur les spectres, signifiant que les pics ayant une intensité inférieure à 500 furent retirés. De plus, seuls les pics dont le ratio masse sur charge sont entre 55.0000 mua et 60.0000 mua furent conservés. Finalement, l'algorithme d'alignement fut appliqué sur les huit spectres avec un paramètre de distance

maximale de 10 ppm.

Finalement, les figures suivantes furent générées. Ces types de graphiques sont des "*heatmaps*". Chaque colonne de la figure est une position sur l'axe m/z, et chaque rangée est un des échantillons. L'intensité de chaque pic est représentée par la couleur, allant de blanc jusqu'au bleu foncé pour des pics de plus haute intensité.

Les figures 1.4 et 1.5 indiquent, respectivement, l'alignement dans le cas des échantillons sans aucun alignement et avec l'algorithme d'alignement avec une distance de 10 ppm appliqué.

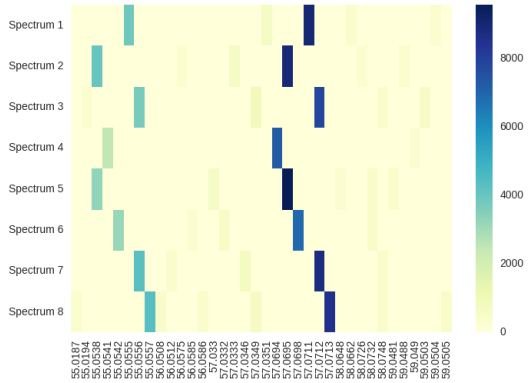

FIGURE 1.4 – **"*Heatmap*" d'alignement de spectres.** Les huit échantillons inclus n'ont pas été alignés.

On remarque à la figure 1.4, où les spectres utilisés n'ont pas été alignés, que l'on a des pics de haute intensité qui sont présents à des masses adjacentes ou proches, mais rarement exactement les mêmes. C'est dû au même effet visible à la figure 1.2.

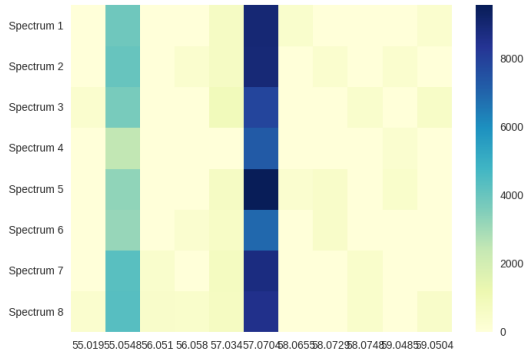

FIGURE 1.5 – **"*Heatmap*" d'alignement de spectres.** Les huit échantillons inclus ont été alignés avec un paramètre de distance de 10 ppm.

Au contraire, on peut remarquer à la figure 1.5 que ces pics de haute intensité ont maintenant, dans chaque échantillon, une masse commune. On a donc à ce point des spectres plus comparables entre eux. On peut donc procéder avec une analyse des pics des spectres en comparant les valeurs d'intensité sur une base plus commune au niveau des masses.

1.3 Correction des déviations par masse de verrouillage virtuelles

Comme mentionné dans l'introduction, il existe un second type de variation des masses. Ce type, au lieu d'être aléatoire à chaque pic, est relativement uniforme à travers le spectre. Cette variation est due à plusieurs facteurs environnementaux, tels que les changements de température environnant le spectromètre ou le taux d'humidité. Le fait de recalibrer l'instrument entre différentes plaques d'échantillons peut aussi induire des déviations. Des masses de verrouillage sont normalement utilisées en spectrométrie de masse pour compenser ces effets. Les masses de verrouillage sont des composés ajoutés dans les échantillons afin de corriger les pics avoisinants du pic associé au composé. Cette correction est déterminée par la différence entre la masse connue du composé et la masse à laquelle on observe le pic associé au composé.

Par contre, vu que l'approche choisie dans ce projet au niveau de la spectrométrie de masse est d'acquérir des spectres à larges bandes afin de doser une grande quantité de métabolites, au lieu d'acquérir de manière précise une petite section de spectre, il est beaucoup plus ardu de faire ces corrections par masses de verrouillage. Pour ce faire, il faudrait ajouter un grand nombre de composés chimiques dans les échantillons avec des masses variées pour permettre une bonne correction à tous les endroits du spectre.

Pour solutionner ce problème de manière *in silico*, un algorithme de correction a été mis au point utilisant des masses de verrouillage virtuelles, ou *Virtual Lock Mass* (VLM). Ces VLM sont simplement des pics que l'on retrouve dans chacun des échantillons sur lequel il est possible de se baser afin de corriger les spectres. L'idée d'utiliser des pics de cette manière vient de la connaissance biologique que les produits sanguins sont fortement structurés et maintenus par homéostasie. On s'attend à retrouver plusieurs métabolites en grandes quantités dans le sang de n'importe quel individu vivant. Contrairement à des masses de verrouillages ajoutées dans les échantillons, les masses réelles des VLM ne sont pas connues. Leur rôle est de corriger tous les spectres dans un jeu de données afin qu'ils soient comparables, donc de replacer les pics homologues dans différents spectres à la même masse.

Plusieurs caractéristiques sont recherchées dans un pic candidat afin d'être un VLM. Une caractéristique très importante est que le pic candidat doit être présent dans tous les spectres du jeu de données. Ceci est nécessaire pour avoir une couverture équivalente dans chaque spectre. Un second critère est qu'il faut que le pic soit de haute intensité. Ce critère est présent pour deux raisons. Premièrement, un pic de plus haute intensité aura moins de variation dans sa masse au moment de la centroïdation. Deuxièmement, des pics de plus haute intensité auront plus tendance à être présent dans tous les échantillons et avec une intensité plus constante, sous l'hypothèse que ces pics sont des éléments importants dans la composition du sang. Finalement, il faut qu'un pic soit isolé d'autres pics afin d'être un bon candidat pour être un VLM. Ce critère est pour s'assurer qu'on n'ait pas d'ambiguïté au moment de la recherche du pic VLM dans les spectres. Si un spectre est seul, ou du moins seul à être de très haute intensité dans une petite fenêtre, on a moins de chance de prendre un pic voisin et ainsi

induire des erreurs dans la correction d'une grande partie du spectre.

1.3.1 Effet des déviations

Un bon indicateur de la nécessité de la correction est visible si l'on approche cette question d'un point de vue d'apprentissage automatique. Tel que mentionné précédemment, les algorithmes d'apprentissage automatique ont de la difficulté, ou même dans certains cas sont incapables, de compenser pour un type de bruit dans les données qui est d'un type où les caractéristiques elles-mêmes sont bruitées, comme dans le cas présent où les caractéristiques sont des intensités associées à une valeur de masse précise.

Voici un exemple qui illustre la problématique. Deux plaques d'échantillons dont les spectres ont été acquis lors de différents jours ont été comparées par apprentissage automatique. La première plaque était considérée la classe positive (dénotée +1) et l'autre plaque a été désignée la classe négative (-1).

Jeu de données d'évaluation des dévations Les échantillons étaient des plasmas sanguins. Ils ont premièrement été décongelés. Par la suite, ils ont subi une extraction à l'ACN :MeOH dans une proportion 75 :25. L'extraction se fait en mélangeant 10 μL d'échantillon de plasma et 90 μL de solution d'extraction ACN :MeOH. Les échantillons ont ensuite subi une sonication pendant 5 minutes. On centrifuge alors les échantillons à 5000 RPM pendant 5 minutes. On récupère le surnageant, qui est la solution de l'échantillon sans protéines. Les échantillons ont aussi été extraits dans une solution de chlorobutane. Dans ce cas, on mélange 10 μL de plasma avec 90 μL de chlorobutane. On fait également une sonication de 5 minutes et une centrifugation de 5 minutes à 5000 RPM. On récupère dans ce cas la phase organique (supérieure) afin de conserver les molécules non polaires du plasma. On a ajouté à ce moment du clomiphène, qui sert de masse de verrouillage, dans chaque échantillon. Les échantillons ont ensuite été dilués dans une proportion de 1 :10 avant d'être pipettés sur la plaque LazWell.

Les spectres ont été acquis par LDTD et sur un spectromètre Synapt G2 de Waters Corporation. On utilise pour ces données le même patron laser qui fut décrit précédemment et montré à la figure 1.3. Les spectres furent acquis en mode *high resolution* pour avoir une meilleure précision sur les masses des pics. Les spectres furent également acquis en mode MS avec une rampe à énergie de collision, afin de fragmenter les molécules. La rampe à énergie variait de 0V à 35V au cours de l'acquisition.

Les spectres acquis ont ensuite été prétraités. Le premier traitement est la centroïdation, faite par le logiciel MassLynx de Waters Corporation. Les échantillons ont ensuite été corrigés par masse de verrouillage réelle. Le pic utilisé était le pic de clomiphène, avec une masse connue de 406.18594223 Da. Une filtration des pics par intensité a ensuite été appliquée aux spectres, enlevant tout pic avec une intensité inférieure à 500. Ce seuil a été choisi afin d'éliminer les pics de faible intensité dus au bruit de fond de l'appareil. Aucune correction par masses de verrouillages virtuelles n'a été appliquée pour cet exemple. Un alignement a ensuite été appliqué aux données, soit l'alignement décrit à la section précédente. Plusieurs paramètres maximaux de distance ont été testés, soit 5 ppm, 10 ppm,

15 ppm et 20 ppm. Par la suite, seuls les 500 pics les plus intenses par spectre ont été conservés pour la classification afin de rendre le problème plus tractable computationellement pour certains des algorithmes. De plus, comme on considère deux spectres par échantillon (extraction ACNMeOH et ions positifs, extraction Chlbut et ions négatifs), on conserve 1000 pics par échantillon. Les pics ont finalement été binarisés, c'est-à-dire que l'on considère seulement si chaque pic est présent ou non et pas son intensité. Ce choix a été fait afin de permettre l'application de l'algorithme de la machine à couverture d'ensembles (*Set Covering Machine*, SCM. Voir le chapitre 2 pour plus d'informations).

Cinq algorithmes ont été testés sur les données ainsi générées. Ils seront détaillés au chapitre 2,qui porte sur l'aspect d'apprentissage automatique du projet. Les échantillons des deux plaques ont été séparés entre un ensemble d'entraînement de 162 exemples et un ensemble de test de 30 exemples. Les paramètres optimaux de chaque algorithme ont été choisis par validation croisée à 5 plis. Chacun des cinq algorithmes a son propre espace de paramètres à explorer.

Les forêts d'arbres décisionnels (*Random Forest*) ont un paramètre du nombre d'estimateurs à utiliser Breiman (2001). Ce paramètre est choisi par validation croisée parmi les valeurs suivantes : $\{1,5,10,20,30,40,50,60,70,80,90,100,150\}$. L'arbre de décision (*Decision Tree*) a deux paramètres à valider Breiman et collab. (1983). Le premier est la profondeur maximale de l'arbre, qui peut être dans l'ensemble $\{2,3,4,5,6,7,8,9,10\}$. Le second est le nombre minimal d'exemples qu'une règle doit séparer afin d'être retenue. Ce paramètre pouvait avoir une valeur dans les suivantes : $\{2,5,10,15,20\}$. Lorsqu'un algorithme a plusieurs paramètres à valider, la validation croisée va essayer toutes les combinaisons possibles des valeurs de ces paramètres afin de trouver l'optimale. Le prochain algorithme est la machine à vecteur de support (*Support Vector Machine*, SVM) utilisant un noyau linéaire, c'est-à-dire essayant de séparer les exemples dans l'espace des caractéristiques fournies à l'algorithme Cortes et Vapnik (1995). Cette version de l'algorithme n'a qu'un seul paramètre, C, qui régule la souplesse de la marge. Ce paramètre C se retrouve dans un espace logarithmique de 7 valeurs entre 10^{-3} et 10^{3}, soit l'ensemble $\{0.001,0.01,0.1,1,10,100,1000\}$. Une autre version du SVM a été utilisée, celle du noyau "*Radial Basis Function*" (RBF) Vapnik (1995). Cette version de l'algorithme contient aussi le paramètre C et aussi un paramètre γ. Ces deux paramètres ont des valeurs choisies par validation croisée dans la même espace que le paramètre C dans le cas du noyau linéaire, soit l'ensemble $\{0.001,0.01,0.1,1,10,100,1000\}$. Finalement, l'algorithme du SCM fut testé Marchand et Taylor (2002). Son paramètre du nombre maximal d'attributs utilisable pouvait avoir des valeurs de $\{1,2,3,4,5,6,7,8,9,10,12,15\}$. Son paramètre p pouvait adopter des valeurs dans un espace logarithmique de 5 valeurs entre 10^{-2} et 10^{2}. Le SCM pouvait aussi avoir un modèle de conjonction ou de disjonctions.

Les résultats de ces classificateurs sont donc présentés à la table 1.4. On remarque rapidement que peu importe le paramètre d'alignement, les algorithmes du SVM, avec ses deux noyaux, et l'algorithme du Random Forest sont capable en tout temps de séparer parfaitement ou presque les exemples par leur plaque de provenance. La tâche semble un peu plus difficile pour les algorithmes parcimonieux, le Decision Tree et le SCM, à certaines valeurs d'alignement, mais les précisions sont tout de même très

TABLE 1.4 – **Résultats d'apprentissage automatique sans correction par VLM sur les échantillons provenant de deux plaques.** La précision marque la performance du classificateur sur l'ensemble de test et indique le pourcentage de bonnes prédictions sur l'ensemble.

Distance d'alignement	Random Forest	Decision Tree	SVM Linéaire	SVM RBF	SCM
Alignement à 5 ppm	97.6%	97.6%	100%	100%	93.3%
Alignement à 10 ppm	100%	86.7%	100%	100%	86.7%
Alignement à 15 ppm	100%	100%	100%	100%	76.7%
Alignement à 20 ppm	96.7%	86.7%	100%	100%	66.7%

élevées, c'est-à-dire qu'ils peuvent faire assez facilement la différence entre les exemples de chaque calibration. On note en particulier qu'à 5 ppm de distance maximale d'alignement, le SCM et le Decision Tree ont une précision de plus de 90% avec, respectivement, 1 attribut et un maximum de 7 attributs (profondeur de 3, donc l'arbre contient entre 3 et 7 règles). Cela signifie donc que la présence ou l'absence d'un seul pic permet de distinguer la plaque source d'un exemple plus de 9 fois sur 10. Cela indique donc qu'il semble que, vu les décalages uniformes différents entre les calibrations, des pics qui sont en réalité homologues et représentent les mêmes molécules ionisées se retrouvent en fait à des masses différentes. De plus, l'ordre de grandeur de ce décalage est trop large pour être corrigé simplement par alignement.

Finalement, cet exemple nous indique qu'il est nécessaire de corriger ce problème de déviations afin de pouvoir faire des comparaisons exactes entre les échantillons. C'est pour régler ce problème que nous avons mis au point l'approche par masses de verrouillages virtuelles. Une première version de cet algorithme est présentée à la section 1.3.2.

1.3.2 Première version de l'algorithme

Dans la première version de l'algorithme de correction, les VLM devaient être fournis manuellement à l'algorithme. Il fallait donc faire le travail manuel afin de trouver les pics candidats et valider les différents critères. Ce travail était très coûteux en termes de temps. Cet aspect a été amélioré au cours du projet et dans la nouvelle implémentation (présenté à la prochaine section). L'algorithme 3 décrit l'algorithme de correction une fois que l'on a une série de points de corrections (VLM) trouvés manuellement.

Détaillons maintenant l'algorithme 3. L'algorithme décrit le traitement appliqué à chaque spectre. Comme mentionné plus haut, il faut aussi lui fournir les points de correction VLM. Comme les points de corrections n'ont pas de masses réelles connues, on assigne la masse moyenne observée pour ce pic au moment de faire le choix du point de correction. La première étape de l'algorithme est de trouver les pics homologues aux points de correction. Ceci est fait en regardant dans une fenêtre d'environ 0.05 Da. Le pic avec l'intensité la plus haute dans cette fenêtre sera considéré le pic équivalent au point de correction. Si un bon choix de VLM a été fait, le pic homologue dans le spectre sera effectivement le

Algorithm 3 CORRECTION VLM MANUELS

Require: Un spectre s à corriger
Require: Une liste de P points de correction de longueur n
  for $i = 0 \quad$ to $\quad n-1$ **do**
    Trouver le pic équivalent dans le spectre, μ_i^o
    $\sigma_i \leftarrow \frac{p_i}{\mu_i^o}$
  end for

  Corriger les masses des pics plus petits que p_1
  for $\mu < \mu_0^o$ **do**
    $\mu' \leftarrow \sigma \cdot \mu$
  end for

  $\mu'_0 \leftarrow p_0$
  for $i = 1 \quad$ to $\quad n-1$ **do**
    $slope \leftarrow \frac{\sigma_{i+1}-\sigma_i}{p_{i+1}-p_i}$
    $incercept \leftarrow \sigma_i - slope \cdot p_i$
    for $p_i < \mu < p_{i+1}$ **do**
      $\sigma \leftarrow slope \cdot \mu + intercept$
      $\mu' \leftarrow \sigma \cdot \mu$
    end for

  end for

  for $\mu > p_{n-1}$ **do**
    $\mu' \leftarrow \sigma_{n-1} \cdot \mu$
  end for
  return la liste de pics à masses corrigées μ'

pic de plus haute intensité (critères de haute intensité et d'isolation). Si l'algorithme ne trouve pas de pics dans la fenêtre spécifiée autour d'un point VLM, alors ce spectre sera considéré comme aberrant et ne sera pas corrigé. L'algorithme peut ainsi servir de contrôle de qualité pour les spectres, car les spectres qui ont eu un problème au cours de l'acquisition ont, expérimentalement, un ou plusieurs points de correction manquants.

On corrige ensuite les masses des pics du spectre section par section, où les sections sont délimitées par des points VLM. La section contenant les plus petites masses est corrigée par le ratio de correction σ_0 du premier point VLM. On fait de même avec les masses plus grandes que le pic associé au dernier point VLM (σ_{n-1}). Ceci revient à appliquer une correction constante sur toutes ces masses, puisqu'on n'a qu'un seul point pour estimer la déviation dans ces zones. Une fois que la première section est corrigée, on corrige le pic homologue au premier point VLM. Les corrections sur les pics associés aux points de VLM sont de simples multiplications par le ratio de correction calculé. Le ratio de correction est le ratio entre la masse attribuée au pic VLM et la masse observée. Cette multiplication va donc simplement ramener le pic associé au point de correction à la masse attitrée.

On fait ensuite la correction des zones entre chaque point. Pour ce faire, on interpole linéairement le facteur de correction nécessaire dans la section selon les facteurs de corrections des points délimitant cette section. Finalement, comme mentionné précédemment, on corrige les pics situés après le dernier point de correction. Au total, on a une interpolation linéaire par morceau nous fournissant les facteurs de correction nécessaire pour chaque pic du spectre, excepté les extrêmes du spectre où l'on applique une approximation en appliquant uniformément le facteur de correction du point VLM le plus près.

1.3.3 Évaluation de la performance de la correction par masses de verrouillage virtuelles trouvées manuellement

Les données sur lesquelles ont été testés les VLM sont des échantillons de plasma sanguin. Les données ont subi le même traitement que les échantillons utilisés pour les résultats d'alignement, présentés à la Section 1.3.1 au paragraphe portant sur le *Jeu de données d'évaluation des déviations*.

Les spectres acquis ont subi un prétraitement de plusieurs étapes. Premièrement, les spectres ont été centroïdés par le logiciel MassLynx de Waters Corporation. On applique alors la correction de masse de verrouillage. On compare dans ces résultats la correction entre une seule masse de verrouillage réelle et la méthode par VLM. Les points VLM utilisés sont décrits plus bas. La masse de verrouillage réelle utilisée est le clomiphène, avec une masse de 406.18594223 Da. Ensuite, un seuil d'intensité a été appliqué sur les spectres, où l'on n'a conservé que les pics dont l'intensité est supérieure à 500. Ce seuil a été choisi afin d'éliminer les pics de très faible intensité dus au bruit de fond. De plus, ce seuil est l'intensité correspondant à environ une collision de molécule d'une masse donnée avec le détecteur à chaque cycle de balayage dans le spectromètre de masse et donc représente une molécule ayant une certaine abondance.

Les données présentées ici représentent des échantillons provenant de plusieurs plaques et qui ont été acquis à différents moments et sur différentes calibrations de l'appareil. On retrouve cinq plaques différentes dans les données présentées ici. De plus, deux de ces plaques ont été décongelées une deuxième fois et leur spectre acquit sur une nouvelle calibration. On peut ainsi comparer si l'on corrige bien les spectres des mêmes échantillons acquis à des moments différents. Les échantillons ont été acquis sur plusieurs calibrations. La liste des plaques et des calibrations dans lesquelles ils ont été acquis est présentée à la table 1.5.

Seuls les résultats des VLM sur deux des combinaisons d'ions acquis et d'extractions seront montrés ici. Ces combinaisons seront les ions positifs de l'extraction à l'acétonitrile (ACNMeOH-pos) et les ions négatifs de l'extraction au chlorobutane (Chlbut-neg). Ces deux combinaisons ont, respectivement, six et sept points de correction par VLM. Les masses de ces points, c'est-à-dire les masses approximatives déterminées manuellement, sont présentées à la table 1.6.

On peut remarquer que, dans le cas de la condition ACNMeOH-positive, on a un point de correction de moins. De plus, ces points se retrouvent plus dans les basses masses du spectre comparé aux points de la condition Chlbut-négative. Les points ont été choisis de telle manière vu qu'un spectre dans la

TABLE 1.5 – **Liste des calibrations et des plaques utilisées dans le test de validation des VLM.** Chaque colonne est une journée différente d'acquisition, et le spectromètre fut recalibré entre chacune de ces acquisitions. Les plaques sont dénotées par un nombre. Les plaques 27-2 et 31-2 sont ainsi dénotées, car elles sont les répétitions des plaques 27 et 31 respectivement.

Calibration 1	Calibration 2	Calibration 3
27	31	27-2
30	34	31-2
39		

TABLE 1.6 – **Liste des points VLM utilisés.**

ACNMeOH-positif	Chlbut-négatif
73.0458	126.9045
145.1024	255.2313
267.0008	329.1945
369.3530	473.2903
577.5212	773.5882
857.7596	1029.1282
	1253.3818

condition ACNMeOH-positive a la majorité de ses pics dans des masses plus basses, dans la région de 50 Da à environ 300 Da. On a aussi une très petite quantité de pics avec une masse supérieure à 900 Da dans cette condition. Au contraire, la condition Chlbut-négative contient une grande quantité de pics tout au long de son spectre. Il faut donc des points de correction à masses plus élevées et couvrant une plus grande partie du spectre.

La performance de l'algorithme de correction a été évaluée à partir de graphiques d'alignement et des consensus qui se retrouvent dans les spectres. Pour ce faire, la méthodologie discutée plus haut a été appliquée. De plus, une étape de prétraitement a été ajoutée. Les spectres ont été alignés avec les algorithmes 2, ALIGNE, et l'algorithme 1, CONSTRUCTION REPÈRE, avec une distance maximale de 5 ppm. Cet alignement a été appliqué, séparément, dans le cas de la correction par masse de verrouillage réelle et dans le cas de correction par VLM. Cet alignement est appliqué simplement pour contrebalancer les petites erreurs comme celles observées à l'évaluation précédente et pouvoir trouver des pics de consensus.

La métrique utilisée à ce point pour évaluer la performance de l'algorithme 3, CORRECTION VLM MANUELS, utilisé en plus de l'algorithme 2, ALIGNE, est le nombre de pics présent dans tous les échantillons donnés. Pour déterminer cette métrique, on utilise la méthode des graphes d'alignement présentés plus tôt dans ce chapitre. On note seulement le nombre de pics consensus, c'est-à-dire présents dans chaque échantillon. Un inconvénient de cette approche est qu'on ne compte que les pics

présents dans *tous* les échantillons. Ainsi, si un pic est absent d'un seul spectre, il ne contribuerait pas au compte du consensus. Malgré cet inconvénient, on s'attend à voir un grand nombre de pics de consensus, puisque les échantillons utilisés sont des échantillons de plasma sanguin. Ainsi, on s'attend à voir un grand nombre de métabolites en communs dans tous les échantillons.

Au cours des prochains tableaux, les résultats de l'algorithme 3, CORRECTION VLM MANUELS, seront comparés à la méthode de correction par masses de verrouillage réelles. La méthode de correction par masses de verrouillage réelles fut appliquée comme suit. Un échantillon par plaque, soit celui en première position, a été exclu. À sa place, seule la solution contenant du clomiphène fut pipettée dans le puits. On sait que le pic associé au clomiphène est d'une masse de 406.18594223 *amu* (*arbitrary mass units*). On calcule alors la correction à appliquer afin que le pic de clomiphène se retrouve à son endroit connu et réel. Cette correction est ensuite appliquée à tous les pics de chacun des spectres de la plaque.

Plusieurs comparaisons ont été effectuées. Prenons en premier la comparaison des consensus de pics sur une seule plaque. Le consensus de pics est le nombre de pics qui est retrouvé dans chacun des spectres d'un ensemble Il est à noter que le consensus est donc sur 96 échantillons.

TABLE 1.7 – **Table de consensus de pics - Consensus à l'intérieur d'une plaque donnée.** RLM dénote le fait qu'une correction par masse de verrouillage réelle (*Real Lock Mass*) fut appliquée. VLM dénote l'application d'une correction par l'algorithme 3 ,CORRECTION VLM MANUELS. ACN-pos dénote la condition d'extraction à l'ACNMeOH et l'acquisition des ions positifs. Chl-neg dénote l'extraction au chlorobutane et l'acquisition des ions négatifs.

Plaque	ACN-pos, RLM	ACN-pos, VLM	Chl-neg, RLM	Chl-neg, VLM
27	252	403	226	255
30	2	430	186	204
31	36	335	208	204
34	0	634	14	242
39	0	479	88	163
27-2	65	535	247	365
31-2	0	438	127	289

Il y a plusieurs éléments à noter dans la table 1.7. Premièrement, on remarque une certaine variation dans les consensus des spectres corrigés par masse réelle. Alors que la plaque 27 a un consensus de 252 pics dans la condition ACNMeOH-positive, aucune autre plaque n'a un consensus de plus de 100. La plupart en ont même 0. Au niveau du cas Chlbut-négatif, on a plutôt un consensus typique d'environ 150-200 pics. La plaque 34 a quand même un consensus très faible de 14 pics. Il est à noter que les variations instrumentales qui causent les déviations à corriger peuvent arriver à l'intérieur même d'une plaque. Il semble plutôt que la plaque 27 soit une exception avec peu de déviations dans le cas ACNMeOH-postitif. Il est aussi à noter que les spectres de la condition Chlbut-négative ont généralement beaucoup plus de pics que les spectres ACNMeOH-positifs. Il est alors peu étonnant de

voir des consensus plus élevés.

En comparaison, lorsqu'on applique la correction par VLM, on a systématiquement un consensus de plus de 400 pics dans le cas des spectres ACNMeOH, même sur les plaques où l'on n'avait qu'un faible consensus avec la correction par masse réelle ou même aucun consensus. Au niveau du cas Chlbut-négatif, la correction par VLM apporte généralement un gain dans le consensus, mais plusieurs des plaques ont un consensus environ équivalent dans les deux méthodes de correction. On s'attend quand même à ce qu'il y ait moins de déviation sur les masses à l'intérieur d'une plaque, alors ce résultat n'est pas étonnant.

Un test plus poussé serait de comparer les performances des deux méthodes sur des plaques différentes. C'est ce qui sera montré dans les prochains tableaux. Il est à noter à ce point qu'en faisant le consensus de 2 plaques, on considère alors le consensus entre 192 pics.

TABLE 1.8 – **Table du nombre de consensus de pics - Consensus à l'intérieur de plaques de la même calibration.** RLM dénote la correction par masse de verrouillage réelle (*Real Lock Mass*) et VLM dénote par masse de verrouillage virtuelle. ACN-pos dénote la condition d'extraction à l'ACN-MeOH et l'acquisition des ions positifs. Chl-neg dénote l'extraction au chlorobutane et l'acquisition des ions négatifs.

Plaques	ACN-pos, RLM	ACN-pos, VLM	Chl-neg, RLM	Chl-neg, VLM
27/30	0	364	0	183
27/39	0	378	0	186
31/34	0	288	0	158
27-2/31-2	0	419	0	365

La table 1.8 indique donc les consensus entre deux plaques acquises sur une même calibration. Un premier élément important à noter est que, par la méthode de correction par masse réelle, tous les consensus sont de 0 pic. Il est donc évident ici que cette méthode est insuffisante par elle-même, car on sait que le plasma sanguin a de nombreuses molécules qui sont systématiquement présentes pour la simple survie de l'organisme.

On remarque aussi à la table 1.8 que la méthode de correction par VLM va fournir des consensus assez importants dans tous les cas. En particulier, dans la condition ACNMeOH-positive, on remarque que tous les nombres de consensus sont près de 300 pics ou plus. Dans le cas Chlbut-négatif, on remarque des consensus plus faibles, souvent dans la région de 150 à 200 pics. Ces nombres sont tout de même assez proches des nombres de consensus sur plaque individuelle dans la même condition et l'on s'attend à voir une dégradation du consensus plus on a d'échantillons (vu l'augmentation statistique des chances de voir des échantillons aberrants).

Étant donné que l'objectif de la méthode est de corriger les différences entre les calibrations, il est nécessaire de valider son efficacité dans ce cas. Le fait d'avoir acquis des spectres de certains échantillons à deux reprises nous est d'une grande aide à ce point et il semble judicieux de considérer ce cas

en particulier.

TABLE 1.9 – **Table du nombre de consensus de pics - Consensus à l'intérieur de plaques de calibrations différentes.** RLM dénote la correction par masse de verrouillage réelle (*Real Lock Mass*) et VLM dénote par masse de verrouillage virtuelle. ACN-pos dénote la condition d'extraction à l'ACN-MeOH et l'acquisition des ions positifs. Chl-neg dénote l'extraction au chlorobutane et l'acquisition des ions négatifs.

Plaques	ACN-pos, RLM	ACN-pos, VLM	Chl-neg, RLM	Chl-neg, VLM
27/27-2	0	374	0	230
31/31-2	0	231	0	196

On peut voir à la table 1.9 les résultats de consensus entre les plaques répétées. Encore une fois, on peut voir que la méthode de correction par masse de verrouillage réelle n'a pas fourni de consensus dans les deux conditions. Au contraire, si l'on considère la condition ACNMeOH-positive et la correction par VLM, on voit que des consensus assez importants sont présents dans les 192 échantillons. On a aussi des bons consensus dans le cas Chlbut-négatif. On remarque aussi que la comparaison des plaques 27 27-2 retrouvent de plus grands consensus que la comparaison entre les plaques 31 et 31-2. Plusieurs explications sont plausibles ici. Il est possible qu'un des réplicats (c.-à-d. acquérir les échantillons d'une même plaque une seconde fois) de la plaque 31 ait eu une moins bonne acquisition et ait moins de pics. Une autre possibilité est qu'une des acquisitions de la plaque 31 aurait pu être aussi plus divergente en termes de variations de masse. Finalement, il est aussi possible que les conditions lors de la calibration aient été plus similaires lors du moment de faire les calibrations 1 et 3, où l'on retrouve les plaques 27 et 27-2, que celles des calibrations 2 et 3 (plaques 31 et 31-2).

D'autres comparaisons entre les plaques acquises lors de calibrations différentes ont été faites. Ces données sont présentées dans la table A.2 en annexe.

TABLE 1.10 – **Table de consensus de pics - Consensus de pics entre des échantillons pigés aléatoirement.** RLM dénote la correction par masse de verrouillage réelle (*Real Lock Mass*) et VLM dénote par masse de verrouillage virtuelle. ACN-pos dénote la condition d'extraction à l'ACNMeOH et l'acquisition des ions positifs. Chl-neg dénote l'extraction au chlorobutane et l'acquisition des ions négatifs.

Nombre d'échantillons	ACN-pos, RLM	ACN-pos, VLM	Chl-neg, RLM	Chl-neg, VLM
100	0	382	44	177
200	0	342	30	153
300	0	321	0	148
400	0	292	0	129

Finalement, une dernière expérience avec les consensus a été faite. Ces résultats sont présentés à la table 1.10. Un nombre d'échantillons variable a été choisi aléatoirement sur les 7 plaques décrites précédemment. Cette expérience va donc refléter un peu mieux comment une calibration par masses

de verrouillage virtuelles peut améliorer les performances prédictives d'algorithmes d'apprentissage automatique.

On remarque donc que plus on augmente le nombre d'échantillons, plus le consensus diminue. Comme mentionné précédemment, on s'attend à voir cet effet. On observe aussi qu'en condition ACNMeOH-positive, il n'y a toujours aucun pic en consensus dans l'ensemble des échantillons lorsqu'on corrige par masse de verrouillage réelle. Il y a par contre un faible consensus lorsque l'on considère seulement 100 ou 200 échantillons en condition Chlbut-négative. On remarque aussi que, malgré que la taille du consensus diminue alors qu'on augmente le nombre d'échantillons tirés, la taille du consensus diminue à un rythme relativement lent (5 à 40 pics perdus par 100 échantillons additionnels, avec une moyenne d'environ 20).

En dernier lieu, revenons à la classification des échantillons provenant de deux plaques mentionnées précédemment, dans la section concertant la table 1.4. Si l'on reprend les mêmes échantillons que la première expérience et que l'on change seulement l'étape de correction en appliquant la correction par VLM avec les points de correction mentionnés dans les résultats précédents, nous obtenons les résultats de la table 1.11. Rappelons-nous que la tâche de prédiction est de classifier les calibrations source des spectres présentés à l'algorithme. Ainsi, si nous obtenons des résultats avec une précision élevée, il est aisé de différencier les spectres. Dans le cas contraire, soit une mauvaise performance du classificateur, c'est un indice qu'il est difficile de déterminer la provenance des spectres.

TABLE 1.11 – **Résultats d'apprentissage automatique avec correction par VLM sur les échantillons provenant de deux plaques.** La précision marque la performance du classificateur sur l'ensemble de test et indique le pourcentage de bonnes prédictions sur l'ensemble.

Distance d'alignement	Random Forest	Decision Tree	SVM Linéaire	SVM RBF	SCM
Alignement à 5 ppm	53.3%	53.3%	53.3%	53.3%	53.3%
Alignement à 10 ppm	53.3%	53.3%	53.3%	53.3%	53.3%
Alignement à 15 ppm	53.3%	46.7%	53.3%	53.3%	53.3%
Alignement à 20 ppm	53.3%	53.3%	53.3%	53.3%	53.3%

On remarque à la table 1.11 un changement important par rapport aux résultats de la correction par masse de verrouillage réelle (à la table 1.4). Alors que la prédiction de la provenance des exemples était très précise avec cette méthode de correction précédemment, il est impossible pour les algorithmes de classification de bien prédire la source des exemples après la correction par VLM. Il est important de mentionner que les résultats de la table sont presque entièrement une précision de 53.3%, car les classificateurs tentaient de prédire que tous les exemples provenaient d'une seule des deux plaques. Or, le tirage aléatoire des exemples de l'ensemble de test avait fourni un ensemble contenant 16 exemples d'une plaque et 14 de l'autre, donc 53% représente un classificateur qui retourne simplement la classe majoritaire. Il est à noter qu'il est tout aussi impossible pour les algorithmes parcimonieux que les non parcimonieux de prédire la provenance des exemples.

Ces résultats indiquent une bonne performance de l'algorithme de correction par VLM. En effet, le fait d'avoir de mauvaises performances avec un classificateur appris par apprentissage automatique indique qu'il est difficile, voire impossible, de faire la différence entre les classes. Dans ce cas, les classes correspondent à des calibrations différentes du spectromètre. Donc, la correction appliquée par l'algorithme de masses de verrouillage virtuelles semble être suffisante afin d'être incapable de distinguer les différentes calibrations lors de l'acquisition.

1.4 Amélioration des algorithmes

Au cours du projet, un article a attiré notre attention. Il s'agissait d'un article par Robert Tibshirani *et al* Tibshirani et collab. (2004). Dans cet article, une méthode d'alignement de pics pour la spectrométrie de masse de type MALDI-TOF (*Matrix-Assisted Laser Desorption/Ionisation - Time of Flight*) est proposée. Cette méthode utilise un regroupement hiérarchique (*hierarchical clustering*). Plus précisément, cette méthode utilise un regroupement hiérarchique à liens complet (*complete linkage*). Cette méthode de regroupement fonctionne en regroupant les points les plus près les uns des autres un après l'autre, étape par étape. Le cluster ainsi formé par deux points est alors représenté par la moyenne des points le formant. On itère ainsi sur les points jusqu'à ce que toutes les données soient regroupées dans un dendrogramme.

La méthode décrite dans cet article n'a pas été directement implémentée et utilisée. Par contre, elle a été incluse dans les algorithmes d'alignement et de correction par masses de verrouillage virtuelles. Plus précisément, cette méthode semble un moyen efficace de construire le spectre de repère pour l'alignement ou bien de détecter automatiquement quels points des spectres peuvent être utilisés comme point de correction VLM. Les adaptations des algorithmes pour inclure cette méthode seront maintenant décrites.

1.4.1 Algorithme de correction par masses de verrouillage virtuelles

Comme mentionné plus haut, la méthode par regroupement hiérarchique semble adaptée à la détection automatique de points de corrections VLM. Dans la première implémentation de cet algorithme, il n'y avait pas de détection automatique de ces points. Il fallait alors un long travail manuel afin de déterminer un certain nombre de points de corrections. Au contraire, une méthode automatique serait beaucoup plus efficace et apte à fournir un plus grand nombre de points de correction, améliorant la performance de l'algorithme.

On inclut le regroupement hiérarchique ainsi. Après un traitement des spectres, une filtration par intensité afin de conserver les pics de haute intensité, on regroupe les spectres dans un ensemble qui servira à déterminer les points de correction. On fait ensuite le regroupement hiérarchique complet de tous les pics dans l'ensemble des spectres. On détermine ensuite la distance maximale que l'on veut entre deux pics afin qu'ils puissent être considérés comme faisant partie du même cluster et donc comme le même point de correction. On "coupe" le dendrogramme à cette distance, c'est-à-dire que

l'on prend tous les clusters dont les deux pics les plus distants ont une distance inférieure à la distance maximale spécifiée.

On choisit ensuite les clusters que l'on juge pouvoir être des points de correction. Le critère déterminé pour cela est qu'un cluster doit contenir le même nombre de pics que le nombre de spectres utilisé afin de faire le regroupement hiérarchique. On a ainsi des clusters qui contiennent putativement un pic de chaque spectre. Cette hypothèse devrait être vraie tant que l'on choisit une distance assez basse, car il est très improbable de retrouver deux pics de haute intensité à de très courtes distances les uns des autres. La masse de référence pour le point de correction VLM sera alors la moyenne pondérée des masses des pics dans le cluster, où la pondération est sur le nombre d'occurrences d'une masse dans le cluster.

Algorithm 4 DÉTECTION_AUTOMATIQUE_VLM

Require: Un ensemble de m spectres, $S =< s_1, ..., s_m >$
Require: Chaque spectre est une liste de pics $s_i =< P_1 =< \mu_i^1, \iota_i^1 >, ..., P_n =< \mu_i^n, \iota_i^n >>$
Require: Une distance maximale de regroupement, d
  Créer une liste pour stocker toutes les masses
  for $i = 1$ to m **do**
    for $j = 1$ to n **do**
      Ajouter μ_i^j à la liste de masses
    end for
  end for
  Faire le regroupement hiérarchique complet
  Arrêter le regroupement à d
  Ajouter les clusters à la liste de points potentiels
  Créer une liste de points de correction
  for all clusters **do**
    if nombre de pics dans le cluster $= m$ **then**
      Ajouter la moyenne pondérée de la masse des pics dans le cluster à la liste des points de correction
    end if
  end forreturn la liste des points de correction

Décrivons en détail l'algorithme 4. On fait une liste contenant tous les pics de tous les spectres. C'est à partir de cette liste que sera bâti le regroupement hiérarchique.

Le regroupement sera arrêté lorsque l'algorithme sera rendu au point de regrouper ensemble un cluster où les deux points les plus distants sont à une distance supérieure à la distance maximale fixée. On ajoute ensuite ces clusters ainsi que les pics qui les composent à une liste des points de correction potentiels. La liste des points de correction potentiels sera alors filtrée. La filtration, comme mentionnée ci-dessus, sera faite sur le critère du nombre de pics formant le cluster. Si un cluster a le même nombre de pics le formant que le nombre de spectres utilisés, alors on considère que ce point est un point de correction VLM.

Un bénéfice de cette méthode est que, en conséquence de l'augmentation du nombre de points de

correction, on peut relâcher un critère précédemment établi pour la correction, c'est-à-dire que chaque spectre à corriger doit contenir tous les points de corrections. C'était auparavant nécessaire, car environ six à huit points de corrections étaient déterminés. Cette méthode de détection automatique peut par contre fournir plusieurs dizaines à plusieurs centaines de points de correction. On peut alors permettre, lors de l'application de la correction, qu'un spectre ne contienne pas tous les points. La correction des dérivations devrait tout de même être suffisante.

Des résultats d'évaluation de cette méthode suivent dans la section "Résultats".

1.4.2 Algorithme d'alignement

On peut faire un exercice similaire avec l'algorithme d'alignement. Un problème que l'on peut concevoir avec la première implémentation est que l'on parcoure le spectre sommé des plus petites masses vers les plus grandes (de gauche à droite) en regroupant les pics. L'approche par regroupement hiérarchique à lien complet enlève ce biais. Cette nouvelle approche va regrouper les pics les plus près dans le spectre en premier et puis les pics plus distants par la suite. Cette approche va donc prioriser le regroupement des pics près les uns des autres au lieu des pics à la "gauche" du spectre.

Algorithm 5 ALIGNE_HIÉRARCHIQUE

Require: Un ensemble de m spectres, $S =< s_1, ..., s_m >$
Require: Chaque spectre est représenté par une liste de pics contenant une masse μ et une intensité ι, $s_i =< s_i^1 =< \mu_i^1, \iota_i^1 >, ..., s_i^n =< \mu_i^n, \iota_i^n >>$
Require: Un paramètre de distance maximale à considérer, d
  Créer une liste pour stocker toutes les masses
  for $i = 1$ to m **do**
    for $j = 1$ to n **do**
      Ajouter le μ_i^j à la liste de masses
    end for
  end for
  Ordonner la liste de masses
  Faire le regroupement hiérarchique complet
  Arrêter le regroupement à d
  Calculer la moyenne pondérée de la masse du cluster à partir des pics le formant
  Ajouter cette masse à la liste des points de référence **return** la liste des points de référence

L'algorithme 5 décrit la méthode d'alignement par regroupement hiérarchique. Cet algorithme est environ le même que celui de détection automatique des points de correction VLM. La différence majeure est que, dans le cas de l'algorithme d'alignement, il n'y a pas à vérifier le nombre de pics qui forme chaque cluster. Vu que l'on cherche simplement à ramener les pics à l'intérieur de la distance maximale spécifiée, on ne se soucie pas de savoir si chacun des spectres comporte ce pic.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

1.4.3 Évaluation quantitative de la performance des algorithmes de correction et d'alignement

Jeu de données d'évaluation des nouveaux algorithmes Les nouvelles implémentations de correction et d'alignements ont été testées sur des nouvelles données. Les échantillons utilisés dans ce cas sont deux plaques de plasma sanguin. Après une décongélation, les échantillons ont été extraits dans un mélange de ACN :MeOH de ratio 75 :25. On prend 10 μL d'échantillon et 90 μL de ACN :MeOH. Les échantillons subissent ensuite une sonication de 5 minutes et une centrifugation de 5 minutes à 5000 RPM. Cette étape cause la précipitation des protéines. On récupère donc le surnageant afin de procéder. Ils ont par la suite été dilués dans une proportion 1 :10 afin d'éviter une suppression ionique durant l'acquisition.

Les spectres ont été acquis avec LDTD et sur un spectromètre Synapt G2. Le patron laser fut modifié. Le laser reste à intensité nulle pour les 2 premières secondes. Il s'ouvre ensuite et monte à 65% de sa force maximale sur 6 secondes. Il reste ensuite à cette intensité pour 3 secondes avant de se fermer. Les spectres sont ensuite acquis en mode MS^e, soit un mode d'acquisition data-indépendant. On acquiert le spectre en deux modes dans ce cas. On acquiert une première fonction qui consiste des molécules ionisées qui circulent dans le spectromètre et sont détectées. On acquiert par la suite une seconde fonction où l'on rajoute une énergie de collision dans la cellule de collision du spectromètre, après la sélection des ions par le quadripôle. Cette énergie va causer de la fragmentation dans les molécules ionisées. On va donc avoir deux spectres par échantillon acquis, contenant respectivement les ions mères et les ions filles. Cette acquisition a été faite en mode *high resolution*. Seuls les spectres de la première fonction ont été utilisés dans cette expérience.

Les spectres acquis ont ensuite été prétraités par le logiciel MassLynx de Waters Corporation afin de faire la centroïdation des pics. Une fois fait, on a séparé les spectres. 12 spectres, choisis aléatoirement, ont été retirés du jeu de données afin de servir comme spectres de test. Les 180 autres échantillons ont subi une filtration par seuil d'intensité de 500, afin d'enlever les pics de trop faible intensité qui sont dus au bruit de fond du spectromètre. Cet ensemble de 180 spectres a servi d'ensemble d'entraînement pour les tests. Pour les tests de correction par VLM, ce sont ces 180 spectres qui ont servi d'ensemble de spectres afin de trouver les points de correction. Ils ont aussi servi d'ensemble d'entraînement pour construire le spectre de référence pour l'algorithme d'alignement. Plus de détails à ce sujet suivront aux sections de résultats respectives à ces algorithmes.

Trois jeux de données de tests des algorithmes ont été générés à partir des spectres retirés. Dans chacun des trois cas, on a pris un des 12 spectres au hasard. Le spectre choisi a ensuite subi une filtration à un seuil d'intensité de 500, pour la même raison que les spectres d'entraînement. On a ensuite ajouté une étiquette à chacun des pics de ce spectre entre les masses de 100 Da et 800 Da.

Pour des fins d'expérimentation sur nos techniques de correction, des perturbations sur les pics du spectre ont ensuite été ajoutées *in silico*. Ces perturbations peuvent être de trois types. Le premier est un décalage uniforme dans tout le spectre. On spécifie une distance en ppm et une direction (gauche

ou droite, soit diminuant la masse ou augmentant la masse) pour cette variation. Ce type de variation nous permet d'émuler *in silico* la dérivation due à un changement de calibration ou de moment d'acquisition entre des échantillons. Le deuxième type de variation induite est une variation d'une magnitude et d'une direction aléatoire sur chacun des pics individuels. Les variations de masse étaient d'un maximum de 5 ppm. On a donc une probabilité de distribution uniforme pour chaque pic qu'il y ait une variation de -5 ppm à +5 ppm. Cette variation permet de simuler l'erreur de lecture et de centroïdation aléatoire sur les pics.

Le troisième type de variation est une variation dans l'intensité du pic. Pour chaque pic, on tire une direction aléatoire (augmentation ou diminution) et un multiplicateur aléatoire. Les variations maximales étaient d'un facteur 2. On a donc une probabilité uniforme sur chaque pic que l'intensité de ce pic soit multipliée par un facteur entre 0.5 et 2. Ce type de variation n'affecte pas la masse, mais pourrait tout de même affecter la correction par VLM, vu que l'on cherche à corriger selon le pic le plus fort dans une fenêtre. Cette variation va donc tester la robustesse de l'algorithme par rapport à l'erreur de lecture sur l'intensité.

Il est à noter que les variations aléatoires de 5 ppm sur la masse et d'un facteur 2 sur l'intensité sont des variations extrêmement fortes. Expérimentalement, une erreur sur la masse de 5 ppm ou d'un facteur 2 sur l'intensité représente les extrêmes de ce qui a été observé dans des données réelles.

TABLE 1.12 – Table des perturbations apportées dans les jeux de données 1 et 2.

Spectre	Modifications	Jeux de données 1	Jeux de données 2
1	Aucune	-	-
2	Décalage uniforme	10 ppm, droite	10 ppm, droite
3	Décalage uniforme	10 ppm, gauche	10 ppm, gauche
4	Variations aléatoires de masse	-	-
5	Décalage uniforme et variations aléatoires de masse	8 ppm, droite	8 ppm, droite
6	Décalage uniforme et variations aléatoires de masse	8 ppm, gauche	8 ppm, gauche
7	Variations aléatoires de masse et intensité	-	-
8	Variations aléatoires de masse et intensité	-	-
9	Décalage uniforme et variations aléatoires d'intensité	6 ppm, droite	6ppm, droite
10	Décalage uniforme et variations aléatoires d'intensité	6 ppm, gauche	6 ppm, gauche
11	Décalage uniforme, variations aléatoires de masse et d'intensité	5 ppm, droite	5 ppm, droite
12	Décalage uniforme, variations aléatoires de masse et d'intensité	5 ppm, gauche	5 ppm, gauche

Décrivons un peu les perturbations apportées aux deux premiers jeux de données.

— Le spectre 1 reste sans perturbation. Ce sera notre point de référence pour l'évaluation des algorithmes. Si les algorithmes arrivent à replacer les pics des autres spectres au même endroit que ceux du spectre 1, alors ils sont performants, car ils peuvent compenser toute perturbation qu'un spectre subit.

— Les spectres 2 et 3 subissent seulement un décalage uniforme sur toute leur longueur, dans un sens différent. Le spectre 2 contient un décalage à « droite », c.-à-d. qui augmente les masses des pics. Le spectre 3 va plutôt contenir un décalage vers la gauche, donc les masses de ses pics sont uniformément plus petits que ceux du spectre 1. On s'attend à ce qu'il n'y ait plus d'erreurs dans ces spectres après seulement la correction par VLM.

— Le spectre 4 subit seulement des variations aléatoires de masse. L'alignement devrait arriver à replacer le spectre.

— Les spectres 5 et 6 subissent à la fois un décalage uniforme et des variations aléatoires. La correction et l'alignement seront nécessaires pour replacer les spectres.

— Les spectres 7 et 8 subissent à la fois les variations aléatoires de masse et d'intensité. L'alignement devrait replacer les pics de ces spectres puisque les variations d'intensité n'influeront pas sur l'algorithme d'alignement.

— Les spectres 9 et 10 subissent le décalage uniforme et les variations d'intensité. Ces spectres, ainsi que les spectres 11 et 12, seront un test plus exigeant pour la correction par VLM.

— Finalement, les spectres 11 et 12 ont les trois types de perturbations. Ils seront les tests les plus durs pour les algorithmes.

Finalement, le jeu de donnée 3 est sur un modèle différent des deux premiers. Ses modifications sont présentées à la table 1.13.

TABLE 1.13 – Table des perturbations apportées dans le jeu de données 3.

Spectre	Modifications	Paramètres
1	Aucune	-
2	Décalage uniforme	7 ppm, droite
3	Décalage uniforme et variations aléatoires de masse	7 ppm, droite
4	Décalage uniforme, variations aléatoires de masse et d'intensité	7 ppm, droite
5	Décalage uniforme	5 ppm, gauche
6	Décalage uniforme et variations aléatoires de masse	5 ppm, gauche
7	Décalage uniforme, variations aléatoires de masse et d'intensité	5ppm, gauche

Encore une fois, dans le jeu de donnée 3, le spectre 1 est le spectre de base non modifié. Dans le cas de ce jeu de données, les spectres 2, 3 et 4 forment un continuum, comme les spectres 5, 6 et 7. Le spectre 2 subit un décalage uniforme. Le spectre 3 subit ensuite le même décalage et une variation aléatoire de la masse de ses pics. Finalement, le spectre 4 a le même décalage, subit exactement les mêmes variations de masses que le spectre 3 et a en plus une variation de l'intensité de ses pics. Les spectres 5, 6 et 7 ont le même traitement, mais avec un décalage différent et les variations de masses et intensité sont différentes de celles des spectres 3 et 4 (mais la variation des masses des spectres 6 et 7 est identique). On peut ainsi constater l'effet plus spécifique des perturbations sur un spectre après l'application des algorithmes de correction VLM et d'alignement. Par exemple, on s'attend à voir exactement les mêmes masses dans les spectres 3-4 et 6-7 après l'application de la correction VLM, si la correction est robuste au bruit dans l'intensité des pics.

Les résultats présentés ci-dessous sont de deux types. Le premier est l'erreur moyenne au carré (*mean squared loss*) sur la distance en ppm entre les pics d'un spectre perturbé et du spectre de référence pour le jeu de données. Ce score est calculé en prenant la masse d'un pic dans le spectre perturbé, en la soustrayant à la masse de son pic homologue dans le spectre de référence. On transforme ensuite la différence obtenue en ppm et on la met au carré afin d'éviter d'avoir des distances négatives. Le score indiqué est la moyenne de ce score sur chaque pic évalué dans le spectre, les pics évalués étant tous les pics dont la masse est entre 100 Da et 800 Da dans le spectre original.

La deuxième métrique calculée sur la correction est la perte zéro-un (*zero-one loss*). On regarde simplement si, après l'application des corrections, on retrouve chaque pic dans le spectre de référence et dans le spectre perturbé au même endroit, à plus ou moins 10^{-4}, soit la précision de l'appareil sur la masse. On donne un score de 0 si c'est le cas ou de 1 si les pics ont des masses différentes. On fait ensuite la moyenne sur le nombre de pics. Ce score reflète donc la proportion d'erreurs dans les masses d'un spectre perturbé par rapport au spectre de référence.

Correction par masses de verrouillage virtuelles

Considérons d'abord les résultats de la correction par VLM sur les différents jeux de données. Pour cette expérience, les points de correction VLM ont été choisis par l'algorithme sur l'ensemble de 180 spectres d'entraînement. La correction a ensuite été appliquée sur tous les spectres du jeu de données. On a finalement comparé les spectres perturbés par rapport au spectre 1 du jeu de données évalué, afin de constater la performance de l'algorithme à replacer les pics perturbés aux endroits attendus sans perturbations. Le paramètre de distance maximale a été fixé à 40 ppm sur tous les jeux de données.

La table 1.14 contient les résultats de l'évaluation de la correction par VLM sur le premier jeu de données. On remarque tout d'abord que les spectres 2, 3, 9 et 10, qui étaient les spectres subissant un décalage uniforme de masse et pas de variations aléatoires de masse, sont remarquablement bien corrigés. Dans ces quatre spectres, l'erreur moyenne au carré est 0.005 ou moins, donc à une erreur moyenne de ± 0.07 ppm ou moins. La perte 0-1 de ces pics est dans la région de 7 à 8 %. Vu la petite

TABLE 1.14 – **Évaluation de la correction VLM sur le jeu de données 1.** Le nombre indiqué dans la case est l'erreur moyenne au carré, en ppm, des pics du spectre. Le nombre entre parenthèses est la perte 0-1, c'est-à-dire la proportion de pics erronés dans le spectre perturbé.

Spectre	Sans correction	Correction VLM
1	0 (0%)	0 (0%)
2	100.118 (100%)	0.005 (8.6%)
3	100.118 (100%)	0.005 (8.7%)
4	8.351 (96.0%)	11.877 (79.8%)
5	72.872 (100%)	11.600 (80%)
6	71.408 (100%)	11.445 (80%)
7	8.264 (96.4%)	11.858 (79.8%)
8	8.228 (96.7%)	11.821 (79.7%)
9	36.007 (100%)	0.004 (7.4%)
10	36.007 (100%)	0.003 (7.3%)
11	33.120 (98.4%)	11.343 (80.3%)
12	33.549 (98.2%)	11.629 (80.2%)

taille de l'erreur moyenne, les erreurs dans ces cas sont très possiblement simplement dues à des erreurs lorsqu'on arrondit les masses au 0.0001 Da le plus proche. Les spectres 5, 6, 11 et 12 comportent tous à la fois un décalage uniforme et des variations aléatoires sur leurs masses. On observe qu'après la correction, on obtient des erreurs moyennes au carré très similaire entre les spectres 5, 6, 11 et 12 et les spectres 4, 7 et 8 qui n'avaient que les variations aléatoires. Cela indique donc encore une fois que la correction performe très bien. On remarque aussi dans les spectres 9 et 10 que la correction a bien fonctionné malgré les variations d'intensité induites.

TABLE 1.15 – **Évaluation de la correction VLM sur le jeu de données 2.** Le nombre indiqué dans la case est l'erreur moyenne au carré, en ppm, des pics du spectre. Le nombre entre parenthèses est la perte 0-1, c'est-à-dire la proportion de pics erronés dans le spectre perturbé.

Spectre	Sans correction	Correction VLM
1	0 (0%)	0 (0%)
2	100.303 (100%)	0.003 (4.9%)
3	100.303 (100%)	0.004 (5.5%)
4	8.462 (96.3%)	10.042 (64.3%)
5	71.837 (100%)	9.331 (64.2%)
6	73.991 (100%)	9.708 (64.7%)
7	8.284 (96.4%)	9.322 (63.8%)
8	8.513 (96.0%)	9.749 (64.4%)
9	35.999 (100%)	0.002 (4.5%)
10	35.999 (100%)	0.003 (4.6%)
11	32.451 (98.1%)	8.961 (64.3%)
12	33.337 (98.5%)	9.754 (64.2%)

On remarque dans la table 1.15 les mêmes tendances que pour le jeu de données 1. Les spectres ayant seulement subi un décalage uniforme au niveau des masses (2, 3, 9 et 10) ont des erreurs moyennes au carré extrêmement faibles et ont même des pertes 0-1 de 5% ou moins. Dans les cas où l'on a induit un décalage et une variation aléatoire des masses (spectres 5, 6, 11 et 12), l'erreur est encore une fois ramenée au même niveau que les spectres où seule une variation aléatoire de masse a été ajoutée (spectres 4, 7 et 8). Finalement, les résultats des spectres 9 à 12 indiquent encore une fois que la correction semble robuste aux changements d'intensité.

TABLE 1.16 – **Évaluation de la correction VLM sur le jeu de données 3.** Le nombre indiqué dans la case est l'erreur moyenne au carré, en ppm, des pics du spectre. Le nombre entre parenthèses est la perte 0-1, c'est-à-dire la proportion de pics erronés dans le spectre perturbé.

Spectre	Sans correction	Correction VLM
1	0 (0%)	0 (0%)
2	49.002 (100%)	0.003 (3.3%)
3	56.321 (100%)	9.071 (62.9%)
4	56.321 (100%)	9.071 (62.9%)
5	25.052 (100%)	0.002 (3.1%)
6	32.269 (97.9%)	8.588 (62.4%)
7	32.269 (97.9%)	8.588 (62.4%)

Finalement, considérons les résultats du jeu de données 3 présentés à la table 1.16. On remarque encore que l'erreur est extrêmement faible pour les spectres 2 et 5, qui n'avaient subi que le décalage uniforme. Dans le cas de ce jeu de données, l'erreur moyenne au carré et la perte 0-1 sont très faibles. On observe aussi que les spectres 3 et 4 ont la même perte avant et après la correction. On remarque la même chose entre les spectres 6 et 7. Ces résultats indiquent donc que l'algorithme de correction est robuste à la variation en intensité.

Alignement

Pour tester l'algorithme d'alignement, les mêmes jeux de données ont été utilisés. De plus, les résultats montrés ici compareront les spectres sans correction ni alignement, les spectres avec alignement et sans correction, et les spectres avec alignement et correction. Les erreurs seront évaluées de manière similaire aux résultats précédents, avec l'erreur moyenne au carré de la distance en ppm entre les pics homologues et la perte 0-1 en parenthèses. Les paramètres étaient les mêmes dans tous les jeux de données. Le paramètre maximal de distance pour la correction VLM était de 40 ppm, ce qui correspond à la même valeur que celle utilisée pour générer les résultats précédents. La distance maximale d'alignement a été fixée à 15 ppm.

Une différence est à noter entre les résultats sur la correction par VLM et l'alignement. La métrique la plus importante pour la correction par VLM est l'erreur carrée moyenne sur la masse des pics, car on recherche à rapprocher les spectres bruités le plus près possible, mais une légère erreur est encore

tolérable. Par contre, au niveau de l'alignement, la perte 0-1 est l'évaluation la plus importante. L'alignement cherche à regrouper les pics homologues ensemble afin que les pics se retrouvent exactement au même endroit avant d'analyser les spectres par apprentissage automatique. Il est alors important que les pics se retrouvent à exactement la même masse (précise à 4 décimales d'uma dans ce cas). S'il y a erreur, la différence entre la masse du pic d'un spectre et de son pic homologue dans son second spectre n'est pas importante puisqu'ils ne sont pas à la même masse exacte.

TABLE 1.17 – **Évaluation de l'alignement sur le jeu de données 1.** Les résultats sont présentés pour les spectres bruités ; alignés, mais non corrigés par VLM ; et corrigés VLM et alignés. L'erreur moyenne au carré de la distance, en ppm, des pics homologues est montrée. Le nombre entre parenthèses est la perte 0-1, c'est-à-dire la proportion de pics erronés dans le spectre perturbé.

Spectre	Bruité	Aligné seulement	Corrigé et aligné
1	0 (0%)	0 (0%)	0 (0%)
2	100.118 (100%)	103.228 (62.5%)	0.114 (1.9%)
3	100.118 (100%)	103.380 (54.1%)	0.249 (1.9%)
4	8.351 (96.0%)	23.727 (28.3%)	27.739 (29.5%)
5	72.872 (100%)	81.962 (53.0%)	27.504 (29.5%)
6	71.408 (100%)	81.132 (47.1%)	27.321 (29.7%)
7	8.264 (96.4%)	23.373 (28.0%)	25.444 (29.2%)
8	8.228 (96.7%)	21.532 (27.3%)	25.630 (29.0%)
9	36.007 (100%)	55.714 (44.4%)	0.092 (1.8%)
10	36.007 (100%)	57.286 (40.8%)	0.232 (1.4%)
11	33.120 (98.4%)	47.086 (39.2%)	25.379 (29.2%)
12	33.549 (98.2%)	49.397 (37.0%)	25.943 (29.6%)

Considérons les résultats de la table 1.17. On remarque très rapidement qu'il est meilleur d'appliquer la correction et l'alignement. Dans les spectres où l'on retrouve un simple décalage (2, 3, 9 et 10), on remarque que l'alignement seul ne performe pas très bien. On y retrouve également des erreurs moyennes au carré très grandes, souvent plus grandes que celles du spectre bruité sans correction. Au niveau de la perte 0-1, on remarque certains progrès (100% d'erreur à environ 40 à 60% d'erreur), mais cette proportion d'erreurs semble encore problématique. Si l'on compare ces mêmes spectres avec le cas où les spectres sont corrigés et alignés, on remarque que la conjonction des deux méthodes corrige beaucoup mieux, avec des pertes 0-1 de l'ordre de 1% à 2% d'erreur. Sur les spectres comportant à la fois un décalage et une perturbation aléatoire induite dans les masses, on observe que l'erreur moyenne au carré ainsi que la perte 0-1 sont ramenées au même niveau que l'erreur sur les spectres ayant seulement eu une perturbation aléatoire de masse. Dans tous les cas, cette erreur est relativement élevée sur ce jeu de données, avec environ 29% d'erreur.

On voit également que dans les cas où il y a seulement une variation aléatoire dans les masses, l'alignement va augmenter l'erreur moyenne au carré. Par contre, comme mentionné précédemment, la perte 0-1 est la métrique plus importante au niveau de l'alignement. La perte 0-1 est plus faible avec alignement que si l'on applique seulement la correction par VLM. On observe donc une meilleure

performance sur la métrique d'intérêt, mais une dégradation des résultats sur la métrique secondaire lorsqu'on applique l'alignement. Il est probable que l'erreur augmente puisque, lorsqu'il y a erreur dans l'alignement, l'algorithme pourrait avoir tendance à éloigner les pics homologues. Il est aussi à noter que si l'algorithme rencontre un pic dans un spectre à corriger qui n'a pas de pic de repère, la masse du pic à aligner sera laissée intouchée. La perte 0-1 élevée en général dans le jeu de données 1 est possiblement due à une forte proportion de pics retrouvés dans ce spectre n'ayant pas d'équivalent dans le spectre de repère.

TABLE 1.18 – **Évaluation de l'alignement sur le jeu de données 2.** Les résultats sont présentés pour les spectres bruités ; alignés, mais non corrigés par VLM ; et corrigés VLM et alignés. L'erreur moyenne au carré de la distance, en ppm, des pics homologues est montrée. Le nombre entre parenthèses est la perte 0-1, c'est-à-dire la proportion de pics erronés dans le spectre perturbé.

Spectre	Bruité	Aligné seulement	Corrigé et aligné
1	0 (0%)	0 (0%)	0 (0%)
2	100.303 (100%)	94.323 (56.3%)	0.070 (0.2%)
3	100.303 (100%)	86.003 (41.2%)	0.513 (0.6%)
4	8.462 (96.3%)	20.036 (14.8%)	23.179 (15.6%)
5	71.837 (100%)	78.091 (45.2%)	22.374 (15.1%)
6	73.991 (100%)	69.123 (34.6%)	22.790 (14.7%)
7	8.284 (96.4%)	18.319 (14.2%)	21.216 (15.4%)
8	8.513 (96.0%)	21.165 (14.6%)	23.675 (15.1%)
9	35.999 (100%)	49.285 (33.6%)	0.062 (0.3%)
10	35.999 (100%)	48.729 (27.1%)	0.161 (0.2%)
11	32.451 (98.1%)	41.262 (27.7%)	21.591 (14.4%)
12	33.337 (98.5%)	40.017 (22.1%)	23.396 (15.1%)

La table 1.18 présente les résultats d'alignement pour le jeu de données 2. On remarque les mêmes tendances que le jeu de données 1. Sur les spectres 2, 3, 9 et 10, qui n'ont que des décalages uniformes, on obtient une perte 0-1 de moins de 1% avec la correction et l'alignement. On voit aussi que les erreurs, en termes de perte 0-1 et d'erreur moyenne au carré, sont ramenées aux mêmes niveaux entre les spectres ayant seulement des perturbations de masse aléatoire (spectres 4, 7 et 8) et les spectres ayant un décalage uniforme suivi de perturbations aléatoires (spectres 5, 6, 11 et 12). On remarque aussi que les erreurs sont plus faibles sur le jeu de données 2 que sur le jeu de données 1 en général, suggérant que le spectre de base du jeu de données 1 était possiblement plus différent comparé à l'ensemble d'entraînement sur lequel le spectre de repère a été bâti.

Finalement, la table 1.19 présente les résultats de l'évaluation de l'alignement sur le jeu de données 3. Les spectres 2 et 5 n'avaient subi que le décalage uniforme de masses. Ces deux spectres sont raisonnablement alignés avec un alignement simple, avec des pertes 0-1 de, respectivement, 36% et 22%. On voit par contre que l'application de la correction et de l'alignement a résulté en un alignement parfait entre les spectres 1 et 2. De plus, le spectre 5 a également une perte 0-1 de moins de 1% lorsqu'on le compare au spectre 1. On remarque aussi des améliorations si l'on fait la correction et l'alignement

TABLE 1.19 – **Évaluation de l'alignement sur le jeu de données 3.** Les résultats sont présentés pour les spectres bruités ; alignés, mais non corrigés par VLM ; et corrigés VLM et alignés. L'erreur moyenne au carré de la distance, en ppm, des pics homologues est montrée. Le nombre entre parenthèses est la perte 0-1, c'est-à-dire la proportion de pics erronés dans le spectre perturbé.

Spectre	Bruité	Aligné seulement	Corrigé et aligné
1	0 (0%)	0 (0%)	0 (0%)
2	49.002 (100%)	54.547 (36.5%)	0 (0%)
3	56.321 (100%)	56.898 (36.5%)	24.196 (14.4%)
4	56.321 (100%)	56.898 (36.5%)	24.196 (14.4%)
5	25.052 (100%)	37.094 (22%)	0.116 (0.2%)
6	32.269 (97.9%)	37.887 (21.5%)	19.135 (12.3%)
7	32.269 (97.9%)	37.887 (21.5%)	19.135 (12.3%)

sur les spectres 3/4 et 6/7. Dans les deux cas, on obtient de bonnes performances d'alignement, avec seulement 12% à 14% d'erreur. Encore une fois, le taux d'erreur plus faible sur ce jeu de données semble indiquer que le jeu de données 1 avait des pertes anormalement élevées.

1.5 Conclusion

En conclusion, plusieurs méthodes de traitement de données de spectres de masse ont été mises au point au cours du projet. De plus, ces méthodes ont été améliorées et évaluées sur plusieurs contextes.

Premièrement, une nouvelle méthode de correction des spectres a été développée, l'algorithme des masses de verrouillage virtuelles (montré aux algorithmes 3 et 4). Cette méthode se base sur le fait que l'on a des échantillons contenant une structure métabolique stable et prévisible en général. Une limitation de cette méthode est qu'elle est dépendante des échantillons et pourrait ne pas être applicable à d'autres types d'échantillons. Notons cependant que cette méthode est applicable sur les spectres d'échantillons sanguins et sur d'autres types d'échantillons ayant une richesse de pic suffisante. De plus, de multiples évaluations de la performance de cette méthode témoignent qu'elle est efficace et performante.

Plusieurs éléments restent encore à développer sur cette méthode de correction par VLM. Premièrement, on utilise une méthode d'interpolation linéaire par morceaux afin de faire la correction des masses. Même si cette méthode s'est révélée performante, il est possible que d'autres méthodes de calcul de facteurs de corrections puissent être aussi bonnes ou supérieures. La faiblesse principale de l'approche par interpolation linéaire est au niveau des facteurs de correction pour les masses se retrouvant avant et après les premiers et derniers points de correction VLM. La méthode utilisée à ce point utilise le facteur de correction du premier ou dernier point appliqué uniformément. D'autres méthodes de calculs pourraient tenter d'extrapoler des facteurs pour ces masses, tel que si l'on calculait les facteurs à partir d'une spline cubique. Des tests pourraient être faits dans le futur afin de vérifier la

performance par cette méthode.

De plus, il est difficile de fixer les paramètres de distance de l'algorithme de correction de manière intuitive. Fixer de mauvais paramètres peut causer une dégradation de la performance de la correction et rendre l'algorithme moins robuste aux variations d'intensité en particulier si l'on fixe la distance comme étant trop large. Des travaux futurs porteront donc sur l'optimisation automatique du paramètre de distance sur un jeu de données.

Deuxièmement, une méthode d'alignement des pics adaptée au LDTD fut aussi mise au point. Cette méthode a été évaluée et est performante. En conjonction avec l'algorithme de correction VLM, cette méthode permet de rendre des spectres de masse acquis à de multiples moments très comparables en termes de masses et ainsi pouvoir préparer les échantillons convenablement afin de pouvoir les utiliser en apprentissage automatique sans avoir de fort biais.

L'alignement a aussi certaines limites. Le paramètre de distance maximale peut être complexe à optimiser. Une valeur trop basse résultera avec un mauvais alignement et des données difficiles à comparer. À l'opposé, une valeur excessivement grande créera de faux positifs et regroupera des pics qui ne sont pas, en réalité, homologues. Une solution est de considérer ce paramètre comme un hyperparamètre additionnel en apprentissage automatique, mais sa validation augmentera le temps de calcul.

Quelques travaux futurs sont toujours possibles pour l'algorithme d'alignement. Même s'il est assez performant, on remarque que plusieurs erreurs surviennent encore lorsqu'on aligne de nouveaux spectres à un spectre repère bâtis sur un ensemble d'autres spectres. Il pourrait être possible d'améliorer les performances en explorant de nouvelles avenues que le regroupement hiérarchique ou une méthode itérant sur les pics des spectres (similairement à la première version de l'algorithme). On pourrait aussi tester des centres alternatifs pour les clusters de pics au lieu de moyenne pondérée afin de constater si cela amènerait une performance accrue.

De plus, on peut considérer l'application des ces algorithmes à d'autres cas et à d'autres instruments de spectrométrie de masse. Les données de toutes les expériences présentées dans ce chapitre ont été générées avec un Synapt-G2 Si de Waters Corporation. C'est un spectromètre de masse de très haute qualité, avec un excellente résolution sur la masse des pics. Dans le cas de spectromètres à moins bonne résolution, il est possible que l'on se retrouve dans une situation où les variations entre les calibrations et l'incertitude sur les pics seront de plus grand ordre. Il serait donc intéressant d'appliquer ces algorithmes dans de nouveaux contextes afin de s'assurer de leur généralité dans le domaine.

Chapitre 2

Application de l'apprentissage automatique

2.1 Introduction

Ce chapitre sera consacré aux résultats découlant de l'application de l'apprentissage automatique aux données de spectrométrie de masse. Il contient aussi les résultats d'une nouvelle méthode qui a été développée au cours de cette maîtrise.

Méthodes à noyau

Il faut par contre introduire les différents algorithmes utilisés. Un type d'algorithme qui fut utilisé au cours des travaux présentés ici est la classe des méthodes à noyau. Ces méthodes furent utilisées malgré le fait qu'elles ne sont pas parcimonieuses et ni interprétables, ce qui est une des caractéristiques recherchées dans ce projet. Par contre, les méthodes à noyau sont aisément adaptées aux problèmes auxquelles elles sont appliquées. Dans ce type de méthode, l'élément principal est la fonction de noyau (*kernel function*) qui projette les exemples d'apprentissage dans une dimensionnalité supérieure afin de pouvoir mieux séparer les classes d'exemples Boser et collab. (1992). Un exemple de ce procédé est présenté à la figure 2.1.

On cherche à avoir une fonction qui permet de faire le produit scalaire entre les vecteurs de caractéristiques des exemples sans avoir en mémoire les vecteurs de caractéristiques dans l'espace de dimensionnalité supérieure (que l'on dénote usuellement ϕ), ce qui est une caractéristique additionnelle des méthodes à noyau. On cherche en fait à obtenir directement la matrice de taille n par n, où n est le nombre d'exemples dans l'ensemble d'entraînement. On nomme cette matrice la matrice de Gram. C'est en se basant sur cette matrice que les algorithmes utilisant les méthodes à noyau peuvent faire de la classification. Cette matrice représente en fait une mesure de similarité entre les exemples, c'est-à-dire que si $k(x,x')$ est grand alors les deux exemples x et x' sont similaires et proches dans l'espace du noyau et si $k(y,y')$ est petit, ou même négatif dans certains cas, alors les exemples y et y' sont

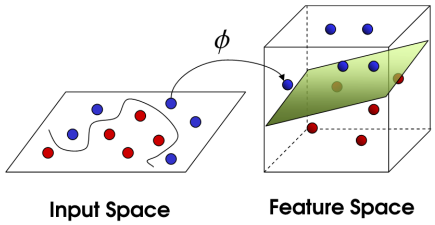

FIGURE 2.1 – **Exemple de méthode à noyau.** On projette les exemples comportant deux dimensions dans un espace de dimensionnalité supérieure, soit trois dans cet exemple, afin de pouvoir y tracer un séparateur linéaire. Ce séparateur devient non linéaire dans l'espace source des exemples.

dissimilaires et donc éloignés dans l'espace du noyau.

$$k(x,x') = \phi(x) \cdot \phi(x') \tag{2.1}$$

L'algorithme utilisant les méthodes à noyau le plus répandu est l'algorithme des machines à vecteur de support (*Support Vector Machine*, SVM) Cortes et Vapnik (1995); Chang et Lin (2011); Fan et collab. (2008). Cet algorithme cherche à tracer un séparateur linéaire entre les exemples de l'ensemble d'entraînement des différentes classes. Un séparateur linéaire est un séparateur ayant une dimension de moins que l'espace des caractéristiques des exemples. Par exemple, un séparateur linéaire est une ligne droite dans le cas où les exemples ont deux dimensions ou caractéristiques. Un séparateur linéaire est un hyperplan à deux dimensions dans le cas où les exemples ont trois dimensions pour les décrire. Par contre, l'algorithme du SVM tente de tracer le séparateur linéaire dans l'espace du noyau fourni. Comme un noyau est de plus haute dimensionnalité des données, le séparateur devient non-linéaire dans l'espace de base des exemples. On peut voir un exemple de cet effet dans les figures 2.1 et 2.2.

De plus, l'algorithme du SVM cherche à maximiser la marge du séparateur avec les exemples. La notion de marge est une notion importante pour minimiser les risques de surapprentissage. L'algorithme cherche à trouver le séparateur linéaire qui divise les deux classes dans l'espace du noyau et qui maximise la marge, soit la distance entre le séparateur et les exemples qui en sont le plus près. Le séparateur se retrouvera donc à une distance égale entre au moins un exemple de chacune des deux classes. Un exemple visuel est présenté dans la figure 2.2. Les exemples qui se retrouvent à la distance minimale du séparateur et qui définissent le séparateur sont appelés les vecteurs de support.

Un avantage que comportent le SVM et les algorithmes utilisant les noyaux est qu'il est possible de choisir différents noyaux plus adaptés au problème étudié, voire d'en mettre au point de nouveaux. Par contre, cet algorithme présente aussi un inconvénient. Alors que, dans le projet présent, les algorithmes plus parcimonieux sont privilégiés, l'algorithme du SVM n'est pas du tout parcimonieux. Le fait d'utiliser un noyau fait qu'il faut considérer toutes les caractéristiques d'un exemple et la classification va en fait dépendre de la comparaison d'un nouvel exemple avec les vecteurs de support. Malgré cet

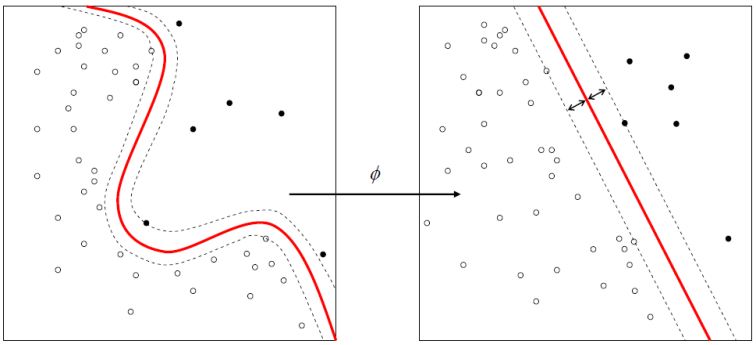

FIGURE 2.2 – **Exemple d'un SVM sur un ensemble de données.** On remarque que, sur l'image de droite, qui représente l'espace projeté du noyau, le séparateur est linéaire. Il devient non-linéaire dans l'espace des caractéristiques à gauche. On remarque aussi la marge qui place le séparateur entre des exemples des deux classes différentes dans l'image de droite.

inconvénient, cette méthode est répandue et performante, en plus d'avoir l'avantage d'être flexible. Elle a donc été incluse dans les tests de ce projet.

2.1.1 Machine à couverture d'ensembles

L'algorithme qui a été le plus privilégié pour ce projet est sans doute l'algorithme de la machine à couverture d'ensembles (*Set Covering Machine*, abbrévié SCM) Marchand et Taylor (2002). La force de cet algorithme et la raison pour laquelle il est privilégié dans ces travaux est qu'il met l'accent sur la parcimonie. Cet algorithme est très parcimonieux au niveau du nombre de caractéristiques des exemples utilisés pour la prédiction, ce qui rend la classification facilement interprétable. Le SCM cherche à classifier les exemples en utilisant des règles simples. L'exemple le plus simple de ces règles est la souche de décision, qui a été utilisée au cours de ce projet avec le SCM. Une souche de décision est une règle où l'on prend une caractéristique d'un exemple et on y applique un seuil. On obtient la forme $x_i > z$ où z est une valeur soit observée ou arbitraire. x_i dénote la caractéristique i d'un exemple x. Le SCM cherche donc la règle ayant la plus grande *utilité* sur les exemples de l'ensemble d'entraînement. L'utilité est définie comme le nombre d'exemples négatifs qui sont bien classés moins le nombre d'exemples positifs qui sont mals classés par cette règle, pondérée par un paramètre p qui est déterminé par validation croisée.

$$U_h = |Q_h| - p \cdot |R_h|,$$

Où U_h est l'utilité de la règle h, Q_h est le nombre d'exemples négatifs bien classifiés et R_h est le nombre d'exemples positifs mal classés. Lorsque la règle optimale a été trouvée, le SCM va ensuite retirer tous les exemples négatifs qui sont bien classés et les exemples positifs mal classés de l'ensemble d'entraînement, avant de faire une autre itération. L'algorithme peut s'arrêter sous deux conditions : soit il n'y a plus d'exemples négatifs à classifier, soit l'algorithme atteint le maximum de règles permises, qui

est aussi un paramètre déterminé par validation croisée. Le fait d'avoir un nombre maximal de règles permises est aussi une manière de réguler le surapprentissage.

Finalement, les règles retenues par l'algorithme sont combinées de deux façons possibles. La première façon est par la conjonction, c.-à-d., le prédicteur ainsi produit est le prédicteur booléen consistant en la conjonction de toutes les règles retenues. La seconde façon consiste à produire une disjonction de toutes ces règles. Une disjonction correspond à un ou exclusif en logique. Par exemple, afin de classifier un exemple par un ensemble de règles disjointes, cet exemples doit répondre à une et une seule de ces règles.

L'algorithme du SCM a donc un avantage clair pour le projet décrit ici qui est sa parcimonie et son interprétabilité. La parcimonie du SCM résulte en un très petit nombre de caractéristiques, soit des pics dans un spectre de masse dans notre cas, qui établit un lien prédictif entre les classes. De plus, l'algorithme fournit une organisation logique entre les caractéristiques utilisées par la classification. Effectivement, les deux types d'organisations logiques des règles du SCM fonctionnent très bien dans un contexte biologique. Une conjonction de caractéristiques indique que plusieurs molécules sont nécessaires afin de faire la différence entre les deux classes, ce qui est un cas courant en biologie. Le cas d'une disjonction est aussi très compatible. De trouver un classificateur ayant une disjonction entre deux pics par exemple pourrait indiquer que deux voies métaboliques distinctes sont impliquées dans la différence entre les classes.

2.1.2 Arbres de décision

Un autre algorithme utilisé dans ce projet est l'algorithme des arbres de décision (*decision trees*) Breiman et collab. (1983). Plus précisément, les arbres de classification sont utilisés ici. L'algorithme d'apprentissage par arbre de décision fonctionne en séparant les données par une série de règles du même type que celles utilisées pour le SCM. On peut en fait voir les souches de décisions, mentionnées dans le cadre de l'algorithme du SCM, comme des arbres de décision avec une seule règle, soit une profondeur de 1. Un exemple d'un de ces arbres est présenté à la figure 2.3. L'algorithme des arbres de décision va ainsi faire un arbre de la forme suivante. On commence à la racine de l'arbre. On a une première décision basée sur une caractéristique des exemples, de la forme $x_i > z$. Selon le résultat de la comparaison, on suit l'arc correspondant dans l'arbre. On peut ensuite arriver à un nouveau noeud. Si c'est le cas, on répète le processus avec la règle encodée dans ce noeud. Si l'on arrive à une feuille, c.-à-d. un noeud terminal sans arc sortant, cette feuille indique dans quelle classe est l'exemple considéré.

Comme tous les autres algorithmes, il y a un risque de surapprentissage avec cet algorithme. On peut facilement former un arbre avec trop de noeuds, trop profond et qui ne généralisera pas à d'autres exemples. Pour pallier ce problème, on utilise la validation croisée pour déterminer les meilleurs paramètres pour l'algorithme. Les paramètres disponibles dans l'implémentation de cet algorithme qui fut utilisé sont la profondeur maximale de l'arbre et le nombre d'exemples minimum à séparer par noeud

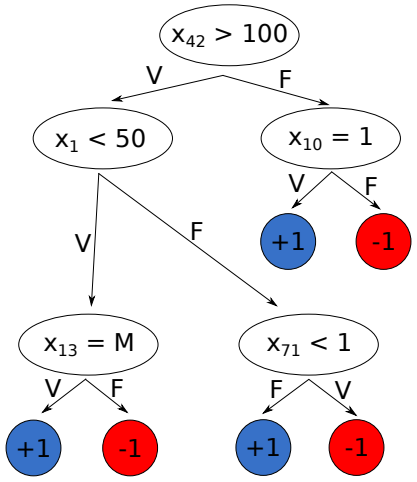

FIGURE 2.3 – **Exemple de la structure d'un arbre de décision.** Les noeuds représentent des règles de décision. Selon si elle est respectée ou pas dans un exemple, on suit le chemin vers le prochain noeud. Les noeuds colorés sont les feuilles, qui représentent la classe majoritaire des exemples se retrouvant dans cette feuille.

Pedregosa et collab. (2011). On utilise donc la validation croisée afin de trouver une combinaison de ces paramètres qui sera performante sur l'ensemble d'entraînement et sera capable de généraliser.

De manière semblable au SCM, on peut voir l'arbre de décision comme un mélange plus complexe de disjonctions et conjonctions de règles. L'algorithme des arbres de décision est aussi un algorithme avantageux pour le projet étant donné qu'il est aussi très parcimonieux et interprétable, surtout comparé aux méthodes à noyaux ou aux réseaux de neurones. L'arbre de décision a par contre tendance à être moins parcimonieux que le SCM. De plus, sa capacité à organiser les règles de décision de manière plus complexe peut l'aider dans certains cas ou rendre le classificateur moins interprétable.

2.1.3 Forêt d'arbres décisionnels

Un dernier algorithme d'apprentissage utilisé au courant de ce projet est celui des forêts d'arbres décisionnels, parfois appelés forêts d'arbres aléatoires, ou *random forest* Breiman (2001). Cet algorithme fait partie de la classe d'algorithmes d'apprentissage que l'on nomme les méthodes d'ensemble. Les méthodes d'ensemble sont des méthodes qui tentent d'obtenir une meilleure performance en prédiction en utilisant un vote de majorité de plusieurs autres algorithmes. Dans le cas des forêts d'arbres décisionnels, un vote de majorité est fait sur un ensemble d'arbres de décisions.

L'algorithme génère un nombre d'arbres de décisions. Les arbres sont générés sur un sous-ensemble d'exemples, tirés par méthode *bootstrap*, c'est-à-dire en pigeant aléatoirement des exemples avec

remise, dans l'ensemble d'entraînement. Ensuite, au moment de calculer les règles à chaque noeud, l'algorithme va prendre un sous-ensemble aléatoire des caractéristiques des exemples et l'utiliser pour choisir la règle. On entraîne ainsi un nombre d'estimateurs spécifié. Pour faire une prédiction, on montre l'exemple à classifier à chacun des arbres estimateurs, qui vont chacun prédire une classe. On fait ensuite un vote de majorité de ces classificateurs avec un poids uniforme, c'est-à-dire que chacun des votants a un poids équivalent aux autres.

Les forêts d'arbres décisionnels sont utilisées dans ce projet pour les raisons suivantes. Une première est que c'est un algorithme populaire et généralement performant. Il est aussi très rapide à entraîner. Cet algorithme a aussi une tendance assez faible à surapprendre et son design est fait pour diminuer le surapprentissage des arbres le composant. Un inconvénient par rapport aux arbres de décision et au SCM est que cet algorithme n'est pas très parcimonieux et n'est pas du tout interprétable.

2.2 Le noyau à boîtes chevauchantes pour l'algorithme du SVM

2.2.1 Motivation et preuve de noyau

Au cours du projet, l'idée d'un nouveau noyau adapté aux défis du travail avec des données de spectromètre de masse est revenue à plusieurs reprises. Finalement, une idée s'est imposée. Elle est inspirée d'une méthode déjà existante en traitement de données de spectrométrie de masse. Cette technique se nomme le *binning*. Simplement, cette méthode consiste à compenser pour le désalignement des pics de plusieurs spectres en regroupant plusieurs pics dans une distance spécifiée en un seul, que l'on nomme un *bin*, ou une boîte. Un exemple est présenté à la figure 2.4. Une manière simple de l'implémenter est d'arrondir d'une ou deux décimales les masses en uma mesurées par un spectromètre et de sommer les pics qui se retrouvent à la même masse.

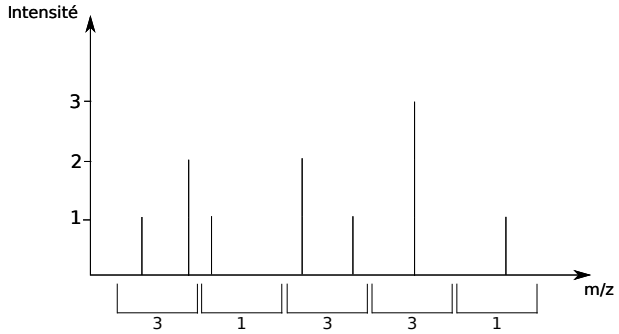

FIGURE 2.4 – **Exemple de la méthode du binning.** Si l'on considère un spectre donné, on regroupe et additionne les pics retrouvés dans un *bin*, ou une boîte.

Certains problèmes sont présents dans cette méthode. Un premier est la possibilité qu'un pic homologue entre deux spectres se retrouve dans des bins différents. En reprenant l'exemple du paragraphe précédent, on peut imaginer un pic x_1 de masse 201.1234 Da qui ait une déviation dans un autre spectre x_2 avec une masse de 201.1239 Da. Le fait d'arrondir pourrait amener la masse de x_1 à 201.123 Da et

celle de x_2 à 201.124 Da. Un autre problème avec l'approche par arrondissement est que les variances aléatoires d'un spectre sont relatives à la masse et donc on aura une déviation plus grande avec de plus grandes masses. Cet inconvénient peut être pallié si l'on imagine les bins comme étant d'une taille fixe en ppm, mais il restera tout de même le fait que des pics peuvent se retrouver d'un bord et de l'autre de la frontière entre deux bins.

C'est donc dans cette optique que nous avons eu l'idée de mettre au point un nouveau noyau pour l'algorithme du SVM basé sur l'idée des bins. Pour régler le problème des pics sur la frontière entre les bins, un nouveau paramètre de chevauchement a été introduit. Dans ce noyau, on représente donc un spectre par une série de boîtes d'une taille constante en ppm, donc dont la taille va augmenter à travers le spectre lorsque la masse augmente, et qui se chevauchent partiellement. Un exemple de cette représentation est montré à la figure 2.5.

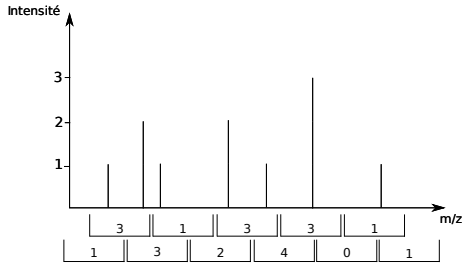

FIGURE 2.5 – **Exemple de la méthode par boîtes chevauchantes (*overlapping bins*).** On considère ici le même spectre qu'à la figure précédente. On considère aussi des boîtes de la même taille, et un chevauchement de 50%. On remarque qu'on a les mêmes boîtes qu'à la figure précédente. En plus, on regroupe dans les nouvelles boîtes des pics qui étaient auparavant séparés par la frontière d'une boîte, malgré une petite distance entre ces pics.

Afin d'être utilisable avec les méthodes à noyau tel que l'algorithme du SVM, il faut faire la preuve que le nouveau noyau, baptisé le noyau à boîtes chevauchantes (*Overlapping Bin Kernel*, abbrévié OBK), est effectivement un noyau. La caractéristique principale des noyaux est présentée à l'équation 2.1. Dans le cas de l'OBK, la preuve est simple. Avec les paramètres de taille de bin et de chevauchement, on peut décrire explicitement le vecteur de caractéristique dans le noyau, puisqu'on connaît le point de départ (la masse la plus petite que l'on a acquise dans l'ensemble des spectres), la taille de chaque boîte et le chevauchement entre chaque boîte. Ainsi, si l'on sait exactement la forme du vecteur $\phi(x)$ pour tout x et donc que $\phi(x')$ aura la même forme pour tout x', il s'ensuit directement que pour $k(x,x') = \phi(x) \cdot \phi(x')$, la mesure de similarité k est donc bien une fonction noyau (c.-à-d., une fonction représentant un produit scalaire dans l'espace vectoriel des caractéristiques). Il peut donc être utilisé par l'algorithme du SVM.

Il est à noter qu'une conséquence de l'utilisation de ce noyau est qu'il n'est plus nécessaire d'appliquer l'algorithme d'alignement décrit au chapitre précédent. Il est aussi théoriquement possible que ce noyau rende redondante l'application de masses de verrouillages, réelles ou virtuelles. Par contre, il faut utiliser des tailles de boîtes beaucoup plus grandes afin de compenser pour les déviations corrigées

par masses de verrouillage. Cela entraîne un risque que la taille de boîte soit trop grande et plus assez spécifique pour servir de bonne fonction de comparaison. En général, la taille des boîtes et le chevauchement seront des hyperparamètres qui seront choisis par validation croisée afin d'éviter un surapprentissage.

Il est aussi à noter que, malgré que le fait que les objectifs de ce projet favorisent l'application d'algorithmes parcimonieux et dont le modèle est interprétable, cette méthode est utilisée en conjonction avec l'algorithme du SVM et n'est donc pas du tout parcimonieuse ou interprétable. Par contre, l'objectif plus général du projet est de classifier des échantillons de produits sanguins selon leur spectre de masse. Il semble alors qu'il ne faut pas laisser passer une opportunité de mettre au point une nouvelle méthode adaptée aux données et donc prometteuse pour la classification, même si elle ne répond pas à toutes les caractéristiques privilégiées.

2.2.2 Présentation de l'algorithme

De premiers tests ont été pratiqués avec une première version très directe et peu optimisée de l'algorithme, afin de vérifier si l'approche avait un bon potentiel. L'algorithme 6 présente une partie commune à la première implémentation et aux versions subséquentes, c'est-à-dire la construction de la matrice de Gram entre les différents exemples. Cette matrice est la matrice de comparaison entre les échantillons, nécessaire et utilisée par l'algorithme du SVM.

Algorithm 6 Construction de la matrice de Gram

Require: Un ensemble de m spectres
  Créer une matrice de taille $m \times m$, la matrice de Gram
  for $i = 0$ to m **do**
    for $j = i$ to m **do**
      Comparer le spectre i et le spectre j avec le noyau à boîtes chevauchantes
      Stocker la valeur de noyau retournée aux indices $[i,j]$ et $[j,i]$
    end for
  end for
  return La matrice de Gram

La méthode de construction de matrices de Gram présentée à l'algorithme 6 est standard. Il faut construire une matrice de comparaison entre toutes les paires d'exemples, où chaque exemple est un spectre. On construit donc un tableau de taille $m \times m$ pour stocker les valeurs du noyau lorsqu'on a m spectres. On parcourt ensuite les exemples, en faisant toutes les comparaisons. Notez qu'Algorithme 6 est sous-optimal, le fait de faire toutes les comparaisons n'est pas la meilleure façon de calculer le noyau OBK. Nous allons d'abord proposer une façon légèrement plus efficace (et surtout mieux détaillée) de faire ce calcul (Algorithme 7) que nous améliorerons à la sous-section suivante. En fait, pour accélérer la méthode, on ne calcule que les valeurs pour la moitié supérieure du tableau, puisque la comparaison entre un exemple x et y sera la même qu'entre y et x. Cela permet d'accélérer cette étape d'un facteur près de deux. L'algorithme 7 va donc présenter la première implémentation du noyau.

Algorithm 7 Première implémentation du noyau à boîtes chevauchantes

Require: Deux spectres, s_1 et s_2
Require: Un paramètre de taille de boîtes, en ppm, σ
Require: Un paramètre de chevauchement, ω
Require: Une masse de départ et une masse finale, μ_d et μ_f
▷ La masse de départ est déterminée par la masse minimale de l'acquisition des spectres. La masse finale doit être déterminée sur l'ensemble des spectres utilisés. Il nous faut donc la masse de plus grande taille observée dans tous les spectres.
for $i = 1$ to 2 **do**
Créer un vecteur contenant les valeurs de boîtes v_i
Trouver les pics de s_i dans la boîte $b_1 = [\mu_d, \mu_d + \mu_d \cdot \sigma]$ et ajouter la somme de leurs intensités au vecteur v_i
while $\mu_f \notin b_j$ **do**
$b_j = [\mu_j, \mu_j + \mu_j \cdot \sigma]$ où $\mu_j = \mu_{i-1} + (\mu_{i-1} \cdot \sigma) \cdot \omega$
Trouver les pics de s_i dont la masse est dans la boîte b_j et ajouter la somme de leurs intensités au vecteur v_i
end while
end for ▷ On a à ce point les vecteurs des valeurs des boîtes des deux spectres
▷ La valeur à tout indice donné j dans chacune de ces listes équivaut à la même boîte dans les deux spectres.
return $v_1 \cdot v_2$

Décrivons ici l'algorithme 7 en plus de détail. Une première spécification à apporter est au niveau des masses de départ et de fin des boîtes. Il faut que, dans toutes les comparaisons de spectres, on ait exactement les mêmes boîtes définies. Étant donné que les boîtes correspondent aux éléments du vecteur $\phi(x)$, il nous faut absolument que tous les ϕ soient identiques au niveau de leur composition. Si l'on définissait les boîtes selon les plus petites masses et plus grandes masses observées dans les deux exemples d'une comparaison, on aurait des vecteurs différents à chaque comparaison et donc nous n'aurions pas un noyau.

Ensuite, on crée la première boîte et l'on somme tout pic se trouvant dans cette boîte. Si aucun ne s'y trouve, on confère une valeur de 0 à cette boîte. On cherche les pics selon une recherche binaire où les bornes sont les masses limites de la boîte. Cela fonctionne puisque nous gardons la liste des pics ordonnée par leurs masses. De plus, la recherche binaire s'opère de manière efficace, avec un pire cas de $O(\log n)$ où n est le nombre de pics dans le spectre. Une fois le contenu de la boîte sommé, on stocke cette somme en mémoire.

Après qu'on ait terminé la première boîte, on avance à la prochaine. On peut calculer la masse de départ de la nouvelle boîte à partir de la taille et du chevauchement. On répète alors la même procédure qu'à la première boîte à cette nouvelle boîte et toutes les boîtes subséquentes. Le critère d'arrêt est que l'on atteigne une boîte qui inclut dans sa fenêtre la masse finale générale aux exemples.

Lorsque ce procédé a été fait sur les deux spectres à comparer, on fait le produit scalaire entre les deux vecteurs de caractéristiques. On itère sur chacune des caractéristiques des vecteurs et on multiplie

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

chacune des boîtes homologues ensemble avant de sommer tous les résultats. On a ainsi le produit scalaire entre les vecteurs. Ce produit est la mesure de similarité entre les deux spectres.

Cette implémentation contient plusieurs inconvénients. Le premier est d'ailleurs que l'on explicite le vecteur ϕ des exemples en mémoire. Cela requiert une grande quantité de mémoire vive et de temps de calcul afin d'ajouter des éléments et en temps d'accès. De plus, la complexité de l'algorithme dépend fortement, dans cette implémentation, du nombre de boîtes nécessaire pour couvrir les spectres, donc essentiellement des paramètres de taille et de chevauchement. Un autre problème de cette implémentation est qu'il faut constamment aller chercher les pics dans le spectre. Même avec un algorithme de recherche efficace tel que la recherche binaire, une recherche est nécessaire pour chaque boîte. Plusieurs seront d'ailleurs inutile si aucun pic ne se trouve dans la boîte cible.

Plusieurs améliorations doivent donc être apportées à l'algorithme afin d'accélérer son fonctionnement, puisque les premiers tests se sont avérés prometteurs. Une première amélioration était au niveau de l'implicitation du vecteur de caractéristiques. Au lieu de créer un vecteur avec b entrées où b est le nombre de boîtes, on crée un dictionnaire vide. Si une boîte ne contient aucun pic, on ne crée pas d'entrée dans le dictionnaire. Si une boîte contient des pics, on somme alors leur intensité et on stocke la valeur dans le dictionnaire, avec la clé d'accès étant les bornes de la boîte. Cette amélioration est avantageuse au niveau de la mémoire utilisée et du temps de calcul pour la gestion de la mémoire. Étant donnée que les dictionnaires en Python sont des tables de hashage [1], on a alors un temps d'accès constant et un temps d'insertion constant en moyenne (en l'absence de collisions). Cette amélioration est aussi bénéfique pour l'étape finale du produit scalaire. En effet, vu le fait qu'un grand nombre de boîtes auront généralement une valeur de 0, nous n'avons pas à nous soucier de leur contribution au produit scalaire. De plus, il faut que les deux boîtes homologues des deux spectres comparés soient non-nulles pour que leur partie du produit scalaire puisse contribuer. On itère alors sur les clés d'un des dictionnaires. Si cette clé n'est pas présente dans le dictionnaire de l'autre spectre, alors on sait que le second spectre n'a pas de pics dans la boîte homologue.

Une autre amélioration à l'algorithme est liée au fait qu'il est possible de construire des équations permettant de calculer dans quelles boîtes se retrouvera un pic d'une masse donnée, connaissant le point de départ des boîtes, la taille des boîtes et le chevauchement. Cette amélioration va donc permettre de rendre la complexité de l'algorithme du noyau dépendant plus fortement du nombre de pics dans un spectre, qui est typiquement inférieur au nombre de boîtes, tout dépendant des paramètres de taille t de chevauchement. L'algorithme devient alors O(n) où n est le nombre de pic dans le spectre à comparer.

1. Une table ed hashage est une structure de données de base. Sa caratéristique principale qui nous intéresse pour cette application est son temps d'accès optimal ainsi que son temps d'insertion très rapide en moyenne. Une table de hashage est représentée en mémoire par un long vecteur, et on "hash" la clé par une fonction afin d'obtenir l'index de la case où se retrouvera l'entrée. Pour plus d'information, consultez tout manuel de base de structures de données.

Amélioration mathématique du noyau à boîtes chevauchantes

Soit $\lambda_1 \in \mathbb{R}$, la masse la plus petite que l'on considère. On considère que chaque boîte est représentée par $(\lambda_i, \varepsilon_i)$ où $\lambda_i, \varepsilon_i \in \mathbb{R}$. Elles sont ordonnées en ordre crossant de λ. On définit aussi k comme étant $k = \frac{\sigma}{10^6}$, où σ est la taille en ppm de la boîte. On a aussi δ qui est défini comme $\delta = 1 - \omega$, où ω est le chevauchement entre les boîtes désiré.

$$\begin{aligned} \lambda_i &= \lambda_{i-1} + k \cdot \delta \\ &= \lambda_{i-1}(1 + k\delta) \end{aligned}$$

On pose alors que $c = 1 + k\delta$. On obtient alors que :

$$\begin{aligned} \lambda_i = \lambda_{i-1} \cdot c \quad \forall i \in \{2,3,4,...\} \subseteq \mathbb{N} \\ \lambda_i = \lambda_1 \cdot c^{(i-1)} \end{aligned} \tag{2.2}$$

On retrouve donc l'équation (2.2) qui détermine simplement la masse de départ λ_i de toute boîte i. On va donc maintenant avoir un moyen de calculer la fin de chaque boîte.

$$\begin{aligned} \varepsilon_i &= \lambda_i + k \cdot \lambda_i \\ &= \lambda_i(1 + k) \end{aligned}$$

$$\varepsilon_i = \lambda_1 \cdot c^{i-1} \cdot (1 + k) \quad \forall i \in \{1,2,3,...\} \subseteq \mathbb{N} \tag{2.3}$$

L'équation (2.3) nous indique alors la borne supérieure en masse de chaque boîte i. Nous cherchons par contre un moyen de calculer dans quelle boîte(s) sera un pic étant donné sa masse. Il nous faut donc supposer que l'on a une masse μ d'un pic quelconque et que $\mu \in \mathbb{R}$. Alors, on cherche à trouver les boîtes où μ peut se retrouver. On commence par déterminer dans quelles boîtes μ est supérieur à λ_i.

$$\begin{aligned} \mu &\geq \lambda_i \\ &\geq \lambda_1 \cdot c^{i-1} \end{aligned}$$

$$\begin{aligned} \log_c(\mu) &\geq \log_c[\lambda_1 \cdot c^{(i-1)}] \\ &\geq \log_c(\lambda_1) + \log_c(c^{(i-1)}) \\ &\geq \log_c(\lambda_1) + (i-1) \end{aligned}$$

$$\log_c(\mu) - \log_c(\lambda_1) + 1 \geq i \tag{2.4}$$

On fait le même exercice avec les masses finales, afin d'avoir les indices i des boîtes dont la masse finale est plus grande que la masse μ.

$$\begin{aligned}\mu &\leq \varepsilon_i \\ &\leq \lambda_1 \cdot c^{(i-1)} \cdot (1+k)\end{aligned}$$

$$\begin{aligned}\log_c(\mu) &\leq \log_c[\lambda_1 \cdot c^{(i-1)} \cdot (1+k)] \\ &\leq \log_c(\lambda_1) + \log_c(c^{(i-1)}) + \log_c(1+k) \\ &\leq \log_c(\lambda_1) + i - 1 + \log_c(1+k)\end{aligned}$$

$$\log_c(\mu) - \log_c(\lambda_1) - \log_c(1+k) + 1 \leq i \tag{2.5}$$

Avec les équations (2.4) et (2.5), on peut alors déterminer les indices i de toutes les boîtes dont la masse μ fera partie.

$$1 + \log_c(\mu) - \log_c(\lambda_1) - \log_c(1+k) \leq i \leq 1 + \log_c(\mu) - \log_c(\lambda_1) \tag{2.6}$$

On peut simplifier l'équation (2.6) à la forme suivante :

$$\log_c(\mu) + b_{low} \leq i \leq \log_c(\mu) + b_{high} \tag{2.7}$$

On définit les constantes b_{low} et b_{high} à l'équation (2.7) aux formes suivantes :

$$b_{low} = 1 - \log_c(\lambda_1) - \log_c(1+k)$$

$$b_{high} = 1 - \log_c(\lambda_1)$$

On considère que b_{low} et b_{high} sont des constantes puisqu'ils dépendent simplement de la masse de départ, de la taille des boîtes et du chevauchement. Tous ces paramètres sont fixés au moment de calculer le noyau et ne varient pas pour toute la matrice de Gram. Il est donc possible de précalculer les valeurs de c, k, λ_1 et donc de b_{low} et b_{high} au début de l'algorithme.

L'application de cette forme de recherche améliore grandement l'efficacité de l'algorithme, lorsqu'appliquée sur des spectres. Si l'on prend b comme étant le nombre de boîtes nécessaire pour couvrir le spectre et n comme étant le nombre de pics par spectre, la première implémentation de l'algorithme avait une complexité de l'ordre de $O(b \cdot \log(n))$ pour faire la comparaison entre deux spectres. Avec la

méthode décrite ci-dessus, on a une complexité de l'ordre de O(n) par comparaison. Étant donné que le nombre de boîtes peut devenir beaucoup plus grand que le nombre de pics par spectre, on a un très bon gain de performance.

La nouvelle implémentation de l'algorithme, décrite à l'algorithme 8, inclut les changements décrits plus haut.

Algorithm 8 Implémentation améliorée du noyau à boîtes chevauchantes

Require: Deux spectres, s_1 et s_2
Require: Un paramètre de taille de boîtes, en ppm, σ
Require: Un paramètre de chevauchement, ω
Require: Une masse de départ, λ_1
  Calculer les constantes c, b_{low}, b_{high}
  for $i = 1$ to 2 **do**
    Créer le vecteur v_i
    for $j = 1$ to n où $n = |s_i|$ **do**
      Calculer dans quelle(s) boîte(s) se retrouve le pic $p_i j$ à partir de sa masse $\mu_i j$
      Ajouter l'intensité $\iota_i j$ du pic dans chacune des boîtes où il se retrouve
    end for
  end for
  return $v_1 \cdot v_2$

Décrivons en détail les changements dans l'algorithme 8. Un premier changement notoire est qu'il n'y a pas de masse finale prédéfinie. Étant donné la nouvelle méthode mathématique, il n'y a pas nécessairement de boîte maximale à définir. Comme mentionné plus haut, on remarque aussi que l'on itère sur les pics du spectre lorsque l'on calcule le contenu des différentes boîtes au lieu d'itérer sur les boîtes mêmes. En utilisant l'équation (2.7), on obtient simplement l'index (ex : la i^{eme} boîte, $\forall i \in \mathbb{N}$) de toutes les boîtes dans lequel le pic se retrouve. Cela donne lieu d'ailleurs à un second changement. On dénote les boîtes dans le dictionnaire avec leur index au lieu de leurs masses frontière. Il serait aisé de calculer les masses frontières à partir des équations (2.2) et (2.3), mais cela n'est pas nécessaire au fonctionnement du noyau et ne nous fournirais aucune information supplémentaire.

Avec ces changements, on remarque une amélioration marquée de la performance. Alors que la première implémentation prenait un temps de calcul d'environs 3h30 (environ 210 minutes) afin de construire la matrice de Gram de 96 exemples, la seconde implémentation peut comparer 185 exemples en moins de 15 minutes [2]. La performance de la nouvelle implémentation est aussi beaucoup moins dépendante des paramètres de taille et de chevauchement que la première version, même si ces paramètres ont encore une influence. Effectivement, le paramètre de taille va faire augmenter le temps de calcul à mesure que la taille diminue, car il y aura plus de boîtes à mettre à jour lors du calcul du vecteur de caractéristiques et il y aura plus de caractéristiques sur lesquelles itérer au moment du produit scalaire. On remarque aussi que le temps de calcule augmente plus il y a de chevauchement.

2. Ces temps de calcul correspondent au temps nécessaire sur un noeud de calcul du supercalculateur Colosse, ayant un processeur Intel Xeon X5560 Nehalem de 2.8 GHz.

De manière comparable à la taille, plus il y a de chevauchement, plus il faut mettre de boîtes à jour par pic. Similairement, le produit scalaire sera plus long à faire puisque plus de boîtes seront non nul. Il faut aussi noter que le chevauchement fait rapidement augmenter le nombre de boîtes nécessaire afin de couvrir le spectre.

Une détail est à noter à propos de l'utilisation de l'algorithme et de ses paramètres. Il semble plus équitable d'utiliser des valeurs répondant à la forme suivante pour ce paramètre :

$$\omega = 1 - \frac{1}{\alpha} \quad \forall \alpha \in \{1,2,3,...\} \subseteq \mathbb{N}$$

La raison est la suivante. On considère par exemple le cas où l'on a un chevauchement de 50%. Cela répond au critère puisque $50\% = 0.5 = 1 - \frac{1}{2}$. On peut voir que, si chaque nouvelle boîte commence à la moitié de la boîte précédente, alors tout endroit dans le spectre couvert sera simultanément dans deux boîtes (à l'exception de la première moitié de la première boîte et deuxième moitié de la deuxième boîte). Un exemple de cela est montré à la figure 2.6. Donc, tout pic sera représenté deux fois dans le spectre. Il en est de même pour toute autre valeur α qui fait partie de l'ensemble $\{1,2,3,...\} \subseteq \mathbb{N}$, qui aura chaque pic représenté dans α boîtes à tout moment.

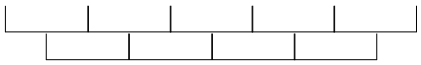

FIGURE 2.6 – **Exemple de boîtes chevauchantes dont le paramètre de chevauchement est à 50%.** On remarque que chaque endroit du spectre est inclus dans les limites de deux boîtes à tout moment, sauf aux extrémités.

Prenons par contre un exemple où le chevauchement a une valeur de 33%. On peut voir un exemple de l'arrangement des boîtes dans ce cas est présenté à la figure 2.7. Si l'on a une boîte de départ et que l'on ajoute seulement la boîte suivante avec 33% de chevauchement alors on peut remarquer dans la figure que certaines zones seront couvertes par deux boîtes, mais d'autres zones auront seulement une boîte pour les couvrir (dans leur centre). Il semble donc que d'avoir une couverture inégale à travers tout le spectre pourrait induire des biais dans la valeur du noyau. Il semble donc préférable d'avoir un paramètre de chevauchement répondant au critère décrit ci-dessus, soit d'utiliser des valeurs de chevauchement de la forme $\omega = 1 - \frac{1}{\alpha}$ pour $\alpha \in \{1,2,3,...\}$.

2.2.3 Résultats expérimentaux

Jeu de donnée Clomiphène-Acétaminophène Le noyau a été testé sur un ensemble de données généré sur des échantillons de plasma. Cet ensemble de données contenait deux plaques de 96 échantillons. La première de ces deux plaques a eu ses échantillons mis en communs et reséparés en 96 aliquots. Ces échantillons seront alors uniformes. Les échantillons de la deuxième plaque n'ont pas

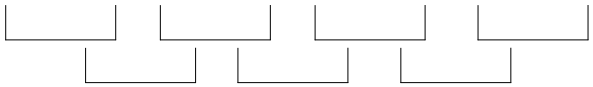

FIGURE 2.7 – **Exemple de boîtes chevauchantes dont le paramètre de chevauchement est à 33%.** On remarque que, contrairement à la figure précédente, la couverture n'est pas uniforme à travers le spectre et certaines zones ne sont couvertes que par une seule boîte.

été mis en commun, mais gardés individuels. Sur le tiers des échantillons de chaque plaque (32 par plaque), les échantillons (ou les aliquots) ont été gardés tels quels. Dans un second tiers des échantillons, du clomiphène a été ajouté aux échantillons. Dans le dernier tiers, une solution de comprimé d'acétaminophène dilué a été ajoutée. On aura donc, par plaque, 32 échantillons sans aucun ajout, 32 échantillons avec du clomiphène ajouté et 32 échantillons avec de l'acétaminophène ajouté.

Ce jeu de données permettra donc de tester les algorithmes d'apprentissage de différentes manières. Étant donné que la première plaque consiste d'échantillons uniformes après la mise en commun, alors les seules différences entre les spectres attendus sont les pics associés aux ajouts. Pour les échantillons de la seconde plaque, on a à la fois les variations interindividuelles du plasma et les molécules ajoutées. Ainsi, on peut tester si l'algorithme est capable d'apprendre sur des échantillons sans variations et de prédire sur les échantillons avec variation ou vice-versa. De plus, les deux molécules ajoutées vont induire différents changements dans les spectres. L'ajout de clomiphène va ajouter un seul pic de haute intensité dans une région du spectre qui n'a normalement pas ou peu de pics. L'ajout de la dilution de comprimés d'acétaminophène de son côté va ajouter plusieurs pics dans le spectre, comme les comprimés ne sont pas composés d'une seule molécule. De plus, il est possible que les donneurs aient consommé de l'acétaminophène avant leurs dons. Étant donné que l'on ajoute une très forte quantité d'acétaminophène, plus forte que ce qu'on retrouverait dans du plasma humain, on va alors observer une forte hausse d'intensité dans des pics avec cet ajout. Il est par contre possible que ce pic soit déjà présent dans des spectres sans ajout, mais à plus faible intensité.

Les échantillons (ou aliquots de la plaque mise en commun) ont premièrement subi une extraction à l'ACN :MeOH. La solution d'extraction était composée de 75% d'ACN et 25% de méthanol. On fait l'extraction avec 10 μL d'échantillon et 90 μL de solution d'extraction. Les solutions subissent ensuite une sonication de 5 minutes. On fait ensuite une centrifugation des échantillons pour 5 minutes à 5000 RPM. On conserve alors le surnageant de l'extraction, laissant les protéines précipitées. On ajoute à ce point le clomiphène ou la solution d'acétaminophène dans les échantillons. On fait finalement une dilution 1 :10 des échantillons afin d'éviter la suppression ionique durant l'acquisition des spectres.

Les spectres ont été acquis avec une source LDTD et un spectromètre Synapt G2 de Waters Corporation. Le patron laser du LDTD était d'une durée de 11 secondes. Le laser reste fermé 2 secondes au

départ. Il s'ouvre ensuite et monte à 65% de sa puissance maximale sur un gradient de 6 secondes. Le laser maintient ensuite son intensité pendant 3 secondes avant de se fermer. Les spectres ont été acquis en mode MS^e sur le spectromètre, qui est un mode d'acquisition data-indépendant. Dans ce mode, on acquiert deux spectres sur l'échantillon. Le premier est une fonction de basse énergie, où l'on n'ajoute aucune énergie de collision dans la cellule de collision du spectromètre. On a donc le spectre contenant les molécules ionisées, mais pas ou peu fragmentées. On acquiert ensuite une fonction de haute énergie où l'on rajoute une énergie de collision, qui causera la fragmentation des molécules ionisées. On acquiert donc un second spectre où l'on mesure les ions filles du premier spectre. Le spectromètre était en mode *high resolution* pour l'acquisition, donnant une précision accrue sur les masses des pics contre une certaine diminution de la sensibilité.

Il est à noter que 7 des échantillons de la plaque mis en communs ont été exclus vu des problèmes expérimentaux et que leurs spectres n'avaient pas été acquis convenablement. On a donc 96 échantillons individuels et 89 exemples de la plaque d'échantillons mis en communs.

Les spectres ont ensuite subi un prétraitement. Ils ont d'abord été prétraités par le logiciel MassLynx de Waters Corporation afin de faire la centroïdation des pics. Ils ont ensuite subi une correction par masses de verrouillage virtuelles (*virtual lock mass*, ou VLM). Les pics ont ensuite subi une filtration par un seuil d'intensité fixé à 500. Les spectres n'ont pas été alignés, car l'on veut vérifier que le noyau OBK soit capable de compenser pour le désalignement aléatoire. Seuls les spectres de la fonction de basse énergie sont utilisés dans la classification.

Les résultats ont été générés avec le noyau OBK et avec l'algorithme du SVM afin de construire un prédicteur basé sur les valeurs de comparaison du noyau. Les hyperparamètres à valider dans ce cas sont donc la valeur C de l'algorithme du SVM, la taille des boîtes ainsi que le chevauchement entre les boîtes. Afin de mieux constater l'influence des paramètres du noyau, les tableaux de résultats présentés ci-dessous montreront une grille des possibilités explorées sur le plan de taille de boîte et de chevauchement entre les boîtes alors que les scores indiqués correspondront aux résultats du meilleur paramètre C, déterminé par validation croisée à 5 plis.

Étant donné que les exemples comportent trois classes, soit les échantillons sans ajout, les échantillons avec clomiphène ajouté et les échantillons avec acétaminophène ajouté, le problème de classification a été séparé en deux. On considérera que soit les échantillons avec clomiphène ou bien les échantillons avec acétaminophène seront la classe positive (+1) et que les échantillons sans ajout et l'autre type d'échantillon avec ajout seront la classe négative (-1). On cherchera donc à classifier acétaminophène (+1) vs sans ajout et clomiphène (-1) ou bien clomiphène (+1) vs sans ajout et acétaminophène (-1). On peut alors toujours considérer le problème comme un problème de classification binaire.

Pour les premiers résultats présentés dans les prochaines tables, l'ensemble d'entraînement était les 89 échantillons provenant de la plaque mise en commun. On s'attend donc à ce que les seules différences entre les spectres soient dues aux ajouts et aux variations de mesure. L'ensemble de test était composé des 96 échantillons provenant de la seconde plaque, restés individuels. On cherche donc à constater si

l'on peut apprendre sur les échantillons avec aucune variation interindividuelle et quand même prédire sur les échantillons individuels, c'est-à-dire si on peut arriver à de bonnes performances prédictives.

Les paramètres utilisés pour ces tests sont les suivants. Les tailles de boîtes testées sont de $\{100, 50, 20, 10, 5, 2\}$ ppm. Les valeurs de chevauchement possibles sont de $\{0\%, 50\%, 66\%, 75\%, 80\%\}$, correspondant à la forme $\omega = 1 - \frac{1}{n}$ où $n \in \{1, 2, 3, 4, 5\}$. Les valeurs du paramètre C du SVM se retrouvaient dans un espace logarithmique de 30 valeurs dont les limites sont de 10^{-20} à 10^{5}. Les résultats indiqués sont la précision des prédictions, soit $precision = 1 - risque$, où le risque est la proportion d'erreurs dans les prédictions. Les paramètres déterminés comme optimaux par validation croisée seront indiqués en gras.

TABLE 2.1 – **Précision de la classification par l'OBK.** La classe positive était les échantillons auxquels a été ajouté de l'acétaminophène. L'ensemble d'entraînement est la plaque d'échantillons mis en communs et l'on prédit sur les échantillons individuels. La valeur en gras représente la valeur optimale déterminée par validation croisée.

Taille / Chevauchement	**0%**	50%	66%	75%	80%
100 ppm	65.17%	64.04%	69.66%	64.04%	70.79%
50 ppm	64.04%	64.04%	89.89%	65.17%	96.63%
20 ppm	70.79%	96.63%	98.88%	96.63%	98.88%
10 ppm	**97.75%**	93.26%	97.75%	97.75%	98.88%
5 ppm	97.75%	96.63%	97.75%	98.88%	98.88%
2 ppm	98.88%	98.88%	98.88%	98.88%	98.88%

On voit à la table 2.1 les résultats lorsqu'on apprend sur une plaque d'échantillons mis en communs afin de prédire si un échantillon contient un ajout d'acétaminophène. On remarque que la validation croisée a obtenu comme résultat optimal la combinaison de paramètres de 10 ppm comme taille des boîtes et aucun chevauchement, ce qui donne une précision de presque 98% sur l'ensemble de test. On remarque également que ce n'est pas la précision optimale sur l'ensemble de test. Ceci est un risque de la validation croisée et seulement visible car on présente des résultats sur l'ensemble de test sur plusieurs paramètres au lieu d'un seul, ce qui n'est pas une pratique standard.

Comparons aux résultats des autres algorithmes, présentés à la table 2.8 plus loin dans ce chapitre. Les algorithmes parcimonieux et le Random Forest ont tous des précisions de 100%. Le SVM à noyau linéaire a une précision de 77% sur cet ensemble de donnée. Le SVM à noyau RBF a quant à lui une précision de 66%. Avec sa précision de 97.75% (choisie par validation croisée), le noyau OBK se compare favorablement aux autres noyaux testés pour l'algorithme du SVM.

On remarque que la précision a tendance à augmenter plus la taille des boîtes diminue. Il est probable que les tailles trop élevées contiennent trop de pics et soient donc trop peu précises et caractéristiques par rapport à certains pics importants. De plus, les boîtes plus grandes ont moins de chances de se retrouver vides, même s'il y a désalignement entre les spectres comparés. On observe aussi que la précision a tendance à augmenter si l'on augmente le chevauchement entre les boîtes. Comparons

maintenant avec les mêmes ensembles d'entraînement et de test, mais en tentant de prédire la présence de clomiphène dans les échantillons.

TABLE 2.2 – **Précision de la classification par l'OBK.** La classe positive était les échantillons auxquels a été ajouté du clomiphène. L'ensemble d'entraînement est la plaque d'échantillons mis en communs et l'on prédit sur les échantillons individuels. La valeur en gras représente la valeur optimale déterminée par validation croisée.

Taille / Chevauchement	**0%**	50%	66%	75%	80%
100 ppm	**28.09%**	28.09%	28.09%	32.58%	32.58%
50 ppm	28.09%	32.58%	49.44%	30.34%	59.55%
20 ppm	28.09%	61.80%	46.07%	29.22%	28.09%
10 ppm	65.17%	28.09%	70.79%	57.30%	29.22%
5 ppm	29.22%	28.09%	29.22%	29.22%	30.34%
2 ppm	39.33%	28.09%	28.09%	55.06%	30.34%

Considérons la table 2.2. Les paramètres choisis par validation croisée sont une taille de boîte de 100 ppm et aucun chevauchement. Cette combinaison de paramètre fait très peu de sens physiquement, puisque une taille de boîte de 100 ppm est extrèmement grand. Par contre, on remarque que la précision avec ces paramètres est très mauvaise. On observe en fait que pratiquement l'entièreté des précisions montrées ici sont mauvaises, puisque le score correspondant à simplement prédire la classe majoritaire est 66% de précision, vu que l'on a 33% d'exemples positifs et 66% d'exemples négatifs. Seule une combinaison de paramètres arrive à battre ce score minimal, soit 70.79% pour des boîtes de 10 ppm et un chevauchement de 66%, ce qui n'est pas un score très performant.

On remarque quand même une certaine tendance où les boîtes plus grandes ont de moins bonnes performances. Cette fois, les performances retombent aussi dans les plus petites boîtes comme 5 ppm et 2 ppm. Les tailles de 10 ppm à 50 ppm contiennent de meilleurs scores. Une légère tendance se dessine au niveau des chevauchements dans ces tailles de boîtes, où les chevauchements de 50% à 75% semblent favoriser la performance.

Comparons les résultats du SVM avec le noyau OBK avec d'autres algorithmes. Ces résultats sont présentés en plus de détails à la table 2.8, plus tard dans ce chapitre. Dans le cas du noyau à boîtes chevauchantes, le classificateur choisi par validation croisée a une précision de 28% environ. Sur le même ensemble de données, un SVM linéaire a un résultat similaire avec 33% de précision. Un SVM avec un noyau RBF ainsi qu'un classificateur Random Forest ont des précisions de 66%. Deux classificateurs parcimonieux, l'arbre de décision et le SCM, ont respectivement 75% et 92% de précision. Le noyau OBK a donc de mauvaises performances sur ce problème comparé aux autres algorithmes.

Une explication possible pour la différence de performance du noyau sur les deux classes est l'effet des différents ajouts sur le spectre. L'acétaminophène, sur lequel l'algorithme performe très bien, va ajouter plusieurs pics et faire de grandes variations d'intensité à plusieurs points. Les pics associés à l'acétaminophène sont aussi possiblement déjà présents dans les échantillons à plus faible intensité. En

contraste, le clomiphène cause l'apparition d'un seul pic de haute intensité. Il n'y a aucun pic présent à cet endroit autrement. Il est possible que, vu l'absence de pics à cet endroit, les boîtes contenant le pic de clomiphène n'aient que des boîtes homologues vides quand on compare un spectre avec clomiphène ajouté et un autre sans clomiphène.

Considérons maintenant le cas où l'algorithme apprend sur les échantillons individuels où l'on a ajouté les molécules pharmaceutiques et où l'on essaie de prédire sur les échantillons mis en communs.

TABLE 2.3 – **Précision de la classification par l'OBK.** La classe positive était les échantillons auxquels a été ajouté de l'acétaminophène. L'ensemble d'entraînement est la plaque d'échantillons individuels et l'on prédit sur les échantillons mis en communs. La valeur en gras représente la valeur optimale déterminée par validation croisée.

Taille / Chevauchement	**0%**	50%	66%	75%	80%
100 ppm	**92.13%**	92.13%	92.13%	91.00%	94.38%
50 ppm	92.13%	91.00%	89.89%	91.00%	91.00%
20 ppm	91.00%	91.00%	89.89%	89.89%	89.89%
10 ppm	87.64%	88.76%	88.76%	88.76%	94.38%
5 ppm	86.52%	95.51%	89.89%	89.89%	89.89%
2 ppm	88.76%	88.76%	88.76%	88.76%	88.76%

La table 2.3 contient les résultats de prédictions sur l'ajout d'acétaminophène, lorsqu'entraîné sur les échantillons individuels. On remarque encore une bonne performance de prédiction, malgré qu'elle soit plus basse que dans le cas précédent. La combinaison de paramètres de taille de boîtes de 100 ppm et aucun chevauchement a été choisie comme étant optimale par validation croisée. Par contre, il faut noter que la plupart des combinaisons de paramètres obtenaient à un risque de validation croisée de 0 (soit une précision de 100%) également. Dans ce cas, il n'y a pas vraiment de tendances claires relié à la taille des boîtes ou le chevauchement, sauf peut-être le fait que les boîtes de 2 ppm aient des performances moindres que les autres tailles.

Comparons aux résultats de la table 2.10, présentée plus loin dans ce chapitre. La précision de 92% du noyau OBK est comparable ou supérieure à celles du Random Forest (76%), Decision Tree (94%), SVM à noyau linéaire (91%) et SVM à noyau RBF (72%). Le seul algorithme ayant une performance clairement supérieure est le SCM, ayant une précision de presque 98%.

Considérons maintenant la table 2.4, présentant les résultats de prédiction de l'ajout de clomiphène. On a encore de mauvais résultats pour la prédiction sur cette molécule. Il est à noter que le score de 64.04% de précision revient très souvent, car il correspond à un classificateur qui prédit seulement négatif sur l'ensemble d'entraînement. Tel quel mentionné précédemment, certains spectres ont du être retirés du jeu de données. L'absence de ces spectres cause ce changement. Encore une fois, certaines combinaisons de paramètres ont une performance meilleure que la performance naive de 64%, mais rien de meilleur que 69.66%. Il semble clair que le noyau n'est pas adapté au problème de la prédiction de l'ajout de clomiphène.

TABLE 2.4 – **Précision de la classification par l'OBK.** La classe positive était les échantillons auxquels a été ajouté du clomiphène. L'ensemble d'entraînement est la plaque d'échantillons individuels et l'on prédit surs les échantillons mis en communs. La valeur en gras représente la valeur optimale déterminée par validation croisée.

Taille / Chevauchement	0%	50%	**66%**	75%	80%
100 ppm	64.04%	64.04%	64.04%	64.04%	64.04%
50 ppm	64.04%	64.04%	64.04%	66.29%	64.04%
20 ppm	64.04%	64.04%	**64.04%**	65.17%	64.04%
10 ppm	64.04%	67.42%	62.92%	64.04%	64.04%
5 ppm	65.17%	65.17%	69.66%	62.92%	67.42%
2 ppm	69.66%	60.67%	64.04%	57.30%	52.81%

Si l'on compare encore une fois aux différents autres algorithmes, dont on peut voir les résultats à la table 2.11 plus loin dans ce chapitre, on remarque que le noyau OBK a la même performance que les autres noyaux du SVM. Les trois autres ont par contre des performances supérieures se trouvant entre 86% de précision (Random Forest) et 97% (Arbre de décision, SCM).

Finalement, le noyau OBK a été testé sur le même ensemble de données, mais avec les ensembles d'entraînement et de tests séparés aléatoirement sur le nombre total d'échantillons au lieu de tester explicitement une plaque sur une autre.

TABLE 2.5 – **Précision de la classification par l'OBK.** La classe positive était les échantillons auxquels a été ajouté de l'acétaminophène. L'ensemble d'entraînement et l'ensemble de tests sont pris aléatoirement parmi les échantillons. La valeur en gras représente la valeur optimale déterminée par validation croisée.

Taille / Chevauchement	**0%**	50%	66%	75%	80%
100 ppm	**98.96%**	98.96%	97.92%	98.96%	97.92%
50 ppm	98.96%	98.96%	97.92%	98.96%	98.96%
20 ppm	96.87%	98.96%	97.92%	97.92%	97.92%
10 ppm	98.96%	98.96%	97.92%	97.92%	97.92%
5 ppm	96.87%	97.92%	95.83%	96.87%	97.92%
2 ppm	97.92%	95.83%	95.83%	95.83%	95.83%

La table 2.5 contient les résultats de la prédiction de l'ajout d'acétaminophène dans les ensembles d'entraînement et de test aléatoires. Encore une fois, on observe une bonne performance de l'OBK afin de faire la prédiction pour l'acétaminophène. La combinaison de paramètres optimaux par validation croisée était une taille de boîtes de 100 ppm et un chevauchement de 0%. Il est à noter qu'encore une fois, plusieurs autres combinaisons de paramètres ont égalé le score de validation croisée de cette combinaison. Étant donnée que les ensembles d'entraînement et de test sont aléatoires dans ce cas, on constate que l'algorithme maintient sa performance.

Comparons aux résultats des autres algorithmes sur le même ensemble de données, qui se trouve à a table 2.12. Encore une fois, pour la prédiction de la présence d'acétaminophène, le noyau OBK a une performance comparable aux autres algorithmes. Le Random Forest a une précision de 96%, l'arbre de décision une précision de 98% et deux algorithmes ont une précision de 99% : le SVM à noyau linéaire et le SCM. Le SVM à noyau RBF a une moins bonne performance avec une précision de 73%.

TABLE 2.6 – **Précision de la classification par l'OBK.** La classe positive était les échantillons auxquels a été ajouté du clomiphène. L'ensemble d'entraînement et l'ensemble de tests sont pris aléatoirement parmi les échantillons. La valeur en gras représente la valeur optimale déterminée par validation croisée.

Taille / Chevauchement	0%	**50%**	66%	75%	80%
100 ppm	70.83%	**68.75%**	70.83%	70.83%	65.62%
50 ppm	66.67%	71.87%	64.58%	67.71%	64.58%
20 ppm	61.46%	67.71%	61.46%	73.96%	65.62%
10 ppm	64.58%	71.87%	64.58%	67.71%	64.58%
5 ppm	61.46%	58.33%	60.42%	61.46%	62.50%
2 ppm	61.46%	61.46%	64.58%	61.46%	61.46%

Comparons à la performance de l'algorithme à la prédiction de la présence de clomiphène sur les mêmes ensembles d'entraînement et de test, présenté à la table 2.6, qui se retrouve plus loin dans ce chapitre. On remarque encore une fois une mauvaise performance de l'algorithme à ce problème. Étant donné les nouveaux ensembles d'entraînement et de test, il semble alors que l'algorithme est effectivement mal adapté à classifier la différence introduite par le clomiphène. On note plusieurs scores de 61.46%, ce qui correspond à la séparation des classes, signifiant que le classificateur dans ces cas ne fait que prédire des exemples négatifs. On observe quelques combinaisons pouvant atteindre une précision supérieure à 70%, mais même ces meilleures performances ne sont pas excellentes ou significativement différentes des autres.

Comparons une dernière fois aux performances des autres algorithmes, présentés à la table 2.13. Avec sa précision d'environ 69%, le SVM avec noyau OBK est moins performant que la plupart des autres algorithmes. Le SVM à noyau RBF a une moins bonne précision, de 63%. Les autres algorithmes ont des précisions allant de 73% jusqu'à 98%. Le noyau OBK semble toujours avoir de la difficulté à prédire la présence de clomiphène.

2.3 Application d'algorithmes existants à la spectrométrie de masse

En plus de mettre au point un nouveau noyau, une grande partie des travaux présentés ici concerne l'application de plusieurs algorithmes déjà existants aux données de spectrométrie de masse. Principalement, on recherche à appliquer des algorithmes dont les modèles de classification sont parcimonieux et interprétables afin de pouvoir trouver des biomarqueurs dans les spectres.

Une grande partie des travaux a été appliquée à des spectres qui ont été générés à partir de technologie LDTD. Les résultats associés à ces jeux de données seront présentés en premier. Une seconde partie concerne des spectres acquis par LCMS. Ils seront présentés par la suite.

2.3.1 Application sur des données générées par LDTD

Jeu de données de plasma dégradé

Un premier test fait était sur deux plaques d'échantillons de plasma sanguin. Une première plaque fut décongelée et laissée à température pièce pendant 7 jours. Cette plaque forme les échantillons de plasma dégradés, formant les exemples de la classe négative. L'autre plaque fut décongelée et traitée immédiatement, formant la classe positive des échantillons frais. L'objectif de cette expérience est de voir si l'on est capable d'apprendre la différence entre les échantillons frais et dégradés à température pièce avec l'apprentissage automatique et l'acquisition de spectres avec la technologie LDTD.

Afin d'éviter tout biais, la moitié des échantillons frais ont été placé sur une plaque avec la moitié des échantillons dégradés. On partage la seconde plaque de la même manière. Ainsi, on évite qu'un biais dans les variations de masses lié aux plaques soit directement corrélé avec les classes.

La méthodologie d'acquisition de ces spectres a déjà été décrite à la Section 1.3.1 de ce mémoire, en tant que *jeu de données d'évaluations des déviations*.

Les spectres ont subi un premier traitement informatique par le logiciel MassLynx de Waters Corporation. Ce premier traitement était la centroïdation des pics. Après cela, une correction par VLM a été appliquée aux spectres des ions positifs de l'extraction ACNMeOH et aux spectres des ions négatifs de l'extraction Chlbut. Les points de correction VLM sont les mêmes qui ont été utilisés à la table1.6 du dernier chapitre. Une filtration des pics par un seuil d'intensité de 500 fut aussi appliquée. On a ensuite fait l'alignement des pics avec une distance maximale de 5 ppm. Le spectre repère fut bâti sur l'ensemble d'entraînement (voir le prochain paragraphe). Finalement, les 500 pics les plus intenses en moyenne par type de spectres furent sélectionnés. On a donc 1000 pics par échantillons afin de faire la comparaison (500 pics de la condition ACNMeOH-positive et 500 de la condition Chlbut négative). Cette dernière filtration a été appliquée afin de rendre l'apprentissage computationnellement assez rapide pour l'algorithme du SCM.

On a alors un total de 141 échantillons. Plusieurs des échantillons acquis ont dû être retirés à cause de problèmes lors de l'acquisition soit du spectre ACNMeOH-positif ou du spectre Chlbut négatif. On compte 73 échantillons de classe "frais" et 68 échantillons de classe "dégradée" dans l'ensemble des données. L'ensemble des données a ensuite été séparé aléatoirement en un ensemble de tests de 30 échantillons et un ensemble d'entraînement de 111 échantillons. Les hyperparamètres optimaux à utiliser furent déterminés par validation croisée à 5 plis.

Plusieurs algorithmes ont été appliqués à ces données. Le premier est celui des forêts d'arbres aléatoires (*Random Forest*). Cet algorithme n'a qu'un hyperparamètre à sélectionner, le nombre d'estima-

teurs. Ce nombre d'estimateurs pourrait se trouver dans l'ensemble $\{1,5,10,20,30,40,50,60,70,80,90,100,150\}$. L'arbre de décision (*decision tree*) a quant à lui deux hyperparamètres à sélectionner. Le premier est la profondeur maximale de l'arbre, qui pouvait se retrouver dans l'ensemble $\{2,3,4,5,6,7,8,9,10\}$. Le second paramètre est le nombre minimal que chaque décision doit séparer. Ce nombre pouvait être choisi dans $\{2,5,10,15,20\}$.

Deux formes de l'algorithme des machines à vecteur de support (*Support Vector Machine*, SVM) ont été essayées sur cet ensemble de données. Le premier est le SVM à noyau linéaire. Dans ce cas, il n'y a pas de fonction de noyau projetant dans une dimension supérieure, seulement l'algorithme SVM qui tente de séparer les données dans l'espace des caractéristiques. Son seul paramètre est le paramètre C, qui peut prendre une valeur dans un espace logarithmique de 7 valeurs entre 10^{-3} et 10^{3}. Le second noyau utilisé est le noyau *Radial Basis Function* (RBF). Vapnik (1995) C'est un noyau utilisé couramment en apprentissage automatique, qui projette les exemples dans un espace de dimensionnalité infinie. Cette forme de l'algorithme a deux paramètres à sélectionner, soit le paramètre C et le paramètre γ. Les deux peuvent choisir des valeurs dans un espace logarithmique de 7 valeurs contenues entre 10^{-3} et 10^{3}. Un espace logarithmique à 7 valeurs entre 10^{-3} et 10^{3} correspond à l'ensemble $\{0.001,0.01,0.1,1,10,100,1000\}$.

Le dernier algorithme utilisé est le SCM. Les règles de classifications pour l'algorithme correspondent à des souches de décision. On définit les souches de décision sur chacune des caractéristiques des exemples. On définit deux souches de décision pour chaque valeur différente dans l'ensemble d'entraînement sur chacune des caractéristiques, une qui est plus grande ou égale à la valeur et une qui est inférieure à la valeur. Par exemple, si l'on retrouve 5 valeurs différentes sur le pic de ratio m/z 123.4567, alors on définira 10 souches qui correspondent à $x \geq val_1$, $x < val_1$, $x \geq val_2$, $x < val_2$, etc. On peut alors avoir jusqu'à un maximum de $2m$ souches de décision par caractéristique, où m est le nombre d'exemples dans l'ensemble d'entraînement. C'est pourquoi on applique une sélection sur les pics, car le nombre de souches serait très grand avec un grand nombre d'exemples et de pics. Le SCM a trois paramètres à sélectionner. Le premier est le type de modèle à bâtir, soit une conjonction de règles ou une disjonction de règles. Un autre paramètre est le nombre de règles maximum à considérer. Ce paramètre est sélectionné dans l'ensemble $\{1,2,3,4,5,6,7,8,9,10,12,15\}$. Finalement, le SCM a un paramètre p. Ce paramètre p pouvait être dans un espace logarithmique de 5 valeurs entre 10^{-2} et 10^{2} ($\{0.01,0.1,1,10,100\}$).

La table 2.7 présente les résultats de classification. On remarque que chaque algorithme n'a pas de bonne précision sur l'ensemble de tests. Les scores de 43.33% et 56.67% correspondent aux classificateurs qui tentent, respectivement, de prédire que tout l'ensemble est fait d'échantillons "dégradés" et "frais". On remarque pourtant que plusieurs des algorithmes ont un score parfait ou presque sur l'ensemble d'entraînement. Le SCM est celui avec une moins bonne performance sur l'ensemble d'entraînement avec environ 83% de performance. Cela est une performance assez basse sur un ensemble d'entraînement. Cela indique que les algorithmes ont pu apprendre des séparations des exemples de l'ensemble d'entraînement, mais que celles-ci ne généralisent pas sur l'ensemble de test. Il semble

TABLE 2.7 – **Résultats de la classification d'échantillons de plasma frais vs des échantillons de plasma dégradés sur plaques mixtes**. Les scores de classification sont indiqués en précision, qui correspond à $precision = 1 - risque$ où le risque est la proportion d'erreurs faites par le classificateur.

Algorithme	Paramètres	Précision empirique	Précision de test
Random Forest	Estimateurs = 100	100.00%	56.67%
Décision Tree	Profondeur = 9, exemples séparés = 2	100.00%	43.33%
SVM Linéaire	$C = 1$	100.00%	56.67%
SVM Noyau RBF	$C = 1$, $\gamma = 0.01$	99.10%	56.67%
SCM	Conjonction, Attributs = 12, $p = 10$	82.88%	43.33%

donc que tous les algorithmes, y compris le SCM, subissent une forme de surapprentissage.

Comme mentionné au paragraphe précédent, on peut remarquer que le SCM a utilisé un grand nombre d'attributs et a quand même une précision de seulement 83% sur l'ensemble d'entraînement. Cela suggère que c'est une tâche très difficile de séparer les échantillons avec peu de règles dans ce cas. Il est également possible qu'un plus grand nombre d'échantillons soit nécessaire à l'apprentissage. De plus, tous les algorithmes n'arrivent pas à généraliser.

Une explication possible pour ces résultats est qu'il n'y a pas de pic permettant de faire la distinction entre les classes "frais" et "dégradé" dans les spectres. L'hypothèse posant que l'on pourrait faire la différence entre ces classes à partir seulement des petits métabolites est possiblement erronée. Une autre possibilité est qu'il y a trop de variation, en termes de masse ou d'intensité, pour que les algorithmes d'apprentissage puissent apprendre et généraliser sur les spectres de masse générés par la méthode décrite plus haut. Il semble alors qu'il faille tester si les méthodes de traitement de données et l'apprentissage automatique sont capables d'apprendre et de généraliser sur des spectres de masse.

Jeu de données de plasma avec molécules pharmaceutiques ajoutées

Étant donné l'incapacité à bien prédire dans l'expérience précédente peut provenir d'un problème au niveau des spectres et leur contenu ou bien du traitement des données et de l'apprentissage automatique, il nous faut tester une de ces deux hypothèses. Pour ce faire, un nouvel ensemble de données a été généré. Il sert à faire une preuve de concept afin de savoir si l'apprentissage automatique est applicable à des spectres de masse acquis par source LDTD. Afin de se faire, il nous faut avoir des spectres dont on connaît le contenu et dont on est certain qu'il contienne un moyen de faire la différence entre les classes.

Le nouvel ensemble de données utilisé pour ce faire était celui d'échantillons de plasma avec molécules pharmaceutiques ajoutées, qui a déjà été mentionné dans la section à propos du noyau à boîtes chevauchantes. Son acquisition est donc décrite à la Section 2.2.3. Ce jeu de données est composé de

deux plaques d'échantillons de plasma. Une première plaque a vu tous ses échantillons mis en communs, c'est-à-dire que les 96 échantillons furent mélangés et re séparés en 96 aliquots. La seconde a gardé ses échantillons individuels, soit 1 échantillon différent par puits. Dans le tiers des échantillons a été ajoutée une solution contenant du clomiphène. Une autre solution à base de comprimés d'acétaminophène a été faite et ajoutée à un second tiers des échantillons. Le dernier tiers est resté sans ajouts. Parmis les 96 échantillons mis en communs, il n'y aura que les ajouts et les différences de mesure expérimentales qui seront différents de puits en puits pour cette portion du jeu de données. Dans un second temps, il y a également 96 échantillons de plasma avec variations interindividuelles en plus des ajouts introduits. Ainsi, on détient un jeu de données avec trois classes d'échantillons où l'on sait que la différence entre les classes se retrouve dans le spectre de masse, soit une classe sans ajout, une classe avec ajout de clomiphène et une classe avec ajout d'acétaminophène. Il existe alors une façon de vérifier si les algorithmes d'apprentissage peuvent retrouver une vérité qui nous est connue.

Les spectres ont reçu un premier traitement informatique dans le logiciel MassLynx de Waters Corporation. Ce traitement était la centroïdation des pics. Une correction par VLM a ensuite été appliquée, utilisant les points de corrections indiqués à la table 1.6 pour les spectres ACNMeOH-positifs. Une filtration des spectres a ensuite été appliquée avec un seuil d'intensité de 500 afin d'éliminer le bruit de fond des spectres. Finalement, un alignement a été appliqué sur les spectres. La distance maximale était de 15 ppm. Le spectre de repère a été bâti sur l'ensemble d'entraînement seulement, puis l'alignement a été appliqué à tous les spectres.

L'ensemble des données est alors composé de 89 échantillons mis en communs, comme certains spectres ont eu des problèmes lors de l'acquisition et ont dû être retirés [3]. Ces échantillons retirés étaient tous des ajouts d'acétaminophène, résultant en 25 de ce type d'échantillon au lieu de 32 sur cette plaque. Nous avons 96 échantillons individuels comprenant 32 exemples de chacune des trois classes.

Trois types d'expériences ont été faites à partir de cet ensemble de données. La première (Expérience 1) est que l'on déclare les échantillons mis en commun comme ensemble d'entraînement et les échantillons individuels comme ensemble de tests. La deuxième expérience (Expérience 2) est l'inverse, avec les échantillons individuels servant d'ensemble d'entraînement. On peut ainsi observer l'effet des variations interindividuelles sur l'apprentissage. Finalement, la troisième expérience (Expérience 3) consiste à séparer l'ensemble des données en deux, aléatoirement. On a alors un ensemble de 93 échantillons pour l'entraînement et 92 échantillons pour le test, chacun des ensembles composés d'échantillons mis en communs et individuels. Dans ces trois expériences, l'objectif est de classifier si un échantillon contient un ajout d'acétaminophène.

Chaque algorithme avait des hyperparamètres à valider par validation croisée à 5 plis. L'algorithme des forêts d'arbres de décision pouvait avoir un nombre d'estimateurs dans l'ensemble $\{1, 5, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 150, 200\}$. L'arbre de décision pouvait avoir une profondeur maximale d'entre 2

3. Problèmes

et 10 ($\{2,3,4,5,6,7,8,9,10\}$) et un nombre minimal d'exemples à séparer par règle dans l'ensemble $\{2,5,10,15,20\}$. Le SVM à noyau linéaire n'a que le paramètre C à choisir, qui pouvait adopter une valeur dans $\{0.001,0.01,0.1,1,10,100,1000\}$. Ce même ensemble était utilisé pour les deux paramètres du SVM à noyau RBF, soit les paramètres C et γ. Finalement, le SCM avait trois paramètres à valider. Le premier est son modèle d'apprentissage, soit par conjonction ou disjonction. Le nombre maximal de règles utilisé par le SCM pouvait être de $\{1,2,3,4,5,6,7,8,9,10,12,15\}$. Finalement, le paramètre p du SCM pouvait prendre des valeurs de $\{0.01,0.1,1,10,100\}$.

Un changement fut également appliqué à l'algorithme du SCM. En cas d'égalité de l'utilité de deux règles, une fonction de bris d'égalité a été mise au point. Cette fonction favorise les règles sur les caractéristiques ayant, en moyenne, une plus forte intensité. Ce choix a été fait afin de favoriser les pics de plus haute intensité pour la décision.

Expérience 1 - Entraînement sur les échantillons mis en commun

TABLE 2.8 – **Résultats de la classification de l'ajout d'acétaminophène dans les échantillons de plasma.** Les algorithmes sont entraînés sur la plaque d'échantillons mis en commun et prédit sur les échantillons individuels (Expérience 1). Les scores de classification sont indiqués en précision, qui correspond à $precision = 1 - risque$ où le risque est la proportion d'erreurs faites par le classificateur.

Algorithme	Paramètres	Précision empirique	Précision de test
Random Forest	Estimateurs = 200	100%	100%
Décision Tree	Profondeur = 2, exemples séparés = 20	100%	100%
SVM Linéaire	$C = 0.001$	100%	77.08%
SVM Noyau RBF	$C = 0.001$, $\gamma = 0.001$	71.91%	66.67%
SCM	Disjonction, Attributs = 1, $p = 0.01$	97.75%	100%

Les résultats de la classification de l'ajout d'acétaminophène alors qu'on s'entraîne sur les échantillons mis en commun (Expérience 1) sont visibles à la table 2.8. On remarque d'excellentes performances sur ce problème par les algorithmes du SCM, de l'arbre de décision et des forêts d'arbres aléatoires. Le SVM à noyau linéaire a aussi réussi à séparer les classes dans l'ensemble d'entraînement, mais le classificateur n'a pas très bien généralisé aux échantillons individuels. On peut remarquer que le SVM à noyau RBF a de mauvaises performances et tend à ne prédire que la classe majoritaire. Malgré des tentatives d'élargir l'espace des paramètres explorés jusqu'à 10^{-20} et 10^{10}, les classificateurs sont toujours restés avec les mêmes risques empiriques et de validation croisée. Les résultats présentés pour cet algorithme dans la table 2.8 et les tables suivantes seront simplement ceux de l'espace des paramètres de base (10^{-3} à 10^{3}).

On peut voir à la table 2.9 les résultats reliés à la classification de l'ajout de clomiphène pour l'expérience 1, appris sur les échantillons mis en communs. Dans ce cas, on observe de moins bons résultats

TABLE 2.9 – **Résultats de la classification de l'ajout de clomiphène dans les échantillons de plasma.** Les algorithmes sont entraînés sur la plaque d'échantillons mis en commun et prédit sur les échantillons individuels (Expérience 1). Les scores de classification sont indiqués en précision, qui correspond à $precision = 1 - risque$ où le risque est la proportion d'erreurs faites par le classificateur.

Algorithme	Paramètres	Précision empirique	Précision de test
Random Forest	Estimateurs = 60	100%	66.67%
Décision Tree	Profondeur = 3, exemples séparés = 5	100%	75.00%
SVM Linéaire	$C = 0.001$	100%	33.33%
SVM Noyau RBF	$C = 0.001$, $\gamma = 0.001$	64.04%	66.67%
SCM	Conjonction, Attributs = 1, $p = 1$	97.75%	92.71%

que la classification de l'ajout d'acétaminophène. On remarque que l'arbre de décision et les forêts d'arbres aléatoires ont une bonne performance sur l'ensemble d'entraînement, mais n'arrivent pas à très bien généraliser sur les nouveaux échantillons. Le score de test des forêts d'arbres aléatoires indique que le classificateur a tenté de classer tous les échantillons en une seule classe. Le SVM a également été incapable de généraliser ses prédicteurs. Finalement, le SCM est le seul algorithme avec une bonne performance sur ce problème. Avec un seul attribut, soit le pic de clomiphène ajouté, ce classificateur arrive à bien apprendre sur les spectres.

Expérience 2 - Entraînement sur les échantillons individuels

TABLE 2.10 – **Résultats de la classification de l'ajout d'acétaminophène dans les échantillons de plasma.** Les algorithmes sont entraînés sur la plaque d'échantillons individuels et prédits sur les échantillons mis en commun (Expérience 2). Les scores de classification sont indiqués en précision, qui correspond à $precision = 1 - risque$ où le risque est la proportion d'erreurs faites par le classificateur.

Algorithme	Paramètres	Précision empirique	Précision de test
Random Forest	Estimateurs = 90	100%	76.40%
Décision Tree	Profondeur = 2, exemples séparés = 2	100%	94.38%
SVM Linéaire	$C = 0.001$	100%	91.01%
SVM Noyau RBF	$C = 0.001$, $\gamma = 0.001$	66.67%	71.91%
SCM	Conjonction, Attributs = 1, $p = 0.01$	100%	97.75%

Considérons maintenant les résultats si l'on inverse les ensembles d'entraînement et de test, apprenant sur les échantillons individuels, soit l'expérience 2. La table 2.10 montre les résultats sur la classification de l'ajout d'acétaminophène dans ce cas. On remarque de bonnes performances des algorithmes de l'arbre de décision et du SCM. Dans les deux cas, la prédiction dépend d'un seul pic. Le SVM

à noyau linéaire arrive aussi à bien généraliser son classificateur entre les deux types d'échantillons. Les forêts d'arbres aléatoires ont par contre une certaine difficulté à généraliser le prédicteur. Il est possible que, vu le principe de fonctionnement à partir d'arbres décisionnels basés sur des exemples aléatoires et des caractéristiques aléatoires, le classificateur ait possiblement trop appris à partir des variations interindividuelles et soit donc incapable de prédire sur des exemples sans ce bruit.

TABLE 2.11 – **Résultats de la classification de l'ajout de clomiphène dans les échantillons de plasma.** Les algorithmes sont entraînés sur la plaque d'échantillons individuels et prédits sur les échantillons mis en commun (Expérience 2). Les scores de classification sont indiqués en précision, qui correspond à $precision = 1 - risque$ où le risque est la proportion d'erreurs faites par le classificateur.

Algorithme	Paramètres	Précision empirique	Précision de test
Random Forest	Estimateurs = 70	100%	86.52%
Décision Tree	Profondeur = 2, exemples séparés = 2	100%	96.63%
SVM Linéaire	$C = 0.001$	100%	64.04%
SVM Noyau RBF	$C = 0.001$, $\gamma = 0.001$	66.67%	64.04%
SCM	Conjonction, Attributs = 1, $p = 0.01$	100%	96.63%

Comparons maintenant avec la classification de l'ajout de clomiphène pour l'expérience 2 à la table 2.11. On remarque de bonnes performances principalement de l'arbre de décision et du SCM. Les forêts d'arbres aléatoires ont une performance plus mitigée, sous 90% de précision. Les différents noyaux du SVM n'ont pu apprendre de classificateur qui généralise pour ce problème. Ce n'est pas étonnant, étant donné que l'ajout de clomiphène ajoute seulement un pic au spectre. Il est donc cohérent que les algorithmes plus parcimonieux aient une meilleure performance pour ce problème.

Expérience 3 - Entraînement pigé aléatoirement

TABLE 2.12 – **Résultats de la classification de l'ajout d'acétaminophène dans les échantillons de plasma.** Les ensembles d'entraînement et de tests sont tirés aléatoirement (Expérience 3). Les scores de classification sont indiqués en précision, qui correspond à $precision = 1 - risque$ où le risque est la proportion d'erreurs faites par le classificateur.

Algorithme	Paramètres	Précision empirique	Précision de test
Random Forest	Estimateurs = 90	100%	95.65%
Décision Tree	Profondeur = 2, exemples séparés = 2	100%	97.83%
SVM Linéaire	$C = 0.001$	100%	98.91%
SVM Noyau RBF	$C = 0.001$, $\gamma = 0.001$	65.59%	72.83%
SCM	Conjonction, Attributs = 1, $p = 0.01$	100%	98.91%

Finalement, comparons les résultats lorsque les ensembles d'entraînement et de test sont tirés aléatoirement dans l'ensemble de données total (Expérience 3). La table 2.12 présente la classification de l'ajout d'acétaminophène dans cette situation. On remarque que tous les algorithmes, excepté le SVM à noyau RBF, ont une bonne performance de classification dans ce cas. On peut également observer que les algorithmes du SCM et de l'arbre de décision arrivent à de bonnes performances avec encore une fois un seul pic utilisé pour la classification.

TABLE 2.13 – **Résultats de la classification de l'ajout de clomiphène dans les échantillons de plasma.** Les ensembles d'entraînement et de tests sont tirés aléatoirement (Expérience 3). Les scores de classification sont indiqués en précision, qui correspond à $precision = 1 - risque$ où le risque est la proportion d'erreurs faites par le classificateur.

Algorithme	Paramètres	Précision empirique	Précision de test
Random Forest	Estimateurs = 40	100%	72.83%
Décision Tree	Profondeur = 4, exemples séparés = 2	100%	94.57%
SVM Linéaire	$C = 0.001$	100%	77.17%
SVM Noyau RBF	$C = 0.001$, $\gamma = 0.001$	67.74%	63.04%
SCM	Conjonction, Attributs = 1, $p = 0.01$	98.91%	97.83%

La table 2.13 quant à elle présente les résultats de la classification de l'ajout de clomiphène pour l'expérience 3. Comme toujours, le SVM à noyau RBF reste incapable d'apprendre sur cet ensemble de données. On remarque de moins bonnes performances que pour l'acétaminophène sur les mêmes ensembles d'apprentissages pour les algorithmes de forêts d'arbres aléatoires et pour le SVM à noyau linéaire. On voit par contre que l'arbre de décision et le SCM gardent de bonnes performances de prédiction pour ce problème. Le SCM n'a encore une fois eu besoin de seulement un pic pour faire la classification.

Les résultats sur cet ensemble de données établissent donc une preuve de concept que les algorithmes d'apprentissage sont capables d'apprendre sur le contenu d'un spectre de masse généré par LDTD et avec les méthodes de prétraitement développées au cours du projet. Plus exactement, cela fournit une preuve que les algorithmes peuvent apprendre si l'on sait que les classes ont une différence observable dans leurs spectres de masse. Par contre, l'ensemble de données a été généré avec des molécules ajoutées dans les échantillons de plasma et ne correspond pas exactement à la réalité.

2.3.2 Application sur des données générées par LC-MS

Jeu de données LC-MS Comme preuve de concept supplémentaire pour l'application de l'apprentissage automatique à la spectrométrie de masse, un nouvel ensemble de données a été généré. Il s'agit d'un ensemble de 48 échantillons de plasma qui ont été acquis par LC-MS. Comme on ne veut pas ajouter de molécules dans les échantillons et classifier les échantillons à partir de caractéristiques de leurs métabolites, nous avons utilisé les métadonnées anonymes de l'ensemble pour séparer ces

échantillons en deux classes. 28 des échantillons proviennent de donneurs masculins et 20 de donneurs féminins. On va ainsi tenter de classifier les échantillons selon le sexe du donneur à partir des spectres de masse du plasma, c'est-à-dire à partir des métabolites sanguins. Une molécule qui devrait nous fournir la différence entre ces classes est l'hormone de la testostérone. On s'attend à voir dans les spectres de plasma masculins des quantités importantes de cette hormone et den observer des niveaux fortement moindres chez les spectres de donneurs féminins.

Les échantillons ont premièrement été décongelés. Ils ont ensuite subi une extraction à l'ACN :MeOH en proportion 75 :25. On extrait 10 μL d'échantillon avec 90 μL de solution d'extraction ACN :MeOH. Les échantillons subissent alors une sonication de 5 minutes. On centrifuge alors les échantillons pendant 5 minutes à 5000 RPM. On récupère alors le surnageant. On a ainsi les petits métabolites du plasma sans les protéines. On prend alors 10 μL d'échantillon pour l'injecter en chromatographie liquide.

Les échantillons ont été acquis par LC-MS dans un spectromètre Synapt G2 de Waters Corporation. La chromatographie liquide s'est faite dans une colonne Acquity UPLC BEH C18 de Waters Corporation. Cette colonne contient des billes de 130 Å et est d'une taille de 2.1mm par 100 mm. La chromatographie s'est effectuée avec un gradient de solvant de 10 minutes. Le solvant était, au début de la chromatographie, composé de 100% d'eau. À la fin de la chromatographie, le solvant est de 100% d'ACN. La composition varie graduellement de 100% d'eau à 100% d'ACN à travers le temps de la chromatographie.

Les spectres ont été acquis en mode MS^e sur le spectromètre, c'est-à-dire un mode d'acquisition data-indépendant. Le spectromètre acquiert alternativement deux spectres sur les ions. Le premier est une fonction de basse énergie où l'on voit les molécules peu ou pas fragmentées. Le second ajoute une énergie de collision dans le spectromètre qui fait fragmenter les molécules. On a ainsi un spectre avec les fragments filles des molécules du premier spectre. Les spectres ont aussi été acquis en mode *resolution* par le spectromètre. Ce mode est moins précis sur les ratios de masse sur charge que le mode *high resolution* utilisée précédemment, mais est plus sensible.

Le prétraitement des spectres a été fait par le logiciel Progenesis QI de Nonlinear Dynamics. Ce logiciel fait la centroïdation des pics, l'alignement des pics ainsi que la normalisation des intensités automatiquement. Le logiciel fait aussi une sélection des pics. Cette sélection a été faite avec le seuil automatique du logiciel.

Dans cette expérience, l'algorithme des masses de verrouillage virtuelles ne fut pas appliqué. L'algorithme n'est effectivement pas nécessaire afin de rendre les spectres comparables dans ce contexte. La méthodologie LC-MS est très répandue dans le domaine de la spectrométrie de masses et des algorithmes de correction et d'alignement existent déjà afin de rendre les spectres comparables. De plus, vu que l'on dispose d'une information supplémentaire sur les pics, soit leur temps d'élution lors de la chromatographie, alors les pics sont caractérisés par le temps d'élution et le ratio de masse sur charge. L'algorithme des masses de verrouillage virtuelles n'est donc pas applicable à ce type de donné sans

changement majeur, puisqu'il est conçu pour corriger *seulement* le ratio de masse sur charge.

L'ensemble des données a été séparé aléatoirement en deux ensembles de 24 exemples. Le premier est l'ensemble d'entraînement et le second l'ensemble de tests. Les hyperparamètres ont été choisis sur l'ensemble d'entraînement par validation croisée à 5 plis. Comme précédemment, les forêts d'arbres aléatoires pouvaient avoir un nombre d'estimateurs parmi $\{1,5,10,20,30,40,50,60,70,80,90,100,150,200\}$. Les arbres décisionnels pouvaient avoir une profondeur d'entre 2 et 10 ($\{2,3,4,5,6,7,8,9,10\}$) et un minimum d'exemples séparés par règle de $\{2,5,10,15,20\}$. Le SCM avait trois hyperparamètres à valider. Le premier est le type de modèle d'apprentissage, soit une conjonction ou une disjonction. Le second est le nombre maximal de règles à utiliser, qui pouvait être dans l'ensemble $\{1,2,3,4,5,6,7,8,9,10,12,15\}$. Finalement, le paramètre p pouvait prendre des valeurs de $\{0.01,0.1,1,10,100\}$.

Pour les SVM, l'étendue des paramètres à tester a dû être grandement élargie. Pour le SVM à noyau linéaire, un espace logarithmique de 50 valeurs entre 10^{-12} et 10^{12} a été testé pour la paramètre C. Pour le SVM à noyau RBF, les deux paramètres avaient le même espace à explorer. Le paramètre C et le paramètre γ pouvaient se retrouver dans un espace logarithmique de 50 valeurs entre 10^{-12} et 10^{12}. Étant donné que l'on valide toutes les combinaisons possibles, on obtient 2500 combinaisons possibles pour le SVM RBF.

TABLE 2.14 – **Résultats de la classification du sexe de donneurs de plasma sur des données acquises par LCMS.** Les scores de classification sont indiqués en précision, qui correspond à $precision = 1 - risque$ où le risque est la proportion d'erreurs faites par le classificateur.

Algorithme	Paramètres	Précision empirique	Précision de test
Random Forest	Estimateurs = 1	83.33%	54.17%
Décision Tree	Profondeur = 2, exemples séparés = 2	100%	91.67%
SVM Linéaire	$C = 9.103 \times 10^{-11}$	100%	54.17%
SVM Noyau RBF	$C = 1 \times 10^{-12}$, $\gamma = 1 \times 10^{-12}$	58.33%	58.33%
SCM	Conjonction, Attributs = 1, $p = 0.01$	100%	95.83%

Les résultats de la classification homme-femme sur les plasmas acquis en LCMS sont présentés à la table 2.14. On remarque que, pour l'algorithme des forêts d'arbres aléatoires, on a le paramètre du nombre d'estimateurs fixé à 1. Cela signifie que ce paramètre a le meilleur risque de validation croisée, ou du moins qu'aucun autre nombre d'estimateurs essayé n'a pu avoir un meilleur risque. Avec ce paramètre, l'algorithme a une assez mauvaise performance sur l'ensemble de tests avec seulement 54% de précision. Les SVM, malgré le très large espace de paramètres exploré, ont aussi de mauvaises performances. Dans le cas du noyau linéaire, le classificateur a pu apprendre sur l'ensemble d'entraînement, mais pas généraliser sur l'ensemble de tests. Pour le noyau RBF, on remarque que les

plus petites valeurs de l'espace ont été choisies et que l'algorithme n'a pas appris ni sur l'ensemble de tests ni sur l'ensemble d'entraînement. Il existait des combinaisons de paramètres où le classificateur pouvait atteindre de bonnes performances sur l'ensemble d'entraînement. Par contre, le risque de validation croisé est égal ou pire que celui des paramètres sélectionnés ici. Il semble alors logique de conclure que le SVM à noyau RBF n'arrive pas à apprendre de classificateur qui généralise sur ce problème.

Les résultats sont par contre bien meilleurs avec les algorithmes parcimonieux de l'arbre de décision et du SCM. Dans le cas de l'arbre de décision, le classificateur utilise deux attributs (pics) pour faire la prédiction. On arrive à environ 92% de précision sur l'ensemble de tests, soit 2 erreurs. Le SCM quant à lui utilise un seul attribut afin de faire la classification. Après vérification, il s'agit du pic putatif de la testostérone. Avec une règle sur ce pic, le classificateur prédit parfaitement l'ensemble d'entraînement et fait une seule erreur sur l'ensemble de tests. Une vérification des données démontre alors qu'un exemple aberrant est présent dans l'ensemble de tests, c'est-à-dire que l'on a un exemple "masculin" dont l'intensité du pic de testostérone est inférieure à l'intensité du pic de testostérone sur deux exemples "féminin". Il est également possible que cela résulte d'une erreur de mesure sur ce pic dans le spectre "masculin". Le classificateur de l'arbre de décision utilise aussi le pic putatif de testostérone dans ses règles de classification.

Cette expérience semble donc indiquer que l'apprentissage automatique peut effectivement être appliqué à des spectres de masse d'échantillons de plasma et apprendre sur des composantes intrinsèques du plasma. De plus, les résultats de cette expérience suggèrent que les algorithmes parcimonieux sont mieux adaptés à ce type de problème.

2.4 Conclusion

En conclusion de ce chapitre, l'apprentissage automatique a été testé sur des données provenant de spectrométrie de masse. Malgré certaines embûches, on arrive à la conclusion que l'application de l'apprentissage automatique est possible sur ce type de données.

Malgré de premiers tests inconclusifs, une série d'autres tests sur des ensembles de données et également l'ajout de la technologie des VLM a prouvé que plusieurs algorithmes d'apprentissage déjà existant sont compatibles avec l'apprentissage automatique et capable de prédire sur des spectres acquis sur des échantillons de plasma et des petits métabolites non protéiques qui s'y retrouvent. De plus, plusieurs de ces tests suggèrent que non seulement les algorithmes utilisant des modèles parcimonieux ont la capacité d'apprendre sur ce type de données, mais que ce sont les algorithmes ayant le plus de facilité à apprendre et à généraliser.

De plus, une nouvelle méthode à noyau a été conçue spécifiquement avec les défis de la spectrométrie de masse avec source LDTD. Lorsque testé sur un jeu de données, ce nouveau noyau a eu de bonnes performances quant à l'identification de l'ajout d'un des deux composés, mais n'a pas bien performé

pour le second. Au total, cette méthode reste encore une piste à explorer.

Il y a de multiples avenues afin de continuer les travaux présentés dans ce chapitre. Au niveau du noyau à boîtes chevauchantes (OBK), plusieurs variantes sont imaginables et qui pourraient améliorer ses performances. Par exemple, on pourrait remplacer la simple somme que l'on fait des pics dans chaque boîte individuelle pour une autre forme de comparaison et de calcul, telle que la somme des logarithmes de l'intensité des pics, la conservation du seul pic maximum ou l'introduction de fonctions non linéaires telles que des sigmoïdes.

D'autres avenues de recherches sont aussi ouvertes dans d'autres champs de recherche de l'apprentissage automatique. Une telle branche est celle de la détection d'anomalie. Alors que les classificateurs de détection d'anomalie sont typiquement très difficiles à entraîner, on ne peut nier qu'il serait intéressant de voir ce que ces algorithmes peuvent faire. Ils apporteraient aussi un bénéfice en ce qui concerne le travail en laboratoire, puisqu'on recherche un grand nombre d'échantillons normaux au lieu de rechercher à faire des ensembles de données relativement équilibrés entre des exemples positifs et négatifs, ce qui peut être ardu lorsque la classe négative est composée d'échantillons correspondant à des pathologies.

Conclusion

En conclusion, plusieurs travaux ont été présentés dans ce mémoire sur différents aspects d'un projet concernant l'analyse automatisée de spectres de masses à large spectre sur des échantillons de plasma. Le but du projet était de prédire un certain phénotype à partir d'un spectre à large bande. L'intérêt de ces spectres à large bande est qu'on peut ainsi tirer de l'information d'un grand nombre de métabolites de l'échantillon, au lieu d'un petit nombre de molécules identifiées. Le premier de ces aspects est le traitement des données provenant de spectres de masse. Les travaux présentés ici étaient spécifiques à la correction de variations dans les ratios de masse sur charge de spectres.

Le premier de ces traitements est la mise au point de l'algorithme de correction par masses de verrouillage virtuelles. De multiples résultats présentés au chapitre 1 établissent l'efficacité de cette méthode afin de corriger les variations observées chez les masses des pics d'un spectre. Ces variations, généralement dues au recalibrations de l'instrument entre les acquisitions de spectres, ajoutent un bruit sur l'aze m/z des pics. Il est nécessaire de réduire ces variations au maximum afin d'éviter des biais lors de l'alignement des pics. Cette méthode a plusieurs impacts potentiels. Premièrement, elle nous permet d'éliminer des biais reliés à des facteurs extérieurs dans des spectres de masse avant une analyse statistique ou une expérience d'apprentissage automatique. De plus, cette méthode peut remplacer la méthode classique de masses de verrouillage ajoutées au spectre dans des cas où cette méthode classique est difficilement applicable. Alors que la correction par masses de verrouillage virtuelles nécessite une certaine structure du spectre et des attentes au niveau des pics qui y seront présents, cette méthode est tout de même avantageuse quand ces conditions sont remplies. Les masses de verrouillage réelles nécessitent d'ajouter plusieurs molécules à chacun des échantillons avant l'acquisition des spectres. De plus, il faut utiliser des molécules qui auront des masses détectées qui sont à des endroits isolés du spectre et à des intervalles réguliers. La méthode par masses de verrouillage virtuelles est donc avantageuse au niveau des coûts expérimentaux. Elle peut aussi détecter et utiliser beaucoup plus de points de correction que la méthode classique. Finalement, cette méthode n'est pas compatible exclusivement avec des spectres générés avec une source LDTD, mais avec toutes sortes de spectres de masse.

Le second traitement est l'algorithme d'alignement des pics. Encore une fois, plusieurs résultats indiquent que cette méthode rends effectivement les spectres plus comparables, tel que montré à la figure 1.5 et aux tables 1.17 à 1.19. Cet algorithme nous permet de corriger les légères erreurs de

lecture du détecteur ainsi que d'autres erreurs pouvant s'ajouter à l'étape de la centroïdation des pics. Cette méthode couplée à la méthode de correction par masses de verrouillage virtuelles permet de faire des analyses statistiques et par apprentissage automatique sur de grands nombres de spectres de masse sans perdre de la précision du spectromètre.

Un aspect futur avec ces deux algorithmes sera de chercher à les appliquer à d'autres contextes. Toutes les données utilisées au cours des travaux présentés dans ce mémoire ont été générées à partir d'un spectromètre de masse à haute résolution. Il serait donc intéressant de voir le comportement des algorithmes proposés sur des données générées à partir d'instrument de moindre précision sur l'axe m/z. De plus, les ensembles de données utilisés avec ces algorithmes de correction et d'alignement étaient tous générés avec un système LDTD comme source d'ionisation. Il serait alors intéressant d'appliquer ces algorithmes à d'autres types de sources d'ionisation qui sont également utilisés avec des spectromètres de masse de type temps de vol. Une de ces sources, la technologie du MALDI-ToF, est particulièrement répandue dans le domaine de la protéomique. On peut également considérer tester expérimentalement les bénéfices de l'application à la fois des masses de verrouillage virtuelles et des masses de verrouillage classiques.

D'autres travaux ont été présentés au cours de ce mémoire au niveau de l'apprentissage automatique. Dans certaines des expériences tentées (table 2.7), aucun résultat probant n'a été obtenu. Dans d'autres cas (table 2.8 à la table 2.13, table 2.14), l'application de l'apprentissage automatique fut un succès. Premièrement, la mise au point mentionnée précédemment des méthodes de correction des masses était une étape critique dans l'avancée du projet. Ensuite, il a été établi sur plusieurs ensembles de données que l'application de l'apprentissage automatique à des spectres de masse était possible et fonctionnelle. On note en particulier qu'un des ensembles de données consistait à faire la différence entre deux classes de donneurs à partir du contenu en petits métabolites (non protéiques) du plasma. De plus, les résultats de ces expériences suggèrent fortement que les algorithmes parcimonieux tels que l'arbre de décision et la machine à couverture d'ensembles soient très performants quand appliqués à des problèmes de spectrométrie de masse.

Une nouvelle méthode d'apprentissage automatique adaptée aux défis de la spectrométrie de masse, le noyau à boîtes chevauchantes, a aussi été développée au cours de ce projet. Bien que non parcimonieuse, cette méthode a prouvé son potentiel pour certaines applications. Il y a d'ailleurs encore plusieurs améliorations possibles à explorer à cette méthode. Bien développée, cette méthode pourrait se révéler intéressante à la communauté de recherche en apprentissage automatique et en spectrométrie de masse.

Plusieurs autres aspects des travaux présentés ici ont encore un potentiel de développement. Un premier de ces aspects se retrouve dans le traitement des données. Malgré le travail important apporté à la correction des masses, l'incertitude sur la vraie valeur de l'intensité d'un pic donné reste grande. Il est certain que des travaux futurs porteront à développer des méthodes de normalisation de l'intensité des pics dans un spectre afin de pouvoir faire de meilleures comparaisons entre les échantillons d'un

ensemble de données.

D'autres aspects de l'apprentissage automatique seront aussi à développer dans des travaux futurs. Une piste intéressante à explorer serait l'application d'algorithmes de détection d'anomalie. Malgré la complexité de ce type d'algorithmes, ils représentent un avantage certain au niveau expérimental. Au lieu de générer des ensembles de données avec un grand nombre d'exemples de chaque classe et de tenter de représenter chaque classe de manière relativement équilibrée, ces algorithmes ne nécessitent qu'un grand nombre d'exemples normaux. Cela est une propriété intéressante en recherche biomédicale vu qu'il peut être très difficile d'avoir un grand nombre d'exemples négatifs, anormaux ou pathologiques. Des travaux sont aussi à envisager en termes d'adaptations à d'autres algorithmes existants afin de les rendre plus performants sur les problèmes reliés à la spectrométrie de masse. Un exemple d'une telle adaptation serait à la manière de définir les souches de décision utilisées par l'algorithme de la machine à couverture d'ensembles (SCM).

Il est aussi à noter que les travaux de ce mémoire portent entièrement sur des résultats reliés à des échantillons de plasma sanguin. Comme mentionné dans l'introduction, un don de sang va permettre d'avoir du plasma sanguin, un culot cellulaire et des plaquettes. Le contrôle de qualité de ces deux derniers produits fait aussi partie du projet à long terme et des travaux se porteront sans doute sur ces nouveaux types d'échantillons. Des travaux futurs porteront aussi à d'autres problèmes de classification sur des spectres de masse de produits sanguins. Entre autres, un projet portera sur le diagnostic de maladies auto-immunes dans des échantillons sanguins. Il serait également possible d'éventuellement étendre les applications de la spectrométrie de masse par LDTD et l'apprentissage automatique afin de faire la détection d'infections virales et bactériennes.

Annexe A

Annexe

A.1 Données supplémentaires au chapitre 1

TABLE A.1 – Variance de l'intensité du pic de clomiphène par individu dans l'expérience de reproductibilité

Individu	Moyenne d'intensité	Écart type	Écart type (en %)
1	37817	6374	16.85%
2	31742	9939	31.31%
3	39499	5994	15.18%
4	37380	6770	18.11%
5	41891	7089	16.92%
6	36218	4825	13.32%
7	35771	5940	16.61%
8	34801	4910	14.11%

TABLE A.2 – Table de consensus de pics - Consensus à l'intérieur de plaques de calibrations différentes. RLM dénote la correction par masse de verrouillage réelle (*Real Lock Mass*) et VLM dénote par masse de verrouillage virtuelle. ACN-pos dénote la condition d'extraction à l'ACNMeOH et l'acquisition des ions positifs. Chl-neg dénote l'extraction au chlorobutane et l'acquisition des ions négatifs.

Plaques	ACN-pos, RLM	ACN-pos, VLM	Chl-neg, RLM	Chl-neg, VLM
27/31	8	187	0	186
27/34	0	382	0	177
30/31	3	207	0	167
30/27-2	0	392	0	184
30/31-2	0	361	0	180
31/39	0	218	0	146
31/27-2	0	257	0	200
34/27-2	0	510	0	198
34/31-2	0	438	0	177
39/27-2	0	418	0	148
39/31-2	0	419	0	140

Bibliographie

Boser, B. E., I. M. Guyon et V. N. Vapnik. 1992, «A training algorithm for optimal margin classifiers», dans *Proceedings of the fifth annual workshop on Computational learning theory*, ACM, p. 144–152.

Breiman, L. 2001, «Random forests», *Machine learning*, vol. 45, n° 1, p. 5–32.

Breiman, L., J. Friedman, R. Olshen, C. Stone, D. Steinberg et P. Colla. 1983, «Cart : Classification and regression trees», *Wadsworth : Belmont, CA*, vol. 156.

Chang, C.-C. et C.-J. Lin. 2011, «Libsvm : A library for support vector machines», *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, n° 3, p. 27.

Cortes, C. et V. Vapnik. 1995, «Support-vector networks», *Machine learning*, vol. 20, n° 3, p. 273–297.

Fan, R.-E., K.-W. Chang, C.-J. Hsieh, X.-R. Wang et C.-J. Lin. 2008, «Liblinear : A library for large linear classification», *The Journal of Machine Learning Research*, vol. 9, p. 1871–1874.

Ge, G. et G. W. Wong. 2008, «Classification of premalignant pancreatic cancer mass-spectrometry data using decision tree ensembles», *BMC bioinformatics*, vol. 9, n° 1, p. 275.

Guo, L., M. V. Milburn, J. A. Ryals, S. C. Lonergan, M. W. Mitchell, J. E. Wulff, D. C. Alexander, A. M. Evans, B. Bridgewater, L. Miller et collab.. 2015, «Plasma metabolomic profiles enhance precision medicine for volunteers of normal health», *Proceedings of the National Academy of Sciences*, vol. 112, n° 35, p. E4901–E4910.

Héma-Québec. 2015, «Rapport annuel 2014-2015 d'héma-québec», .

Marchand, M. et J. S. Taylor. 2002, «The set covering machine», *The Journal of Machine Learning Research*, vol. 3, p. 723–746.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot et E. Duchesnay. 2011, «Scikit-learn : Machine learning in Python», *Journal of Machine Learning Research*, vol. 12, p. 2825–2830.

Psychogios, N., D. D. Hau, J. Peng, A. C. Guo, R. Mandal, S. Bouatra, I. Sinelnikov, R. Krishnamurthy, R. Eisner, B. Gautam et collab.. 2011, «The human serum metabolome», *PloS one*, vol. 6, n° 2, p. e16 957.

Shin, H. et M. K. Markey. 2006, «A machine learning perspective on the development of clinical decision support systems utilizing mass spectra of blood samples», *Journal of Biomedical Informatics*, vol. 39, n° 2, p. 227–248.

Swan, A. L., A. Mobasheri, D. Allaway, S. Liddell et J. Bacardit. 2013, «Application of machine learning to proteomics data : classification and biomarker identification in postgenomics biology», *Omics : a journal of integrative biology*, vol. 17, n° 12, p. 595–610.

Tibshirani, R., T. Hastie, B. Narasimhan, S. Soltys, G. Shi, A. Koong et Q.-T. Le. 2004, «Sample classification from protein mass spectrometry, by 'peak probability contrasts'», *Bioinformatics*, vol. 20, n° 17, p. 3034–3044.

Vapnik, V. 1995, *The nature of statistical learning theory*, Springer Science & Business Media.

West, P. R., D. G. Amaral, P. Bais, A. M. Smith, L. A. Egnash, M. E. Ross, J. A. Palmer, B. R. Fontaine, K. R. Conard, B. A. Corbett et collab.. 2014, «Metabolomics as a tool for discovery of biomarkers of autism spectrum disorder in the blood plasma of children», *PloS one*.

Xu, X., J. Lan et W. A. Korfmacher. 2005, «Rapid lc/ms/ms method development for drug discovery», *Analytical chemistry*, vol. 77, n° 19, p. 389–A.

Zang, X., C. M. Jones, T. Q. Long, M. E. Monge, M. Zhou, L. D. Walker, R. Mezencev, A. Gray, J. F. McDonald et F. M. Fernandez. 2014, «Feasibility of detecting prostate cancer by ultraperformance liquid chromatography–mass spectrometry serum metabolomics», *Journal of proteome research*, vol. 13, n° 7, p. 3444–3454.