Méthodes de Machine Learning appliquées à des données de périnatalité

Flavien Alonzo

CENTRALE NANTES

INSTITUT PASCAL
sciences de l'ingénierie et des systèmes

Méthodes de Machine Learning appliquées à des données de périnatalité

Flavien Alonzo

Encadrants : Emmanuelle Abisset-Chavanne (ECN) & Dr Lemlih Ouchchane (UCA)
Lieux du stage : Laboratoire I.C.I. & CHU Clermont-Ferrand U.B.I.M.

3 avril 2018 - 17 août 2018

REMERCIEMENTS

Je tiens à remercier les personnes suivantes pour leurs interventions lors de mon stage :

— Anaïs Barasinski pour m'avoir proposé et accepté pour effectuer ce stage.

— Emmanuelle Abisset-Chavanne pour avoir été ma tutrice dans les locaux de Nantes lors de mon premier mois de stage.

— Lemlih Ouchchane pour avoir été mon tuteur sur Clermont-Ferrand, pour son sérieux, ses compétences et connaissances qu'il m'a apportés tout au long du stage.

— Le reste de l'équipe de Bio-informatique (Adrian Kriche, Aline Guttmann et Sylvie Roux) pour l'accueil à la faculté de médecine de Clermont-Ferrand.

— Le reste de l'équipe PEPRADE (notamment Anne Legrand et Didier Lemery) pour l'intérêt porté à mon stage.

RÉSUMÉ

Cette collaboration Institut Pascal - Ecole Centrale de Nantes s'appuie sur les données disponibles au sein de PEPRADE (Périnatalité, grossesse, Environnement, PRAtiques médicales et DEveloppement) concernant une cohorte de suivis de (23 000) grossesses s'étalant sur la décennie en cours, avec de forts enjeux cliniques et de santé publique.

La typologie des données est très diverse incluant des données numériques structurées en bases, des données d'imagerie essentiellement échographiques, des signaux électrophysiologiques et des informations textuelles. Ces données sont massives quant à leur taille car elles impliquent plusieurs individus statistiques dont la parturiente, son conjoint, la grossesse en elle-même (depuis la conception jusqu'à l'accouchement, ses conditions et son environnement médical), le ou les fœtus en question, ainsi que l'enfant en période néonatale. L'objet de cette collaboration est de mettre en œuvre l'ensemble d'une chaîne de traitement de données depuis leur extraction/préparation jusqu'à leur exploitation avec un focus particulier sur quelques problématiques cliniques relatives au développement staturo-pondéral fœtal, et à la morbidité périnatale. Outre l'intégration des données, je propose la mise en œuvre d'un catalogue quasi-complet de techniques de machine learning dédiées à l'exploitation de la valeur informationnelle de ces données périnatales dans le but de prédire/anticiper des issues telles que le poids de naissance d'un enfant, le caractère morbide de son développement staturo-pondéral (hypotrophie/macrosomie) et les conditions de sa naissance (accouchement compliqué, réanimation néonatale, etc).

Ces études montrent des résultats plus satisfaisants que ceux obtenus avec les méthodes actuelles et sont résumés sous-formes de diagramme pour pouvoir évaluer et comparer chaque modèle selon le jeu de données et la méthode de régression / classification utilisée.

TABLE DES MATIÈRES

TABLE DES MATIÈRES

TABLE DES FIGURES

LISTE DES TABLEAUX

Introduction

Mon stage marque le commencement d'une collaboration entre l'Ecole Centrale de Nantes et L'Institut Pascal, plus particulièrement avec l'équipe PEPRADE. L'enjeu de cette collaboration est de pouvoir rassembler plusieurs domaines de compétences sur un même sujet d'étude. Plus particulièrement ici, le sujet d'étude concerne plusieurs aspects de la périnatalité. Je m'intéresse ici à la prédiction des différentes données liées à la morbidité d'un accouchement pour anticiper les différentes complications qui peuvent arriver le jour de l'accouchement.

L'objectif du stage a donc été de mettre en place des méthodes de Machine Learning sur des données de périnatalité dans la prédiction de morbidités liées à l'accouchement. La première partie de ce stage s'est déroulée au sein du bâtiment T de l'Ecole Centrale de Nantes sous la tutelle d'Emmanuelle Abisset-Chavanne où l'objectif a été d'acquérir et d'expérimenter les différentes méthodes de Machine Learning existantes. La deuxième partie s'est déroulée dans l'unité U.B.I.M. (Unité de Biostatistique et Informatique Médicale) de l'UFR de médecine de Clermont-Ferrand sous la tutelle de Lemlih Ouchchane où les objectifs étaient de mettre en place les différentes méthodes de Machine Learning sur des données réelles de périnatalité pour pouvoir faire de la prédiction de morbidité.

Le stage a donc duré 20 semaines : du 3 avril au 17 août 2018. Il a permis de mettre en application les connaissances acquises pendant ma deuxième année à l'Ecole Centrale de Nantes dans l'option BioStic (Sciences du Numérique pour les Sciences de la Vie et de la Santé).

CHAPITRE 1

GÉNÉRALITÉS DU STAGE

1.1 PRÉSENTATION DE L'ENTREPRISE

Je suis intervenu dans le cadre d'une collaboration entre l'Institut Pascal [1] et l'Ecole Centrale de Nantes [2](Institut de Calcul Intensif). Sur la figure1.1, on retrouve une partie de l'organigramme de L'Institut Pascal. Celui-ci regroupe plusieurs domaines de recherche dont la partie Thérapies Guidées par l'Image (TGI) qui abrite l'équipe PEPRADE (Périnatalité, grossesse, Environnement, PRAtiques médicales et DEveloppement), entité qui m'a recruté pour effectuer mon stage.

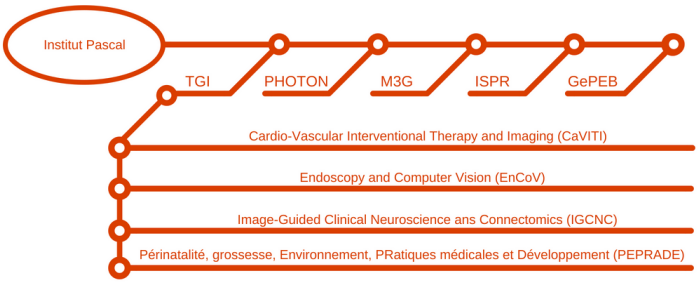

FIGURE *1.1* – *Organigramme de l'Institut Pascal*

PEPRADE est une équipe de 14 enseignants-chercheurs, 3 personnels techniques et 3 doctorants qui est consacrée à l'évaluation médicale dans le contexte de la périnatalité (pré-grossesse, accouchement, période néo-natale).

Cette recherche contribue à un objet de Santé Publique car elle permet d'identifier des éléments impactant le déroulement de la grossesse et l'avenir de l'enfant et de proposer des pratiques plus adaptées. Elle comporte des sous-objectifs translationnels notamment concernant le diagnostic pendant la grossesse.

L'évaluation médicale, qui est la spécificité de l'Institut Pascal, est orientée dans deux directions :

— évaluer une « intervention » sur le patient (une pratique professionnelle, un type d'imagerie, une stratégie de prise en charge ou une action de prévention) ;

1. http://www.institutpascal.uca.fr/index.php/fr/
2. https://www.ec-nantes.fr/

— évaluer un (des) risque(s) au(x)quel(s) est potentiellement exposé la mère ou l'enfant (toxicité d'un produit, effet physique des ultrasons de cisaillement, environnement, précarité).

L'interfaçage transdisciplinaire clinique, épidémiologique, mathématique, économique, statistique, sociologique permet de mieux cerner les indicateurs à évaluer et de construire le design des études à mener.

Le Dr Lemlih Ouchchane qui était mon encadrant sur Clermont-Ferrand fait partie de cette équipe.

L'Ecole Centrale de Nantes est une école d'ingénieurs généralistes qui accueille 2500 étudiants chaque année. On retrouve parmi eux les élèves issus des cursus de classes préparatoires aux grandes écoles scientifiques mais une partie des étudiants est recrutée après des licences ou maîtrises scientifiques ou dans le cadre d'échanges internationaux. Sur la figure1.2, on peut observer les différents laboratoires de recherche présents au sein de l'Ecole Centrale de Nantes.

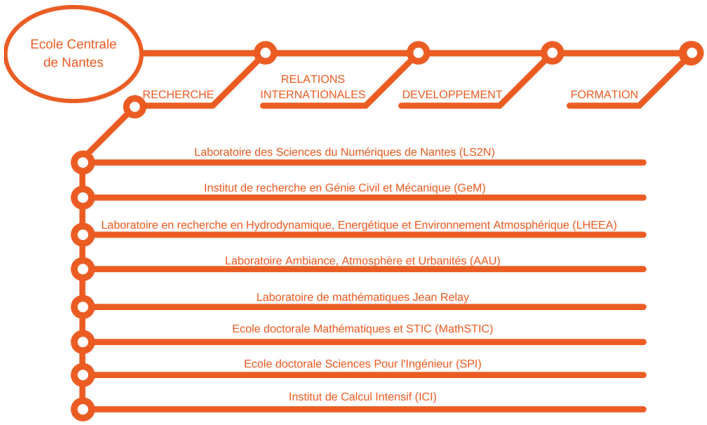

FIGURE *1.2* – *Organigramme de l'Ecole Centrale de Nantes*

Le recours toujours plus systématique au calcul intensif est lié à la fois au progrès continuel des puissances de calcul disponibles et aux compétences en calcul parallèle, c'est-à-dire la maîtrise des méthodes mathématiques et algorithmiques, en permanente évolution avec les architectures. L'équipe actuelle de l'Institut de Calcul Intensif, est une des spécialistes de niveau international en maillage et adaptation massivement parallèle, et participe régulièrement à des conférences invitées et à l'organisation de symposiums dans les grandes conférences sur ces thématiques de recherche.

Mon encadrante sur Nantes, Emmanuelle Abisset Chavanne, fait partie de ce laboratoire de recherche.

1.2 ENJEUX CLINIQUES DU STAGE

Il est important de prêter attention au développement staturo-pondéral d'un enfant puisque c'est un proxy direct de son développement et donc de sa santé. Un très bon indicateur du développement staturo-pondéral est le poids fœtal (à terme), il est donc stratégique d'essayer de prédire le poids fœtal le plus tôt possible avant l'accouchement pour anticiper tous les risques de morbidité qui sont liés directement à l'accouchement ou tout autre risque post-natal que l'on pourrait anticiper.

De ce point de vue, il est important d'être capable de prédire des morbidités comme la nécessité d'une réanimation, la macrosomie (poids élevé vis-à-vis de l'âge gestationnel) et le pronostic vital

d'un enfant (vivant ou mort-né).

Des études [**1**] et [**2**] ont déjà été réalisées autour de la prédiction de la macrosomie, et les résultats montrent que les méthodes de détection actuelles sont très inefficaces pour distinguer les enfants macrosomes des non-macrosomes (Dice [3] de 6,2% et 7,6% respectivement). C'est pour cela qu'il est pertinent d'essayer d'améliorer ces résultats avec de nouvelles approches.

L'objectif de mon stage a donc été de mettre en place différentes méthodes algorithmiques de Machine Learning pour effectuer une prédiction de variables liées à la morbidité à partir de données de périnatalité renseignées au cours de la grossesse pour pouvoir anticiper ces morbidités et adapter les soins au cours de la grossesse. Mais l'objectif est aussi une opportunité de mettre en œuvre tout un arsenal de méthodes statistiques d'apprentissage automatique afin de faire le point sur leurs objectifs, leurs interprétations et enfin leur utilité pratique.

1.3 MISSIONS DU STAGE

Les missions de mon stage ont été multiples et adaptées au fur et à mesure de l'avancement du stage :

- Se renseigner sur le Machine Learning, comprendre les méthodes qui existent et les mettre en place dans un langage informatique (R) ;
- Se familiariser avec les données de périnatalité disponibles et effectuer un traitement de ces données pour pouvoir les utiliser dans les algorithmes de prédiction ;
- Mettre en place un ensemble de méthodes de régression pour prédire le poids fœtal à terme d'un enfant ;
- Mettre en place un ensemble de méthodes de classification pour prédire une variable binaire correspondant à la différenciation des enfants entre macrosome ou non-macrosome à terme ;
- Mettre en place un ensemble de méthodes de classification pour prédire une variable binaire correspondant à la nécessité, ou non, d'une réanimation postnatale ;
- Mettre en place un ensemble de méthodes de classification pour prédire une variable multimodale correspondant au pronostic postnatal d'un enfant ;
- Effectuer une analyse critique de mes résultats vis-à-vis des méthodes déjà existantes.

J'ai effectué la première mission de ce stage à Nantes au sein du laboratoire I.C.I. puis j'ai effectué le reste des missions directement au sein de l'unité U.B.I.M. (Unité de Biostatistiques et d'Informatique Médicale) du CHU de Clermont Ferrand.

J'ai eu l'occasion d'effectuer une présentation de mon travail lors d'une journée scientifique organisée par l'axe T.G.I. de l'Institut Pascal, qui s'est déroulée le vendredi 15 juin 2018. Ma présentation a eu pour but de donner un exemple concret de travail sur le thème applicatif de l'intelligence artificielle.

3. voir la définition du coefficient : Annexe B page 36

CHAPITRE 2

AUTOUR DU MACHINE LEARNING

2.1 DESCRIPTION DES DONNÉES

Pendant mon stage, j'ai en particulier utilisé le logiciel R pour écrire et exécuter mes programmes. Pour aider la visualisation et la lisibilité du code et ses résultats, j'ai utilisé des fichiers Rmarkdown et Shiny.

Les données que j'ai utilisées dans mon stage proviennent de bases de données réelles donnant des informations sur l'ensemble des grossesses qui ont été gérées par le CHU de Clermont-Ferrand (Estaing) entre 2010 et 2017. Ces données regroupent des informations sur environ 23 000 grossesses sur des variables issues de données échographiques, de données de consultation, de caractéristiques anthropométriques de la mère, des conditions d'accouchements avec les différentes morbidités associées et des caractéristiques physiques de l'enfant à la naissance.

Je constitue alors une base de données pour rassembler les variables explicatives de l'étude. Je sélectionne uniquement des variables mesurables avant le jour de l'accouchement, sinon je ne peux pas faire de prédiction et donc anticiper la valeur de la variable que je cherche à prédire.Je regroupe alors :

- données relatives à la maman :
  - l'IMC usuel de la mère (avant la grossesse) ;
  - si la mère fumait avant la grossesse ;
  - si la mère a fumé pendant la grossesse ;
  - l'origine géographique de la mère ;
  - le niveau d'étude de la mère ;
  - la profession de la mère ;
  - si la mère a des antécédents HTA ;
  - si la mère a des antécédents diabète ;
  - l'âge de la mère pendant le second trimestre de la grossesse.

- données relatives à la grossesse :
  - la date de début de grossesse ;
  - si la mère est atteinte d'HTA lors du deuxième et troisième trimestre ;
  - si la mère est atteinte du diabète lors du deuxième et troisième trimestre.
- données relatives à l'enfant :
  - le sexe de l'enfant.
- données relatives à l'échographie :
  - le diamètre bipariétale du fœtus lors de la deuxième et troisième échographie ;
  - le périmètre crânien du fœtus lors de la deuxième et troisième échographie ;
  - le périmètre abdominal du fœtus lors de la deuxième et troisième échographie ;
  - la longueur fémorale du fœtus lors de la deuxième et troisième échographie ;
  - la taille du cervelet du fœtus lors de la deuxième et troisième échographie ;
  - le volume de liquide amniotique du fœtus lors de la deuxième et troisième échographie.

Le nouveau fichier de données a une structure de matrice $X \in M_{np}$, avec $n \approxeq 10\,000$ grossesses uniques où les données sont disponibles et $p = 27$ variables explicatives.

2.1.1 DESCRIPTION DU POIDS FŒTAL

La première mission de mon stage consiste à prédire la valeur du poids fœtal à terme d'un enfant. Cela revient à faire une régression pour prédire une variable qui peut prendre n'importe quelle valeur dans $\mathbb{R}^{+*}$.

Sur la figure 2.1, on peut observer la distribution des valeurs du poids fœtal pour toutes les grossesses présentes dans le jeu de données. La distribution a une allure de gaussienne avec 3270g comme valeur moyenne.

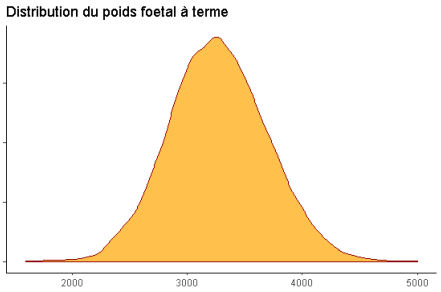

FIGURE 2.1 – *Distribution du poids foetal*

Dans le domaine de la maïeutique, il existe un très grand nombre de méthodes ([**3**] et [**4**] par exemple) pour prédire le poids fœtal en fonction, notamment, de paramètres de croissance et d'attributs relatifs à la maman. Parmi toutes les formules qui existent, il y a une formule qui est toujours nettement la plus utilisée actuellement : une des formules de Hadlock à 3 paramètres [**5**].

2.1.2 DESCRIPTION DE LA CLASSIFICATION RELATIVE À LA MACROSOMIE

La deuxième mission de mon stage consiste à prédire la caractéristique macrosome (poids fœtal $> 4000g$ ou poids fœtal $> 95^{e} Percentile$) ou non d'un enfant. Cela revient à faire une classification pour prédire une variable qui peut prendre la classe {macrosome} ou {non-macrosome}. Ici, je définis les macrosomes selon le critère poids fœtal $> 4000g$ parce que j'ai restreint le fichier aux enfants nés à terme, puisque mon objectif est de prédire le poids à terme à partir de données disponibles avant le terme.

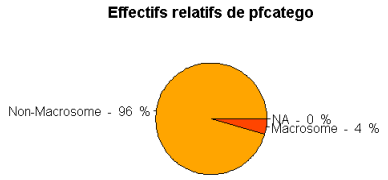

FIGURE 2.2 – *Distribution de la macrosomie*

Sur la figure 2.2, on peut observer la répartition des enfants parmi la catégorie {macrosome} et {non-macrosome}, représentant respectivement 4% et 96% de la cohorte (0% de données manquantes).

Il est possible de donner un statut-staturo-pondéral, à terme, plus précis que macrosome/non-macrosome à un enfant en fonction de son poids fœtal à terme par rapport au reste de la population :

- Macrosome : regroupe les 5% des enfants des poids fœtaux les plus élevés, cela correspondant aussi à la condition $poidsFœtal > 4000g$ (on peut retrouver les deux définitions). Il est important de trouver les macrosomes parce qu'ils réclament plus d'attention le jour de l'accouchement dû aux risques accrus de complications ;
- Hypotrophe : regroupe les 5% des enfants des poids fœtaux les moins élevés. Ces enfants correspondent quasi-intégralement aux enfants nés prématurement, ce qui explique qu'ils aient un poids fœtal à terme inférieur aux autres ;
- Eutrophique : regroupe tous les enfants qui ne sont ni macrosomes ni hypotrophes.

2.1.3 DESCRIPTION DE LA CLASSIFICATION RELATIVE À LA RÉANIMATION

La troisième mission de mon stage consiste à prédire la nécessité ou non d'une réanimation d'un enfant après l'accouchement. Cela revient à faire une classification pour prédire une variable qui peut prendre la classe {réanimation} ou {pas-de-réanimation}.

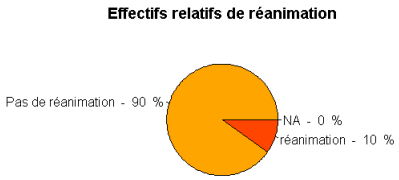

FIGURE 2.3 – *Distribution de la réanimation*

Sur la figure 2.3, on peut observer la répartition des enfants parmi la catégorie {réanimation} et {pas-de-réanimation}, représentant respectivement 10% et 90% de la cohorte (0% de données manquantes).

Environ 10% des nouveau-nés nécessitent une assistance respiratoire à la naissance. Moins de 1% ont besoin d'une réanimation complexe. Les étiologies sont nombreuses, mais la plupart impliquent une anoxie ou un trouble respiratoire.

2.1.4 Description de la classification relative au pronostic vital d'un fœtus

La quatrième mission de mon stage consiste à prédire le pronostic vital d'un enfant. Cela revient à chercher à faire une classification pour prédire une variable qui peut prendre la classe {0 : Vivant} ou {1 : Décédé avant travail} ou {2 : Décédé pendant travail} ou {3 : Décédé sans précision chronologique} ou {4 : Décédé au décours IMG(Interruption Médicale de Grossesse)}.

Sur la figure 2.4, on peut observer la répartition des enfants parmi les catégories possibles, en remarquant que la quasi-totalité des enfants appartiennent à la catégorie {0}, ce qui est logique, et qu'aucune grossesse n'appartient à la catégorie {2}.

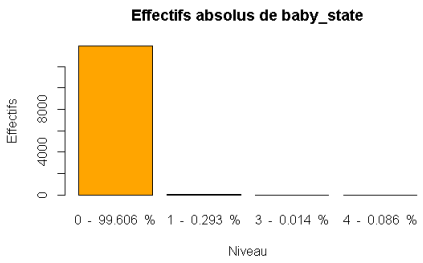

Figure 2.4 – *Distribution du diagnostic vital*

Dans son acception courante, on parle d'un enfant mort-né pour désigner tous les cas où, soit il sort du ventre de la mère en étant déjà mort, soit il meurt immédiatement à la naissance. Il s'agit de tous les cas où la déclaration de naissance à l'état civil n'a pu être faite avant le décès.

2.2 Méthodes de Machine Learning

Le Machine Learning, champ d'étude de l'intelligence artificielle, concerne la conception, l'analyse, le développement et l'implémentation de méthodes permettant à une machine (au sens large) d'évoluer par un processus systématique, et ainsi de remplir des tâches difficiles ou problématiques par des moyens algorithmiques plus classiques.

Les algorithmes utilisés permettent, dans une certaine mesure, à un système piloté par ordinateur (un robot éventuellement), ou assisté par ordinateur, d'adapter ses analyses et ses comportements en réponse, en se fondant sur l'analyse de données empiriques provenant d'une base de données ou de capteurs.

La difficulté réside dans le fait que l'ensemble de tous les comportements possibles compte tenu de toutes les entrées possibles devient rapidement trop complexe à décrire (on parle d'explosion combinatoire). On confie donc à des programmes le soin d'ajuster un modèle pour simplifier cette complexité et de l'utiliser de manière opérationnelle.

Un domaine utilisant beaucoup le Machine Learning est l'imagerie médicale : on cherche par exemple à automatiser la détection de tumeurs sur des images. Concrètement il faut avoir plusieurs images sur lesquelles sont annotées manuellement les tumeurs quand il y en a. Les différentes méthodes de Machine Learning vont donc chercher à relier les éléments graphiques de l'image (valeurs RGB des pixels par exemple) aux positions des tumeurs. Une fois un modèle construit, on peut donner de nouvelles images et le modèle va estimer la présence ou non de tumeur et éventuellement la position des tumeurs s'il y en a sur les nouvelles images.

J'invite le lecteur à regarder les parties annexes A B et C pour avoir l'explication résumée des méthodes de Machine Learning dont je parle dans les trois paragraphes suivant. ([**6**] , [**7**])

2.2.1 MÉTHODES DE RÉGRESSION

Le but d'une méthode de régression est de pouvoir trouver une fonction $\mathbf{f} : \mathbb{R}^p \to \mathbb{R}$ pour prédire une variable quantitative **y** à partir d'un jeu de données **x**, de **p** dimensions. Ces méthodes seront exclusivement dédiées pour prédire le poids fœtal. Pour évaluer une fonction **f**, on calcule la **RMSE** (Root Mean Squared Error) :

$$RMSE = \sqrt{\frac{1}{n_{TS}} \sum_{i \in \mathbf{TS}} (y_i - f(x_i))^2}$$

avec **TS** le jeu de données test et n_{TS} le nombre de grossesses dans **TS**. Ce coefficient s'interprète comme étant la valeur moyenne des erreurs qui sont commises lors de la prédiction du jeu test.

J'ai pu expérimenter 6 méthodes différentes pour effectuer une régression :

— RN : régression naïve (*Annexe A page 30*) ;

— RLR : régression linéaire régularisée (*Annexe A page 31*) ;

— RP : régression polynomiale (*Annexe A page 32*) ;

— RdN : réseau de neurones artificiels (*Annexe A page 32*) ;

— RF : forêts aléatoires (*Annexe A page 33*) ;

— GB : gradient boosting (*Annexe A page 34*).

2.2.2 MÉTHODES DE CLASSIFICATION

Une méthode de classification consiste à trouver une fonction $\mathbf{g} : \mathbb{R}^p \to \{0, 1, ...m-1\}$ qui attribue pour chaque grossesse **x**, une des **m** modalités de la variable $\mathbf{z_m}$. Ces méthodes seront exclusivement dédiées pour prédire l'appartenance à une classe d'une variable multimodale : macrosomie (binaire), réanimation (binaire), statut vital.

Pour évaluer une méthode de classification bimodale vis-à-vis de la catégorie 1 (par rapport à la catégorie 0), on calcule le coefficient de **Dice** :

$$Dice = \frac{TP}{TP + 1.3FN + 0.7FP}$$

avec $TP = card(\{\widetilde{z_m} = 1 \cap z_m = 1\})$, $FP = card(\{\widetilde{z_m} = 1 \cap z_m = 0\})$ et $FN = card(\{\widetilde{z_m} = 0 \cap z_m = 1\})$

Le déséquilibre entre les poids de FN et FP n'est justifiable que sur la base d'une estimation du coût d'une mauvaise décision, ici privilégiant la minimisation des "loupés" aux dépens des "fausses alertes", considérées moins coûteuses.

Pour évaluer une méthode de classification multimodale (avec $m > 2$), on calcule un pourcentage d'accord observé χ :

$$\chi = \frac{tr(M)}{sum(M)}$$

avec $M \in \mathscr{M}_{mm}(\mathbb{R})$ la matrice de contingence : $\forall i, j \in [\![1;m]\!]$, $M_{ij} = card(\{\widetilde{z_m} = i \cap z_m = j\})$

J'ai pu expérimenter 9 méthodes différentes de classification :

— classification naïve (*Annexe B page 37*) ;
— régression logistique (*Annexe B page 37*) ;
— SVM : machine à vecteurs de support (*Annexe B page 38*) ;
— kSVM : machine à vecteurs de support à noyau (*Annexe B page 38*) ;
— RdN : réseau de neurones artificiels (*Annexe B page 39*) ;
— bagging (*Annexe B page 40*) ;
— RF : forêts aléatoires (*Annexe B page 40*) ;
— boosting (*Annexe B page 42*) ;
— kNN : plus proches voisins (*Annexe B page 42*).

2.2.3 MÉTHODES DE RÉDUCTION DE DIMENSION

Une méthode de réduction de dimension consiste à trouver une fonction $\mathbf{h} : \mathbb{R}^p \to \mathbb{R}^d$. Le but d'une telle méthode est de pouvoir extraire les informations et d'enlever les corrélations du jeu de données initial en synthétisant les **p** variables en **d** dimensions. C'est une étape préalable aux approches précédentes. Ces méthodes sont qualifiées de non-supervisées car elles ne prennent pas en compte la variable à expliquer. Elles permettent aussi de réduire les temps des calculs algorithmiques nécessaires pour effectuer les modèles.

Il y a plusieurs façons de raisonner pour réduire les dimensions : raisonner sur la variance du jeu initial, raisonner sur la structure géométrique initiale du jeu de données et raisonner sur la conservation des voisinages.

J'ai pu expérimenter 6 techniques de Réduction de Dimension différentes :

- Raisonner sur la variance :
  — FAMD : analyse factorielle de données mixtes (*Annexe C page 43*) ;
  — kPCA : analyse en composantes principales à noyau (*Annexe C page 44*) ;
- Raisonner sur la structure géométrique :
  — MDS : échelle multidimensionnelle (*Annexe C page 44*) ;
  — Isomap (*Annexe C page 45*) ;
- Raisonner sur la conservation des voisinages :
  — LLE : plongement locale linéaire (*Annexe C page 45*) ;
  — t-SNE : plongement stochastique des voisins t-distribués (*Annexe C page 46*).

2.3 APPROCHE DE RÉSOLUTION

Pour répondre aux différentes missions, la marche à suivre reste la même peu importe la variable à prédire. Il faut de façon préliminaire créer les jeux de données qui utilisent les méthodes de réduction de dimension ; il faut savoir que les méthodes FAMD et kPCA ne me permettent pas de réduire le jeu de données initial en moins de 27 variables (nombre initial de variables) selon les critères de Kaiser-Guttman ou Karlis-Saporta-Spinaki, donc je n'ai pas retenu les jeux de données engendrées par ces méthodes.

Pour chaque méthode de régression ou classification (selon la nature de la variable à expliquer), j'optimise les hyperparamètres des modèles afin de minimiser la RMSE (régression) ou maximiser le Dice (ou χ) (classification).

Comme je crée mes modèles avec une répartition de 50% des grossesses dans le jeu d'apprentissage et 50% dans le jeu test, il y a possiblement introduction d'un biais dans le modèle : un modèle dépend des grossesses du jeu d'apprentissage donc le modèle est différent si le jeu d'apprentissage est différent, ce qui implique que la prédiction d'une valeur d'une grossesse peut être différente selon le jeu d'apprentissage. Pour compenser ce biais, je réalise 100 fois un modèle en prenant à chaque fois aléatoirement les 50% des grossesses du jeu d'apprentissage et celle du jeu test. Je peux alors calculer puis confronter la moyenne des RMSE, Dice ou χ pour comparer les différentes méthodes de Machine Learning sur les jeux de données disponibles.

CHAPITRE 3

RÉSULTATS

3.1 RÉDUCTIONS DE DIMENSION

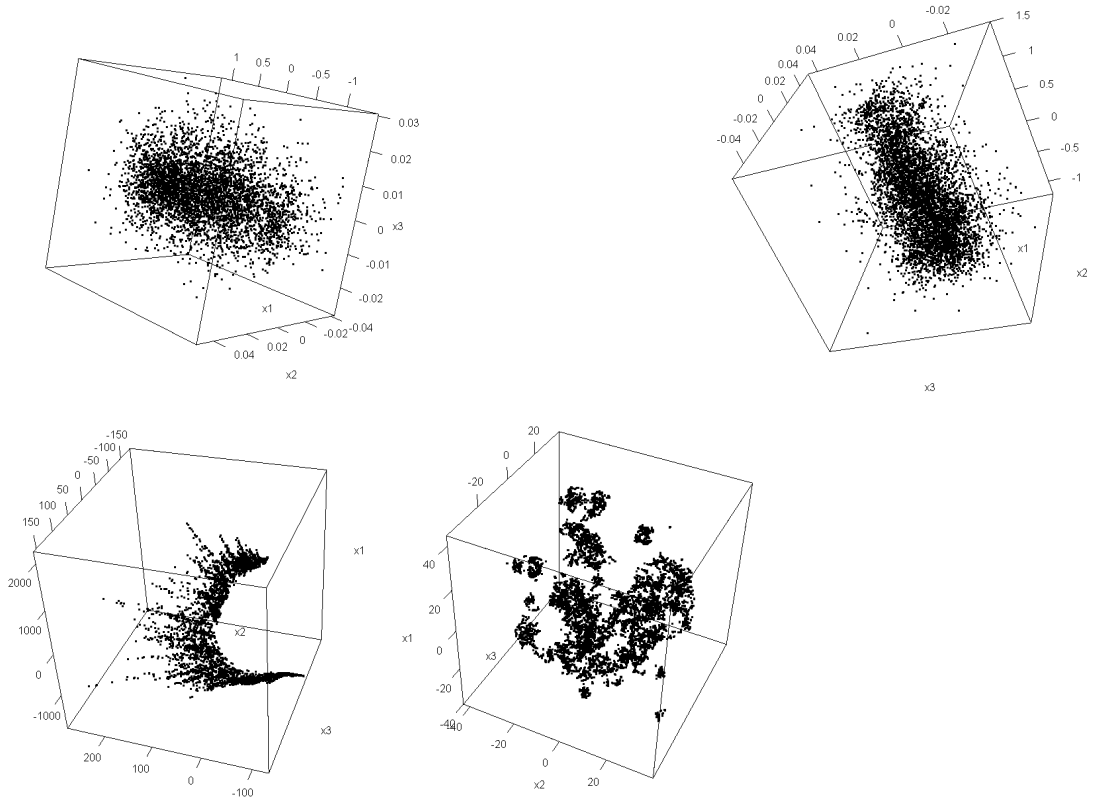

FIGURE 3.1 – *Représentation des jeux de données obtenus après réduction de dimension. En haut à gauche : MDS métrique, en haut à droite : MDS non-métrique, en bas à gauche : Isomap et en bas à droite : tSNE.*

Pour obtenir tous mes modèles, j'ai construit l'ensemble des jeux de données issus de réduction de dimension. Je pars donc du jeu de données brutes, représenté par la matrice $X \in M_{np}$ avec $n \approxeq 10\,000$ grossesses uniques et $p = 27$ variables explicatives, pour essayer les différentes réductions de dimension.

Dans le cadre des méthodes FAMD et kPCA, les critères de réduction de Kaiser-Guttman et Karlis-Saporta-Spinaki ne sont pas respectés malgré une réduction de dimension de $\mathbb{R}^{27} \to \mathbb{R}^{27}$ donc je n'ai pas gardé les jeux de données issus de ces deux méthodes.

Pour la méthode MDS, le critère de réduction sur le *STRESS* est satisfait pour toutes les dimensions même de $\mathbb{R}^{27} \to \mathbb{R}$. J'ai donc décidé de faire une réduction MDS métrique et non-métrique de $\mathbb{R}^{27} \to \mathbb{R}^3$ pour pouvoir observer le jeu de données en 3 dimensions. J'ai effectué une réduction de dimension de $\mathbb{R}^{27} \to \mathbb{R}^3$ pour les méthodes Isomap et tSNE car ces méthodes n'ont pas de critère pour trouver la dimension optimale d'arrivée et que la dimension 3 est visible géométriquement.

Pour la méthode LLE, l'algorithme donne une dimension optimale $d = 11$. Il n'est donc pas possible d'observer géométriquement la nouvelle structure des données comme avec les autres jeux de données.

La structure géométrique des jeux réduits en 3 dimensions est présente dans la figure 3.1. On peut observer que les méthodes MDS métrique et non-métrique donnent une structure assez similaire : il y a une structure globale de cylindre avec une plus grande agglomération des points au niveau de l'axe principal du cylindre. Le jeu obtenu par réduction Isomap a une structure très spécifique : l'ensemble des points s'aligne le long d'une surface courbée bien marquée. La méthode tSNE permet d'effectuer un clustering des grossesses puisqu'on observe la formation de sous-groupes de grossesses isolés géométriquement ce qui permet, si la formation des sous-groupes est pertinente, de faciliter les méthodes de Machine Learning qui ont un raisonnement géométrique puisque les sous-groupes de grossesses sont déjà séparés géométriquement.

3.2 RÉGRESSION POUR LE POIDS FŒTAL

L'enjeu de la prédiction du poids fœtal est à la fois de mettre en place les méthodes de Machine Learning sur les différents jeux de données, mais aussi d'étudier la fiabilité du modèle actuel qui est utilisé actuellement dans la quasi totalité des hôpitaux en France : la formule de Hadlock. [**5**]

Hadlock a développé en 1984, une formule qui dépend de trois paramètres : Périmètre Cranien (PC), Périmète Abdominal (PA) et Longueur Fémorale (LF). Il s'agit d'une régression polynomiale de degré 2 avec les monômes : PC, PA, LF et PA*LF.

Pour obtenir sa formule, Hadlock a construit son modèle de régression avec les hypothèses de ne prendre que des résultats échographiques provenant de 533 grossesses uniques dont les mesures ont été réalisées au maximum une semaine avant l'accouchement.

J'ai donc créé un jeu de données (Hadlock) spécial dont les grossesses répondent aux mêmes critères d'inclusion que ceux donnés dans l'étude d'Hadlock. Ce jeu est donc simplement dérivé du jeu de données brutes.

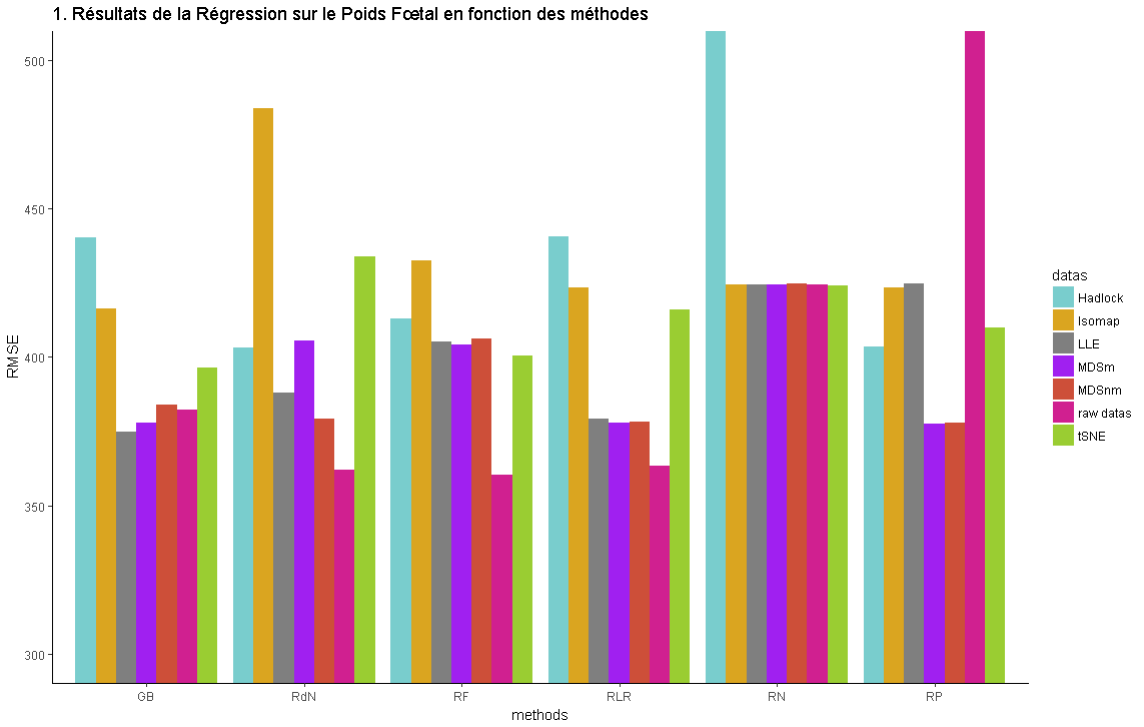

FIGURE 3.2 – *Résultats de la régression sur le poids fœtal en fonction des méthodes. GB : gradient boosting, RdN : réseau de neurones artificiels, RF : forêts aléatoires, RLR : régression linéaire régularisée, RN : régression naïve, RP : régression polynomiale, MDSm : MDS métrique, MDSnm : MDS non-métrique, raw datas : données brutes*

J'ai indiqué sur les figures 3.2 et 3.3 les résultats synthétiques des méthodes de Machine Learning en montrant la valeur moyenne des RMSE obtenue pour chaque méthode sur chaque jeu de données. Tous mes résultats sont illustrés par des histogrammes car cela me permet de représenter la valeur moyenne du critère d'optimisation (RMSE) avec une distinction visuelle des différentes méthodes et jeux de données utilisés.

On peut observer que c'est généralement sur le jeu de données brutes qu'il y a les meilleurs résultats. Pour ce problème il n'y avait pas grand intérêt à faire de la réduction de dimension pour améliorer les performances des modèles. Il y a deux résultats nettement plus élevés que les autres : régression polynomiale sur données brutes et régression naïve sur le jeu Hadlock. Ceci s'explique respectivement par un phénomène de sur-apprentissage qui empêche une bonne généralisation du modèle et par la présence d'un biais dans la sélection des grossesses qui augmente la variance du jeu de données.

On observe aussi que les méthodes de réduction LLE, MDS métrique et MDS non-métrique donnent des résultats similaires.

Le jeu de données Hadlock ne donne jamais le meilleur résultat peu importe la méthode, et dans la moitié des méthodes il s'agit du jeu de données donnant le plus mauvais résultat. Donc

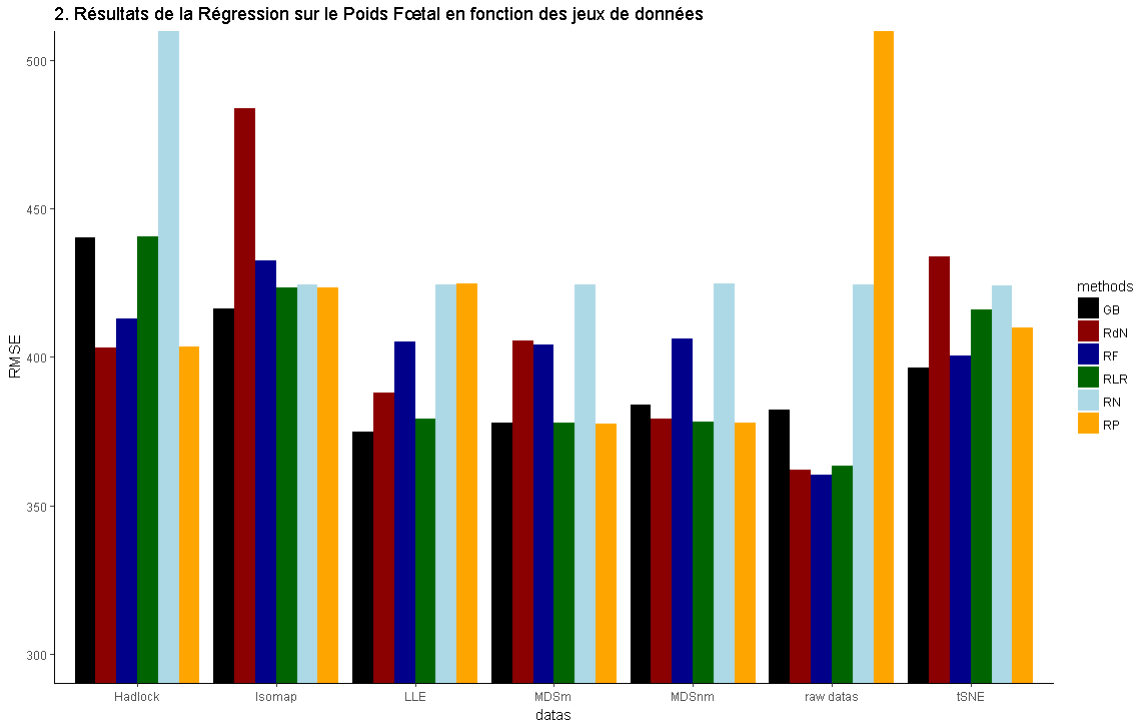

FIGURE 3.3 – *Résultats de la régression sur le poids fœtal en fonction des jeux de données. GB : gradient boosting, RdN : réseau de neurones artificiels, RF : forêts aléatoires, RLR : régression linéaire régularisée, RN : régression naïve, RP : régression polynomiale, MDSm : MDS métrique, MDSnm : MDS non-métrique, raw datas : données brutes*

pour le reste des problèmes je n'ai pas réutilisé ce jeu puisqu'il ne donnait jamais un meilleur résultat que les autres jeux de données. Ces mauvais résultats peuvent s'expliquer du fait que la méthode développée par Hadlock n'est pas bien généralisable, c'est-à-dire que comme son étude a nécessité plusieurs critères de sélection (notamment sélectionner des grossesses dont les mesures échographiques sont connues une semaine avant l'accouchement), il y a la présence d'un biais dans le jeu dont il s'est servi. Cela fait que les grossesses qui ne répondent pas à ses critères de sélection ne sont pas adaptées au modèle et donc n'obtiennent pas de bons résultats.

Le meilleur résultat correspond à une forêt aléatoire sur le jeu de données brutes avec un score RMSE de 360g.

3.3 CLASSIFICATION POUR LA MACROSOMIE

Les résultats sur la détection de la macrosomie sont les plus importants cliniquement puisque la prédiction d'une macrosomie permet d'anticiper une prise en charge et la mobilisation d'un plateau technique spécifique plus adapté, ce qui est source de réduction de morbidité induite par une macrosomie susceptible de nécessiter des manœuvres d'extraction ou de césarienne décidée en

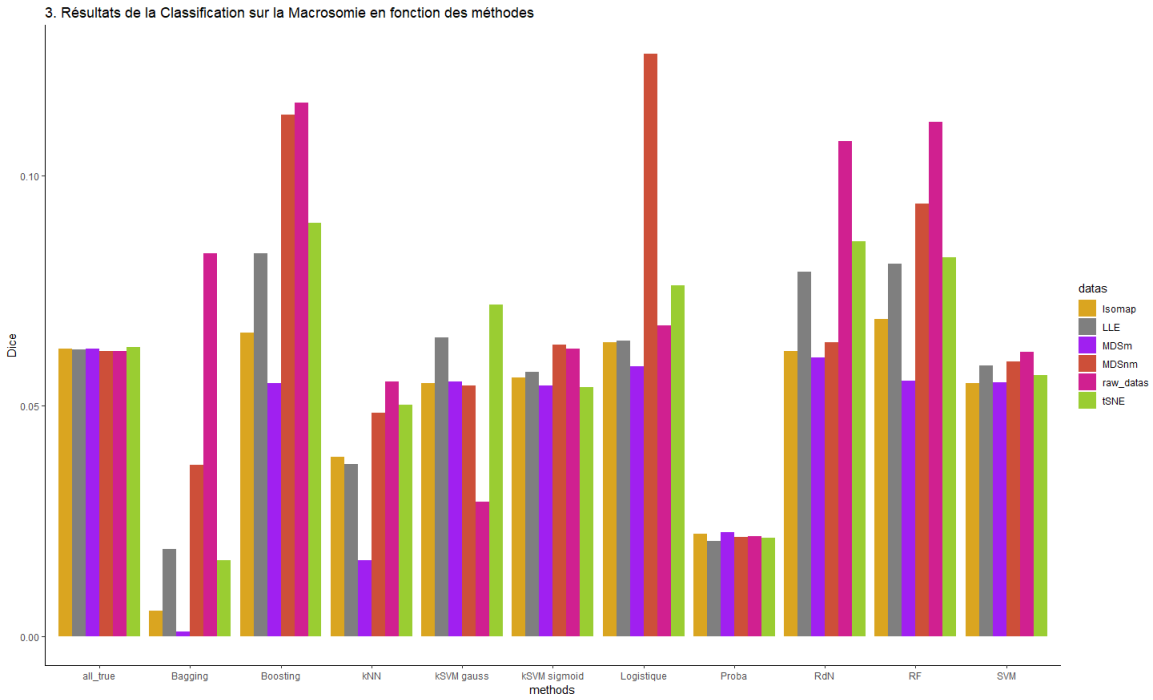

FIGURE 3.4 – *Résultats de la Classification sur la macrosomie en fonction des méthodes. all_true : classification naïve 1, kNN : plus proches voisins, kSVMgauss : SVM à noyau gaussien, kSVMsigmoid : SVM à noyau sigmoïdal, Logistique : régression logistique, proba : classification naïve 2, RdN : réseau de neurones artificiels, RF : forêts aléatoires, MDSm : MDS métrique, MDSnm : MDS non-métrique, raw_datas : données brutes*

urgence de façon non-programmée.

J'ai deux stratégies pour établir une macrosomie : soit on effectue une classification car la prédiction de la macrosomie revient à prédire une variable binaire, soit on utilise les résultats de la régression du poids fœtal puis on regarde si le poids prédit correspond à une macrosomie ou non.

Sur les figures 3.4 et 3.5, il y a la synthèse des résultats obtenus par les méthodes de classification avec la moyenne des Dice obtenue pour chaque méthode de classification sur chaque jeu de données.

Les méthodes {all_true} et {proba} correspondent à des méthodes de classification qui fonctionnent au hasard et qui n'apprennent pas des variables explicatives, elles servent à comparer les classifications à une classification naïve. Il est important de voir que la plupart des méthodes peuvent obtenir des meilleurs résultats que ces deux méthodes.

On remarque que les jeux de données brutes et MDS non métrique donnent des résultats en moyenne meilleurs que les autres jeu de données. Le meilleur Dice est obtenu en combinant une régression logistique au jeu MDS non-métrique avec un Dice de 12,6%, ce qui est deux fois plus performant que la méthode {all_true} et deux fois plus performant que les méthodes actuelles. [**1**] [**2**]

En utilisant les résultats de la régression pour le poids fœtal on obtient les résultats décrits dans

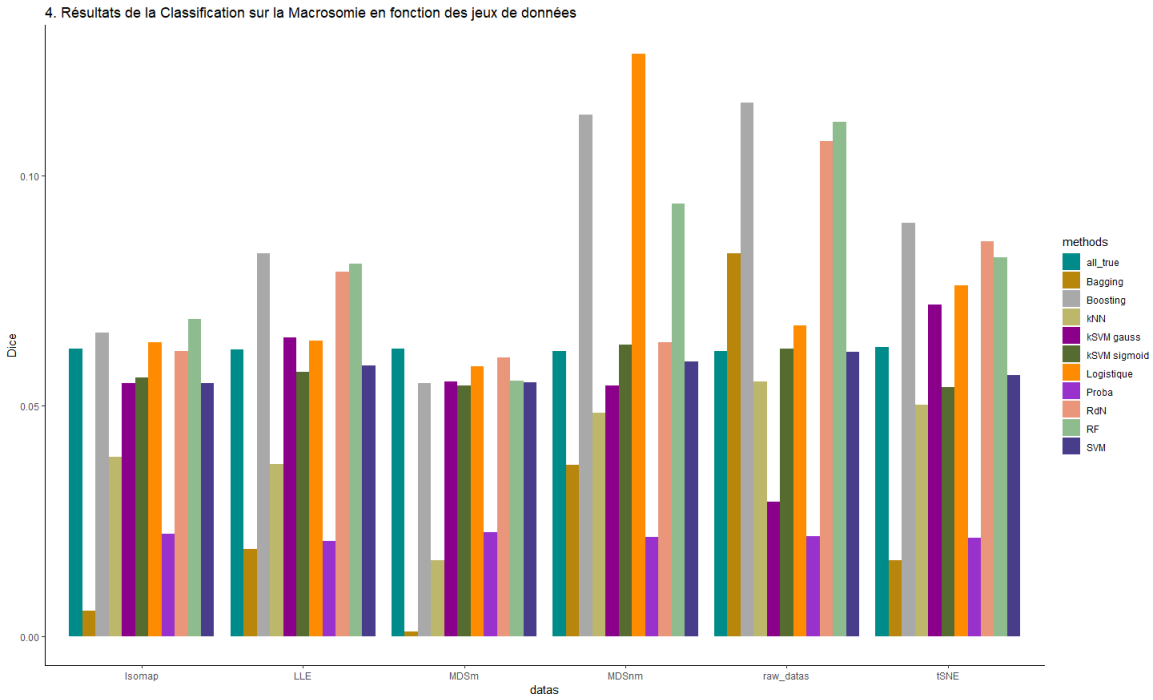

FIGURE 3.5 – *Résultats de la classification sur la macrosomie en fonction des jeux de données. all_true : classification naïve 1, kNN : plus proches voisins, kSVMgauss : SVM à noyau gaussien, kSVMsigmoid : SVM à noyau sigmoïdal, Logistique : régression logistique, proba : classification naïve 2, RdN : réseau de neurones artificiels, RF : forêts aléatoires, MDSm : MDS métrique, MDSnm : MDS non-métrique, raw_datas : données brutes*

le tableau 3.1. On a des résultats différents selon la définition de la macrosomie : soit une définition par rapport à une valeur de référence : poids>4000g ; soit une définition par rapport à la population auquel cas on est macrosome si on a un poids supérieur au 95^e percentile des données obtenues par la régression (respectivement $def1$ et $def2$).

Méthodes	GB	RdN	RF	RLR	RN	RP
$RMSE_{def1}$	1.8%	1.8%	0%	1.3%	0%	0%
$RMSE_{def2}$	14.8%	16.5%	14.4%	15.9%	0%	14.3%

TABLE 3.1 – *Dice obtenus par les méthodes de régression. GB : gradient boosting, RdN : réseau de neurones artificiels, RF : forêts aléatoires, RLR : régression linéaire régularisée, RN : régression Naïve, RP : régression polynomiale. macrosomie def1 : poids > 4000g, def2 : poids > 95^e percentile*

On observe que la première définition de la macrosomie ne donne pas du tout de bons résultats alors que revenir à l'approche de population via la définition 2 est plus efficace que les méthodes de classification elles-mêmes. Je peux donc conclure que les méthodes de régression ne permettent pas de bien attribuer la valeur du poids fœtal pour un macrosome, cependant la régression conserve bien

les poids élevés, c'est pour cela que l'on obtient de bons scores de classification avec la définition 2. Les résultats de la régression naïve est toujours 0% puisque cette méthode prédit toujours la même valeur pour toute grossesse, elle ne prédit donc jamais de macrosomie.

3.4 CLASSIFICATION POUR LA RÉANIMATION

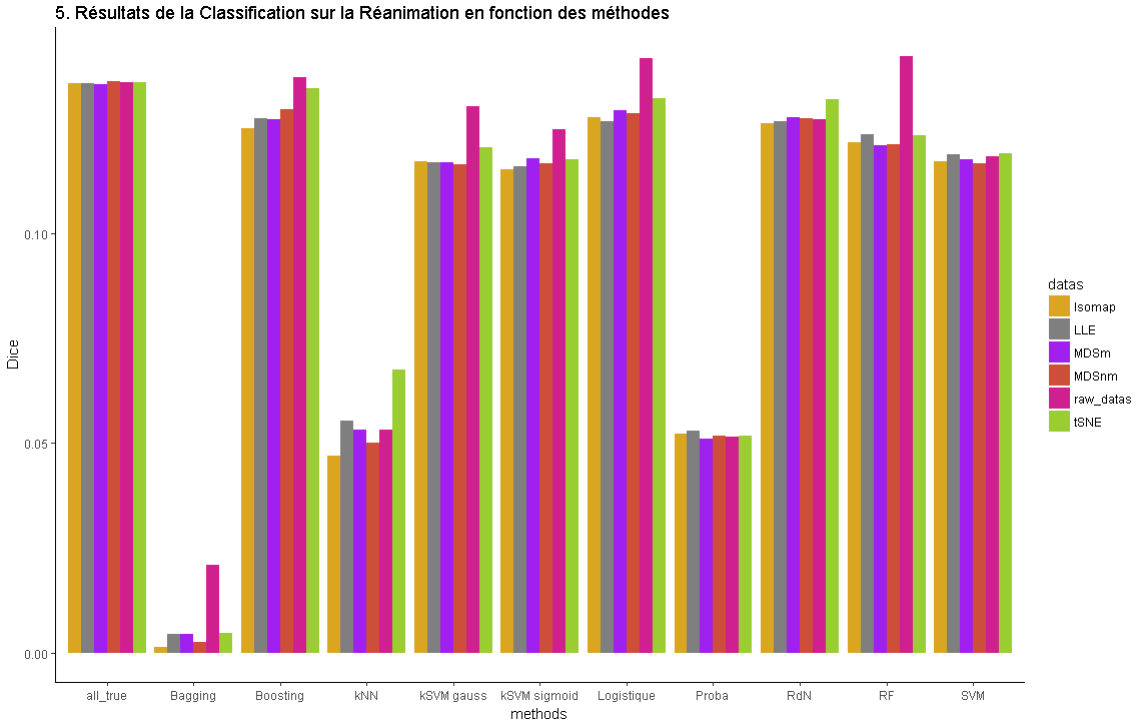

FIGURE 3.6 – *Résultats de la classification sur la réanimation en fonction des jeux de données. all_true : classification naïve 1, kNN : plus proches voisins, kSVMgauss : SVM à noyau gaussien, kSVMsigmoid : SVM à noyau sigmoïdal, Logistique : régression logistique, proba : classification naïve 2, RdN : réseau de neurones artificiels, RF : forêts aléatoires, MDSm : MDS métrique, MDSnm : MDS non-métrique, raw_datas : données brutes*

Les résultats obtenus pour la réanimation sont synthétisés dans les figures 3.6 et 3.7 pour chaque méthode de classification sur chaque jeu de données.

La méthode {all_true}, qui correspond à une méthode naïve de classification, se retrouve à être une des méthodes les plus efficaces alors qu'elle ne prend pas en compte les variables explicatives. Cependant dans certaines méthodes le jeu de données brutes conduit à des résultats légèrement plus élevés que la méthode {all_true}.

Hormis le jeu de données brutes qui obtient des résultats légèrement plus élevés que les autres jeux de données, aucun jeu de données issus de réduction de dimension n'obtient de résultats nettement différents, donc dans ce problème de classification la réduction de dimension n'améliore

pas les résultats.

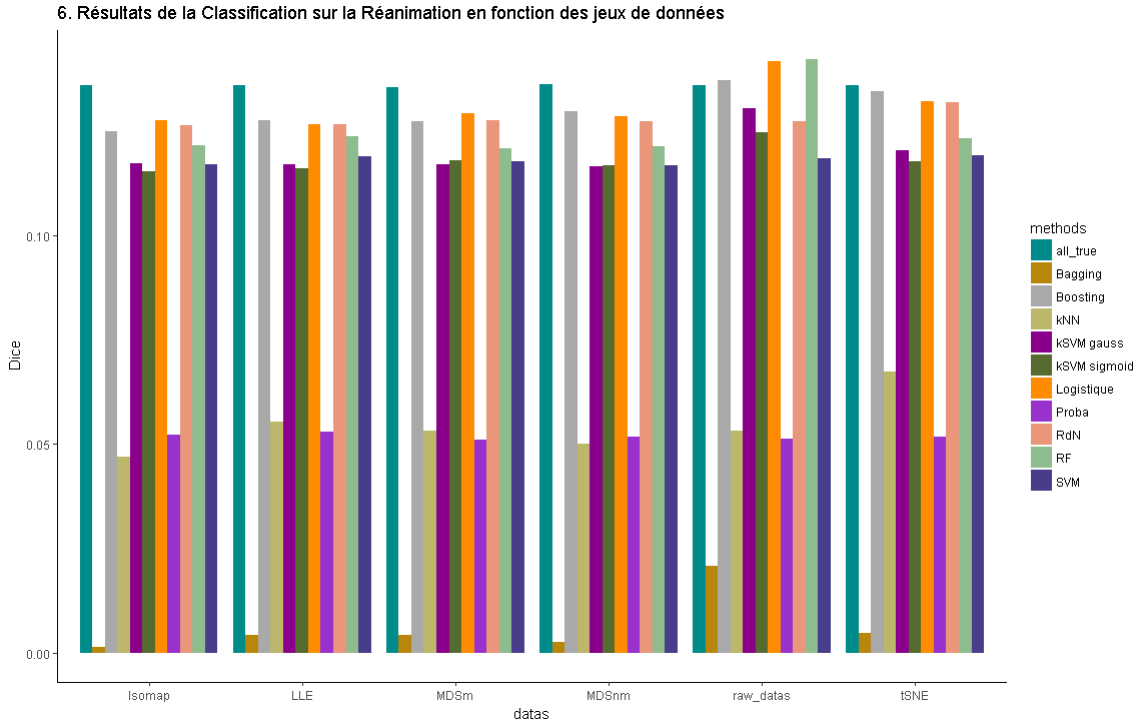

FIGURE 3.7 – *Résultats de la classification sur la réanimation en fonction des méthodes. all_true : classification naïve 1, kNN : plus proches voisins, kSVMgauss : SVM à noyau gaussien, kSVMsigmoid : SVM à noyau sigmoïdal, Logistique : régression logistique, proba : classification naïve 2, RdN : réseau de neurones artificiels, RF : forêts aléatoires, MDSm : MDS métrique, MDSnm : MDS non-métrique, raw_datas : données brutes*

Le Dice le plus élevé (14.2%) a été obtenu sur une Forêt Aléatoire avec le jeu de données brutes.

Les méthodes qui sont nettement moins efficaces que les autres sur le problème de la réanimation sont le Bagging, la méthode des Plus Proches Voisins et la Classification Naïve 2.

Le reste des méthodes restent globalement proches au niveau des résultats obtenus.

Une manière d'améliorer l'étude sur la réanimation serait de considérer la différence entre les réanimations complexes et les simples assistances respiratoires beaucoup plus courantes. L'objectif serait alors de plus valoriser la prédiction des réanimations complexes car ce sont celles qui nécéssitent le plus d'attention.

3.5 CLASSIFICATION POUR LE PRONOSTIC VITAL

Les derniers résultats obtenus pour la prédiction du pronostic vital d'un enfant sont synthétisés dans le tableau 3.2.

Méthodes	all_true	Bagging	Boosting	Log	Proba	SVM	kSVMgauss	kSVMsig
χ	75.6%	12.2%	15.7%	18.2%	0.2%	0%	1.1%	12.7%

TABLE 3.2 – *χ obtenus par les méthodes de classification. all_true : classification naïve 1, Log : régression logistique, Proba : classification naïve 2, SVM : machine à vecteurs de support, kSVMgauss : SVM à noyau gaussien, kSVMsig : SVM à noyau sigmoïdien*

Il faut savoir que le calcul du coefficient χ est légèrement différent de celui de la définition ($\chi = \frac{tr(M)-M_{00}}{sum(M)-M_{00}}$) car j'ai enlevé l'influence du nombre de grossesses correctement prédites dans la catégorie $\{0\}$ car il y a en réalité plus de 99% des grossesses qui y appartiennent. De plus il est plus intéressant de prédire correctement les autres catégories afin de pouvoir anticiper et éviter les morts-nés.

La méthode {all_true} se trouve très avantagée par la définition de ce coefficient ($\frac{TP}{FP} = 2,96$) c'est pour cela que cette méthode a un coefficient χ aussi élevé.

Autrement les méthodes de bagging, boosting, SVM à noyau sigmoïdal et de la régression logistique donnent de bons résultats pour cette classification multimodale ($m > 2$) par rapport aux autres méthodes. Par ailleurs, ces méthodes obtiennent des résultats plus élevés que les résultats obtenus sur les problèmes de classification bimodale.

3.6 CONCLUSIONS SUR LES TRAVAUX DE HADLOCK

Actuellement dans la majorité des hôpitaux, les professionnels utilisent les formules de Hadlock pour estimer le poids foetal in utero, et ainsi situer la croissance staturo-pondérale d'un foetus.

Pour montrer la différence entre une des méthodes de classification et celle qui est retenue avec Hadlock, j'ai construit un coefficient *alpha* qui intervient dans la construction des modèles de classification : informatiquement le modèle attribue à chaque grossesse une probabilité p d'être classée comme {macrosome}. Le modèle considère que si $p > seuil$ alors la grossesse doit être étiquetée avec la catégorie {macrosome}. La valeur *seuil* étant préalablement optimisée par ROC-analyse.

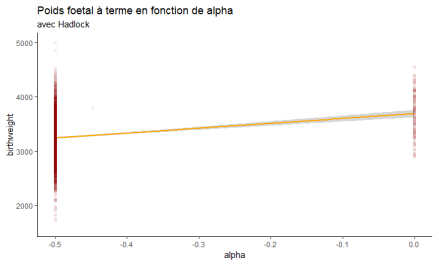

FIGURE 3.8 – *Poids fœtal en fonction de alpha : avec Hadlock*

L'expression de alpha est croissante avec la probabilité p, le modèle est correct si plus alpha est grand alors plus le poids fœtal est grand :

$$alpha = \frac{2}{\pi} arc\tan(\frac{p}{seuil} - 1)$$

Comme p et $seuil \in [0;1]$, on a $alpha \in [-0.5;1]$,de plus si $alpha < 0$ le nouveau-né est prédit {non-macrosome} sinon si $alpha \geq 0$ le nouveau-né est prédit {macrosome}.

Sur les figures 3.8 et 3.9 j'ai représenté le poids fœtal en fonction du coefficient alpha respectivement avec la méthode de classification d'Hadlock et sans la méthode d'Hadlock (régression logistique avec le jeu MDS non-métrique).

Le critère représenté par $alpha$ est alors considéré comme pertinent si le poids fœtal est proportionnel avec $alpha$. Cela revient à dire qu'il faut que la fonction u qui approxime le modèle **poids fœtal = u(alpha)** soit croissante. En effet si u est croissante, on sait alors qu'une valeur de $alpha \simeq 1$ est associé à un risque élevé de macrosomie et qu'une valeur de $alpha \simeq -0.5$ est associé à un risque faible de macrosomie.

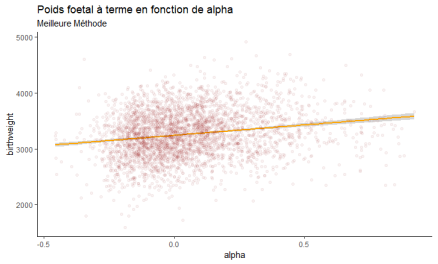

FIGURE 3.9 – *Poids fœtal en fonction de alpha : sans Hadlock*

On observe que la méthode d'Hadlock obtient des valeurs de alpha très concentrées autour des bornes de l'intervalle $[-0.5; 1]$ alors que ma méthode disperse bien les valeurs de alpha sur l'ensemble de cet intervalle. Avec Hadlock on ne peut donc pas vraiment tracer de courbe de tendance au centre de l'intervalle car il y a très peu de grossesses qui ont un coefficient alpha dans ces eaux-là. En revanche on peut aisément tracer une courbe de tendance avec mon modèle et on peut observer qu'une courbe de tendance linéaire donne une droite de pente positive même si la méthode n'est pas parfaite.

Il faut donc retenir que les travaux de Hadlock sont améliorables et que la nécessité d'améliorer ces travaux est importante puisque l'impact est directement lié aux morbidités des grossesses.

Chapitre 4

Conclusions du Stage

4.1 Conclusions sur les méthodes de Machine Learning

Globalement pour toutes les questions auxquelles j'ai répondu, les méthodes de Machine Learning n'ont pas donné des résultats vraiment satisfaisants. Il faut savoir cependant que je me suis intéressé à des questions d'intérêts cliniques et que j'y ai répondu avec toutes les armes que j'avais à ma disposition.

De plus même si les résultats sont loin de la perfection, ils sont quand même meilleurs que les résultats des différentes formules de prédiction qui sont utilisés dans les hôpitaux actuellement.

Le plus alarmant étant de voir que les techniques de détection des morbidités actuellement utilisées ne sont pas meilleures que le hasard, alors que sur mes résultats, notamment vis-à-vis de la macrosomie, les résultats sont deux fois plus corrects que ce hasard.

Pour améliorer ces résultats il y a plusieurs pistes de recherche :

1. Ajouter une autre partie de méthodes non-supervisées en pré-processus à l'aide de méthodes de clustering. Le clustering permettrait d'effectuer des sous-groupes de grossesses sur lesquels il est possible d'effectuer un modèle de régression ou classification. Le problème étant de savoir comment choisir les critères d'adhésion à un sous-groupe. De plus il faut alors relancer toutes les méthodes sur tous les jeux de données pour chaque sous-groupe ce qui augmente encore considérablement le nombre de modèles et de résultats.
2. Lors du traitement des bases de données initiales, je me suis aperçu que beaucoup de variables ne sont pas complétées pour une grande majorité de grossesses. Etant donné que j'effectue de la réduction de dimension, il aurait pu être intéressant d'avoir accès à un nombre plus grand de variables explicatives (autour du volume de liquide amniotique ou sur les indices de résistance/pulsivité des artères,...).
3. Avoir une approche longitudinale du problème car les mesures qui sont relevées le sont à des dates précises et selon les grossesses, on peut avoir plus de mesures que pour d'autres grossesses. Le problème est alors d'être capable de prendre en compte le temps et le nombre de mesures qui sont spécifiques à chaque grossesse.

4.2 CONCLUSIONS PERSONNELLES DU STAGE

Le stage s'est très bien passé, ce fût un plaisir d'avoir pu l'effectuer avec l'ensemble de l'équipe avec qui je suis intervenu. L'originalité de ce stage pour moi était le fait de pouvoir réellement apporter des connaissances à l'équipe et ne pas juste avoir l'équipe qui m'apporte des connaissances.

Le stage m'aura personnellement introduit aux méthodes de Machine Learning, au métier de data scientist et au domaine de la recherche médicale. Il faut savoir que je n'ai pas eu de cours lors de ma formation sur les méthodes de Machine Learning et que mon cursus ne cherche pas spécialement à former au métier de data scientist.

Le stage m'a permis de me perfectionner dans l'utilisation du logiciel R et notamment avec certains aspects de programmation que je n'avais pas eu l'occasion d'apprendre avec mon cursus, cela comprend la possibilité de réaliser de la programmation parallèle, l'utilisation des fichiers Rmarkdown pour stocker et simplifier l'exécution des programmes, l'utilisation de l'outil Shiny qui permet de développer une interface graphique pour créer des programmes interactifs et facilement modifiable sans modifier le programme source. J'ai eu aussi l'occasion d'apprendre à utiliser Latex pour pouvoir rédiger des rapports ou faire des diaporamas.

J'ai beaucoup apprécié ce stage car il avait un intérêt réel puisque les questions cliniques auxquelles j'ai répondu sont des questions que se posent toujours actuellement les professionnels. Du fait que j'ai une formation d'ingénieur contrairement à tous les autres membres de l'équipe, je n'avais pas la même façon d'aborder les questions et d'y répondre, ce qui a permis une grande pluridisciplinarité des compétences et l'utilisation de nouvelles méthodes qui ont porté leurs fruits dans la résolution des problèmes. Ce stage demandait aussi à juste titre une grande capacité d'autonomie parce que j'ai dû apprendre tout seul à maitriser les méthodes de Machine Learning et personne dans l'équipe n'avait la même formation vis-à-vis de la programmation sur R.

Je suis ravi de savoir que mon travail permet une avancée dans le domaine de la périnatalité et sur le traitement des grossesses. Il est même question de publier mes résultats dans un article scientifique dans une revue de biostatistiques / bioinformatiques.

Pour donner des aspects négatifs, il faut savoir que la programmation sur R peut être longue et fastidieuse (un programme peut mettre 10 jours pour s'exécuter) et que le traitement devient assez redondant une fois que l'on a traité un problème de régression et un problème de classification puisqu'il suffit juste de changer la variable à expliquer pour effectuer le même traitement statistique. Mais malgré ces aspects négatifs, ça a été un grand honneur de réaliser ce stage.

ANNEXE A

MÉTHODES DE RÉGRESSION

Annotations

- **LS** = Learning Set = {données du jeu d'apprentissage}
- **TS** = Testing Set = {données du jeu test}
- $\mathbf{X} \in \mathscr{M}_{np}$ = **LS** $\bigcup$ **TS** = ensemble des données
- $\mathbf{x} \in \mathscr{M}_{1p} \subset \mathbf{X}$ = une grossesse
- $\mathbf{Y} \in \mathscr{M}_{n1}$ = une variable quantitative d'intérêt pour chaque grossesse
- $\mathbf{Y} \in \mathbb{R}^{+*}$ = une variable quantitative d'intérêt pour une grossesse
- $\mathbf{n_{LS}}$ = nrow(**LS**) = nombre de grossesses dans le jeu d'apprentissage
- $\mathbf{n_{TS}}$ = nrow(**TS**) = nombre de grossesses dans le jeu test
- $\mathbf{n} = \mathbf{n_{LS}} + \mathbf{n_{TS}}$ = nombre total de grossesses

Le but d'une méthode de régression est de pouvoir trouver une fonction $f : \mathbb{R}^p \to \mathbb{R}$ pour prédire une variable quantitative **y** à partir d'un jeu de données **x**, de **p** dimensions. Pour évaluer une fonction **f**, on calcule la **RMSE** (Root Mean Squared Error) :

$$RMSE = \sqrt{\frac{1}{n_{TS}} \sum_{i \in \mathbf{TS}} (y_i - f(x_i))^2}$$

Ce coefficient s'interprète comme étant la valeur moyenne des erreurs qui sont commises lors de la prédiction du jeu test.

RÉGRESSION NAÏVE

Le but de cette méthode est de trouver la meilleure fonction constante qui approxime le jeu d'apprentissage. Dans le cadre d'une variable quantitative, cette constante correspond à la moyenne

de la variable quantitative en question pour le jeu de données d'apprentissage.

$$f_{RN}(x) = mean((y_i)_{i \in \mathbf{LS}}) = \frac{1}{n_{LS}} \sum_{i \in \mathbf{LS}} (y_i)$$

RÉGRESSION LINÉAIRE RÉGULARISÉE

La régression linéaire régularisée est une régression linéaire : $y(x) = {}^T\beta . \begin{bmatrix} x \\ 1 \end{bmatrix}$, avec $\beta \in \mathbb{R}^{p+1}$. On rajoute des contraintes sur le choix de β pour éviter les phénomènes de sur-apprentissage, prendre en compte les corrélations entre les variables et enlever les variables peu explicatives. Pour prendre en compte les corrélations entre les variables, il faut rajouter à la régression linéaire un terme $\lambda_{RR}\|\beta\|_2^2$ correspondant à une régression ridge.

$\lambda_{RR} \in \mathbb{R}$ et $\|\beta\|_2^2 = \sum_i \beta_i^2$.

La régression ridge nous permet de réduire l'amplitude des coefficients d'une régression linéaire et d'éviter le sur-apprentissage. Cependant, on peut souhaiter pousser les choses plus loin, et annuler certains coefficients. Les variables qui auront un coefficient égal à zéro ne feront plus partie du modèle, qui en sera simplifié d'autant.

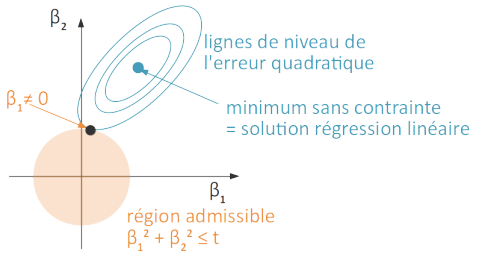

FIGURE *A.1* – *Solution régression ridge*

Pour cela il faut rajouter à la regression linéaire le terme $\lambda_{lasso}\|\beta\|_1$ correspondant à une régression lasso.

$\lambda_{lasso} \in \mathbb{R}$ et $\|\beta\|_1 = \sum_i |\beta_i|$

Pour combiner la régression ridge et la méthode lasso, on réalise donc la régression linéaire régularisée (méthode elastic net), ce qui vient à contruire la fonction

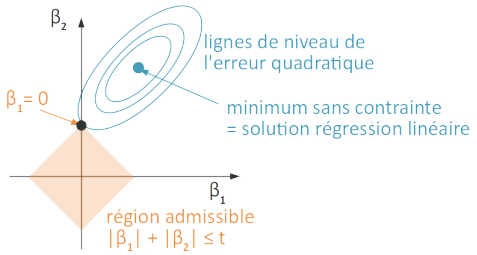

FIGURE *A.2* – *Solution Lasso*

$$f_{RLR}(x) = {}^T\beta . \begin{bmatrix} x \\ 1 \end{bmatrix} + \lambda . (\alpha\|\beta\|_1 + (1-\alpha)\|\beta\|_2^2)$$

avec les hyperparamètres $\lambda \in \mathbb{R}$ et $\alpha \in [0;1]$.

On obtient le vecteur β en résolvant :

$$\underset{\beta \in \mathbb{R}^{p+1}}{arg\,\min} \|Y - X.\beta\|_2^2 + \lambda . (\alpha\|\beta\|_1 + (1-\alpha)\|\beta\|_2^2)$$

qui admet une solution déterministe.

RÉGRESSION POLYNOMIALE

La régression polynomiale est une combinaison linéaire de monômes à une indéterminée et de monômes à deux indéterminées : $y(x) = (Ax|x) + bx$ avec $A \in \mathscr{M}_p(\mathbb{R}), b \in \mathbb{R}^p$ et $. \mapsto (.|.)$ le produit scalaire canonique de $\mathscr{M}_p(\mathbb{R})$.

Ce modèle peut très facilement créer de l'overfitting car le modèle contient beaucoup de variables ($p^2 + p$). Pour éviter cela il faut imposer beaucoup de zéros aux coefficients de A et b et ne sélectionner que les coefficients devant les monômes les plus explicatifs vis-à-vis de la variable d'intérêt.

La library GAMLSS sur le logiciel R permet de construire ce type de modèle en se basant sur le critère du GAIC (Generalised Akaike Information Criterion) pour sélectionner les monômes. Le GAIC d'un modèle s'exprime par la formule :

$$GAIC(modele) = 2k - \ln(2L)$$

avec k : le nombre de variables dans le modèle et L le maximum de la fonction de vraisemblance du modèle.

Ce critère permet de prendre en compte l'efficacité d'un modèle (augmentation de L) et de garder un nombre de variables raisonnable (k).

Cela permet d'obtenir une régression polynomiale de la forme :

$$f_{RP}(x) = \sum_{k \in \mathscr{P}([\![1;p]\!])} \beta_k x_k + \sum_{i,j \in \mathscr{P}([\![1;p]\!])} \alpha_{ij} x_i . x_j$$

avec $\forall i, j \in [\![1;p]\!], \alpha_{ij}, \beta_i \in \mathbb{R}$ et x_i la i^{eme} variable de x.

RÉSEAU DE NEURONES ARTIFICIELS

Le modèle du réseau de neurones artificiels s'inspire de la structure d'un neurone : il y a une couche de neurones d'entrée (qui n'ont pas de prédécesseurs), une couche cachée à N neurones et la couche de sortie qui a autant de neurone que de réponse attendue donc pour prédire **y** uniquement un neurone.

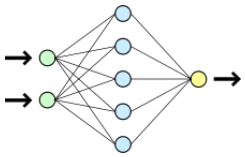

FIGURE A.3 – *Réseau de Neurones Artificiels*

Pour le i^{eme} neurone de la couche $n+1$ possédant k_n prédécesseurs prend donc une valeur particulière en fonction de la valeur des k_n neurones précédents :

$$\begin{aligned} U_{n+1}[i] &= a(U_n) \\ &= {}^T\Omega i_n . U_n \\ &= a(u_n[1], \ldots, u_n[k_n]) \\ &= \sum_{j=1}^{k_n} {}^T\omega i_{nj} u_n[j] \end{aligned}$$

avec $U_n \in \mathscr{M}_{k_n 1}(\mathbb{R})$: la valeur des neurones de la n^{eme} couche , $\Omega i_n \in \mathscr{M}_{k_n 1}(\mathbb{R})$.

En posant $\Omega_n = (\Omega 1_n, \Omega 2_n, ..., \Omega k_n{}_n), \Omega_n \in \mathscr{M}_{k_n k_{n+1}}(\mathbb{R})$, on a $\forall n$:

$$\begin{aligned} U_{n+1} &= {}^T\Omega_n . U_n \\ &= {}^T\Omega_n . {}^T\Omega_{n-1} ..... {}^T\Omega_1 . U_1 \end{aligned}$$

On a donc un modèle de réseau de neurones artificiels via la fonction :

$$f_{RdN}(x) = {}^T\Omega_{m-1} . {}^T\Omega_{m-2} ..... {}^T\Omega_1 . x$$

avec m le nombre de couches de neurones, $\forall i \in [\![1; m-1]\!] \Omega_i \in \mathscr{M}_{k_i k_{i+1}}(\mathbb{R})$ et k_i le nombre de neurones dans la i^{eme} couche.

Les matrices $(\Omega_i)_{i \in [\![1;m-1]\!]}$ sont calculées itérativement avec l'algorithme du gradient en minimisant l'erreur quadratique moyenne.

FORÊTS ALÉATOIRES

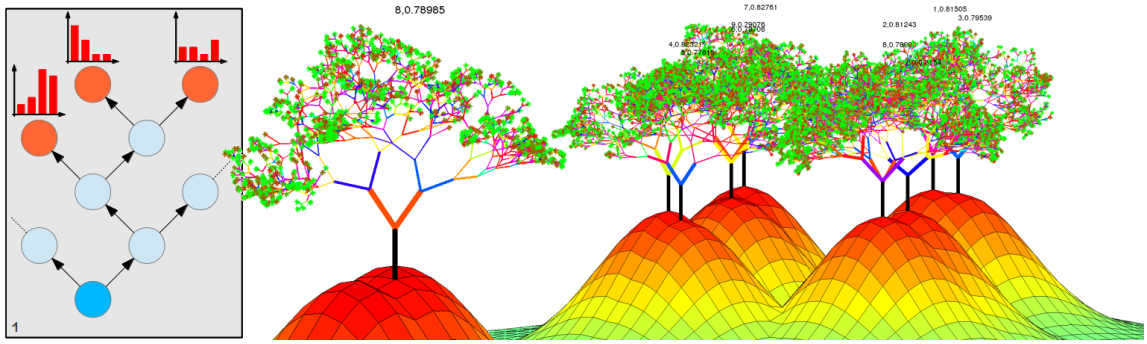

FIGURE *A.4* – *Forêt Aléatoire*

Une forêt aléatoire est un modèle additif qui se construit comme étant une somme pondérée d'arbres de décisions binaires.

Un arbre de décisions binaires correspond à une structure qui permet à partir d'une racine de scinder un jeu de données amont en deux sous-jeux de données à chaque bifurcation dans l'arbre (branche). Un chemin suivi le long des branches se termine à une feuille où il n'y ainsi plus de bifurcation. A une bifurcation, le jeu de données est séparé à partir d'une variable ($\mathbf{V}$) et une valeur associée ($\mathbf{V_m}$) de sorte à séparer les grossesses qui ont $\mathbf{V} < \mathbf{V_m}$ de celles qui ont $\mathbf{V} > \mathbf{V_m}$. La variable $\mathbf{V}$ est choisie de façon à avoir des sous-jeux de données avec un minimum de variance vis-à-vis de la variable d'intérêt $\mathbf{y}$. On ne fait plus de bifurcation si les variances des sous-jeux de données ne sont pas plus faibles que le jeu de données amont.

A chaque grossesse est donc associéee une feuille dans l'arbre, la valeur de **y** prédite pour cette grossesse est la moyenne des valeurs de **y** pour les grossesses appartenant à la même feuille. On note : $x \mapsto h(x)$ une fonction qui associe la valeur prédite de **y** pour une grossesse **x** selon un arbre de décisions binaires. On a alors :

$$f_{RF}(x) = \sum_{i=1}^{N} \gamma_i h_i(x)$$

avec N le nombre d'arbres de décisions binaires dans la forêt et γ_i une pondération de la valeur renvoyée par la fonction h_i.

On construit itérativement le modèle $F_m()$ à partir du modèle de la forêt aléatoire contenant m-1 arbres : $F_{m-1}()$ et un nouvel arbre de décisions binaires h_m grâce à la méthode de descente du gradient via l'algorithme suivant :

$$\begin{cases} F_m(x) & = F_{m-1}(x) - \gamma_m \sum_{i=1}^{\mathbf{n_{LS}}} \nabla_F L(y_i, F_{m-1}(x_i)) \\ \gamma_m & = arg\min_{\gamma} \sum_{i=1}^{\mathbf{n_{LS}}} L(y_i, F_{m-1}(x_i) - \gamma \frac{\partial L(y_i, F_{m-1}(x_i))}{\partial F_{m-1}(x_i)}) \end{cases}$$

Avec L correspondant à la fonction de perte c'est-à-dire la fonction des moindres carrés.

GRADIENT BOOSTING

Un Gradient Boosting est un modèle additif qui se construit comme étant une somme pondérée d'arbres de décisions binaires.

Un arbre de décisions binaires correspond à une structure qui permet à partir d'une racine de scinder un jeu de données amont en deux sous-jeux de données à chaque bifurcation dans l'arbre (branche). Un chemin suivi le long des branches se termine à une feuille où il n'y ainsi plus de bifurcation. A une bifurcation, le jeu de données est séparé à partir d'une variable (**V**) et une valeur associée ($\mathbf{V_m}$) de sorte à séparer les grossesses qui ont $\mathbf{V} < \mathbf{V_m}$ de celles qui ont $\mathbf{V} > \mathbf{V_m}$. La variable **V** est choisie de façon à avoir des sous-jeux de données avec un minimum de variance vis-à-vis de la variable d'intérêt **y**. Dans le Gradient Boosting, tous les arbres ont la même profondeur.

A chaque grossesse est donc associée une feuille dans l'arbre, la valeur de **y** prédite pour cette grossesse est la moyenne des valeurs de **y** pour les grossesses appartenant à la même feuille. On note : $x \mapsto h(x)$ Une fonction qui associe la valeur prédite de **y** pour une grossesse **x** selon un arbre de décisions binaires. On a alors :

$$f_{GB}(x) = \sum_{i=1}^{N} \gamma_i h_i(x)$$

avec N le nombre d'arbres de décisions binaires utilisés dans la méthode et γ_i une pondération de la valeur renvoyée par la fonction h_i.

On construit itérativement le modèle $F_m()$ à partir du modèle de la forêt aléatoire contenant m-1 arbres : $F_{m-1}()$ et un nouvel arbre de décisions binaires h_m grâce à la méthode de descente du gradient via l'algorithme suivant :

$$\begin{cases} F_m(x) & = F_{m-1}(x) - \gamma_m \sum_{i=1}^{\mathbf{n_{LS}}} \nabla_F L(y_i, F_{m-1}(x_i)) \\ \gamma_m & = \underset{\gamma}{arg\,min} \sum_{i=1}^{\mathbf{n_{LS}}} L(y_i, F_{m-1}(x_i) - \gamma \frac{\partial L(y_i, F_{m-1}(x_i))}{\partial F_{m-1}(x_i)}) \end{cases}$$

Avec L correspondant à la fonction de perte c'est-à-dire la fonction des moindres carrés.

ANNEXE B

MÉTHODES DE CLASSIFICATION

Annotations

- **LS** = Learning Set = {données du jeu d'apprentissage}
- **TS** = Testing Set = {données du jeu test}
- $\mathbf{X} \in \mathscr{M}_{np}$ = **LS** $\bigcup$ **TS** = ensemble des données
- $\mathbf{x} \in \mathscr{M}_{1p} \subset \mathbf{X}$ = une grossesse
- $\mathbf{Z_m} \in \mathscr{M}_{n1}(\{0,1,...,m-1\})$ = une variable catégorielle (m catégories) d'intérêt pour chaque grossesse
- $\mathbf{z_m} \in \{0,1,...,m-1\}$ = une variable catégorielle (m catégories) d'intérêt pour une grossesse
- $\mathbf{n_{LS}}$ = nrow(**LS**) = nombre de grossesses dans le jeu d'apprentissage
- $\mathbf{n_{TS}}$ = nrow(**TS**) = nombre de grossesses dans le jeu test
- $\mathbf{n} = \mathbf{n_{LS}} + \mathbf{n_{TS}}$ = nombre total de grossesses

Une méthode de classification consiste à trouver une fonction $g : \mathbb{R}^p \to \{0,1,...m-1\}$ qui attribue pour chaque grossesse $\mathbf{x}$, une des $\mathbf{m}$ modalités de la variable $\mathbf{z_m}$.

Pour évaluer une méthode de classification bimodale vis-à-vis de la catégorie 1 (par rapport à la catégorie 0), on calcule le coefficient de Dice :

$$Dice = \frac{TP}{TP + 1.3FN + 0.7FP}$$

avec $TP = card(\{\widetilde{z_m} = 1 \cap z_m = 1\})$, $FP = card(\{\widetilde{z_m} = 1 \cap z_m = 0\})$ et $FN = card(\{\widetilde{z_m} = 0 \cap z_m = 1\})$

Pour évaluer une méthode de classification multimodale, on calcule un taux de réussite χ :

$$\chi = \frac{tr(M)}{sum(M)}$$

avec $M \in \mathscr{M}_m(\mathbb{R})$ la matrice de contingence : $\forall i, j \in [\![1;m]\!], M_{ij} = card(\{\widetilde{z_m} = i \cap z_m = j\})$

CLASSIFICATIONS NAÏVES

Cette méthode s'apparente à une prédiction au hasard pour attribuer une catégorie à une grossesse. Il peut il y a voir plusieurs façons de raisonner :

— On prédit toujours la modalité d'intérêt.

$$g_{\text{alltrue}}(x) = \{k\}, \text{avec k la modalité d'intérêt}$$

— On prend au hasard la modalité d'un individu qui était présent dans l'échantillon.

$$g_{proba}(x) = {}^{k}z_m, \text{ avec } k \in [\![1;\mathbf{n_{LS}}]\!] \text{une personne au hasard dans le jeu d'apprentissage}$$

RÉGRESSION LOGISTIQUE

Le modèle de la régression logistique est une méthode de classification qui combine les résultats de plusieurs régressions linéaires à l'utilisation de fonctions *logit()*.

Pour chaque modalité $Z_m\{k\}$ de la variable Z_m on réalise une régression linéaire combinée à la fonction *logit()* :

$$r_k(x) = \frac{1}{1 + e^{-{}^{T}\beta_k.x}}$$

avec $\forall k \in [\![0;m-1]\!], \beta_k \in \mathbb{R}^p$

On peut alors définir une probabilité d'appartenance de la variable Z_m pour chaque modalité :

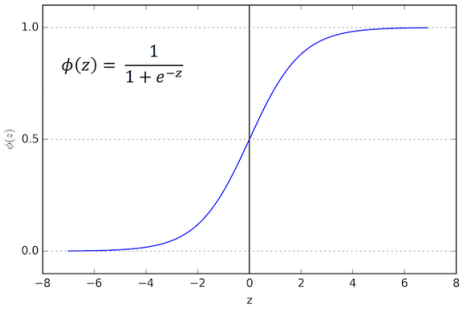

FIGURE *B.1* – *Fonction logit*

$$\forall k \in [\![0;m-1]\!], h_k(x) = p(z_m = \{k\}) = \frac{e^{{}^{T}\beta_k.x}}{\sum_i e^{{}^{T}\beta_i.x}} = \frac{1}{\sum_{i \neq k} e^{{}^{T}\beta_i.x} + e^{-{}^{T}\beta_k.x}}$$

La régression logistique prédit alors la modalité qui a la grande probabilité prédite :

$$g_{RL}(x) = arg\max_k(h_k(x))$$

Pour trouver chaque vecteur $(\beta_k)_{k \in [\![0;m-1]\!]}$ on résout le problème de minimisation suivant :

$$min_{\beta} \frac{1}{2} {}^{T}\beta\beta + \sum_{i \in LS} \log(\exp(-Z_m[i]({}^{T}X[i,].\beta)) + 1)$$

MACHINE À VECTEURS DE SUPPORT

Une méthode utilisant une machine à vecteurs de support est une méthode de classification qui utilise un raisonnement géométrique pour classifier les grossesses. Le principe est de réaliser un hyperplan pour séparer géométriquement deux modalités distinctes.

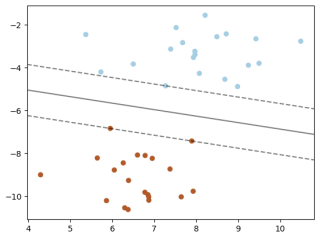

FIGURE *B.2* – *SVM*

Pour chaque couple de modalités distinctes possibles, on attribue ainsi un hyperplan pour séparer ces deux modalités : on renomme les deux modalités par les valeurs $\{-1;1\}$. On note $\Theta = \{-1;1\}^n$ et pour trouver l'hyperplan, il faut résoudre le problème :

$$\begin{cases} \min\limits_{\alpha} \frac{1}{2}\,{}^T\alpha.{}^TX.\Theta.{}^T\Theta.X.\alpha - {}^Te.\alpha \\ {}^T\Theta.\alpha = 0_{\mathbb{R}^p} \\ \forall i \in [\![1;p]\!] : 0 \leq \alpha_i \leq 1 \end{cases}$$

avec $\alpha \in \mathbb{R}^p$, e le vecteur colonne de $\mathbb{R}^p$ contenant des 1.

Pour prédire la modalité d'une nouvelle grossesse il suffit alors de regarder géométriquement où se trouve le point correspondant à la grossesse par rapport à l'hyperplan créé. Pour déterminer la modalité $\{-1;1\}$ d'une grossesse on regarde alors la valeur de la fonction :

$$sign(\sum_{i=1}^{n_{LS}} \theta_i \alpha_i\,{}^TX_i.x)$$

Pour déterminer la modalité d'une grossesse, il faut alors déterminer la modalité qui est la plus prédite lorsque l'on réalise tous les hyperplans pour chaque couple de modalités.

MACHINE À VECTEURS DE SUPPORT À NOYAU

Une méthode utilisant une machine à vecteurs de support à noyau est une méthode de classification qui utilise un raisonnement géométrique pour classifier les grossesses. Le principe est de réaliser un hyperplan pour séparer géométriquement deux modalités distinctes en passant par une étape préliminaire qui utilise une fonction noyau pour réorganiser géométriquement les coordonnées des points correspondant aux grossesses pour améliorer la classification en hyperplan.

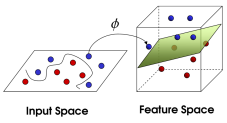

FIGURE *B.3* – *Utilisation d'une fonction noyau*

Pour chaque couple de modalités distinctes possibles, on attribue ainsi un hyperplan pour séparer ces deux modalités : on renomme les deux modalités par les valeurs $\theta = \{-1;1\}$. On note $\Theta = \{-1;1\}^n$ et pour trouver l'hyperplan, il faut résoudre le problème :

$$\begin{cases} \min\limits_{\alpha} \frac{1}{2}\,{}^{T}\alpha.Q.\alpha - {}^{T}e.\alpha \\ {}^{T}\Theta.\alpha = 0_{\mathbb{R}^p} \\ \forall i \in [\![1;p]\!] : 0 \leq \alpha_i \leq 1 \end{cases}$$

avec $\alpha \in \mathbb{R}^p$, e le vecteur colonne de $\mathbb{R}^p$ contenant des 1, $Q \in \mathscr{M}_n, Q_{ij} = \theta_i\theta_j k(X_i, X_j)$, k() une fonction à noyau.

Pour prédire la modalité d'une nouvelle grossesse il suffit alors de regarder géométriquement où se trouve le point correspondant à la grossesse par rapport à l'hyperplan créé. Pour déterminer la modalité $\{-1;1\}$ d'une grossesse on regarde alors la valeur de la fonction :

$$sign(\sum_{i=1}^{n_{LS}} \theta_i \alpha_i k(X_i, x))$$

Pour déterminer la modalité d'une grossesse, il faut alors déterminer la modalité qui est la plus prédite lorsque l'on réalise tous les hyperplans pour chaque couple de modalités.

On peut utiliser plusieurs fonctions à noyau :

— Gaussien : $k(x,x') = exp^{-\frac{<x-x'|x-x'>^2}{2\sigma^2}}$

— Polynomial : $k(x,x') = (<x|x'> +1)^d$

— Sigmoïde : $k(x,x') = \tanh(\alpha x^T x' + c)$

RÉSEAU DE NEURONES ARTIFICIELS

Le modèle du réseau de neurones artificiels s'inspire de la structure d'un neurone : il y a une couche de neurones d'entrée (qui n'ont pas de prédécesseurs), une couche cachée à N neurones et la couche de sortie qui a autant de neurone que de modalités possibles donc pour prédire $\mathbf{z_m}$ il y a m neurones.

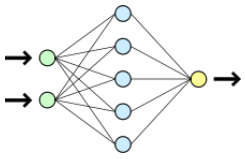

FIGURE B.4 – *Réseau de Neurones Artificiels*

Pour le i^{eme} neurone de la couche $n+1$ possédant k_n prédécesseurs prend donc une valeur particulière en fonction de la valeur des k_n neurones précédents :

$$\begin{aligned} U_{n+1}[i] &= a(U_n) \\ &= softmax({}^{T}\Omega i_n.U_n) \\ &= a(u_n[1],...,u_n[k_n]) \\ &= softmax(\sum_{j=1}^{k_n} {}^{T}\omega i_{nj} u_n[j]) \end{aligned}$$

avec $U_n \in \mathscr{M}_{k_n 1}(\mathbb{R})$: la valeur des neurones de la n^{eme} couche , $\Omega i_n \in \mathscr{M}_{k_n 1}(\mathbb{R})$.

En posant $\Omega_n = (\Omega 1_n, \Omega 2_n, ..., \Omega k_n{}_n), \Omega_n \in \mathscr{M}_{k_n k_{n+1}}(\mathbb{R})$, on a $\forall n$:

$$\begin{aligned} U_{n+1} &= softmax({}^T\Omega_n.U_n) \\ &= softmax({}^T\Omega_n.softmax({}^T\Omega_{n-1}.....softmax({}^T\Omega_1.U_1).....)) \end{aligned}$$

On a donc un modèle de réseau de neurones artificiels via la fonction :

$$g_{RdN}(x) = arg\max(softmax({}^T\Omega_{m-1}.softmax({}^T\Omega_{m-2}.....softmax({}^T\Omega_1.x)).....))$$

avec m le nombre de couches de neurones, $\forall i \in [\![1; m-1]\!] \Omega_i \in \mathscr{M}_{k_i k_{i+1}}(\mathbb{R})$ et k_i le nombre de neurones dans la i^{eme} couche.

Les matrices $(\Omega_i)_{i \in [\![1;m-1]\!]}$ sont calculées itérativement avec l'algorithme du gradient en minimisant l'erreur quadratique moyenne.

BAGGING

On va former un boostrap de notre jeu données et sur chacun des sous-jeux, on va attribuer un arbre de décisions binaires qui va constituer un apprenant faible pour l'échantillon global. On réalise donc un nombre N de sous-jeux de données provenant des grossesses initialement renseignées sur lesquels on effectue un arbre de décisions binaires.

FIGURE B.5 – *Illustration de la méthode Bagging*

Un arbre de décisions binaires correspond à une structure qui permet à partir d'une racine de scinder un jeu de données amont en deux sous-jeu de données à chaque bifurcation dans l'arbre (branche). Un chemin suivi le long des branches se termine à une feuille où il n'y ainsi plus de bifurcation. A une bifurcation, le jeu de données est séparé à partir d'une variable ($\mathbf{V}$) et une valeur associée ($\mathbf{V_m}$) de sorte à séparer les grossesses qui ont $\mathbf{V} < \mathbf{V_m}$ de celles qui ont $\mathbf{V} > \mathbf{V_m}$. La variable $\mathbf{V}$ est choisie de façon à avoir des sous-jeux de données avec un minimum de variance vis-à-vis de la variable d'intérêt $\mathbf{z_m}$. On ne fait plus de bifurcation si les variances des sous-jeux de données ne sont pas plus faibles que le jeu de données amont.

La modalité d'une nouvelle grossesse s'obtient alors par un vote à la majorité.

FORÊTS ALÉATOIRES

Une forêt aléatoire est un modèle additif qui se construit comme étant une somme pondérée d'arbres de décisions binaires.

Un arbre de décisions binaires correspond à une structure qui permet à partir d'une racine de scinder un jeu de données amont en deux sous-jeux de données à chaque bifurcation dans l'arbre

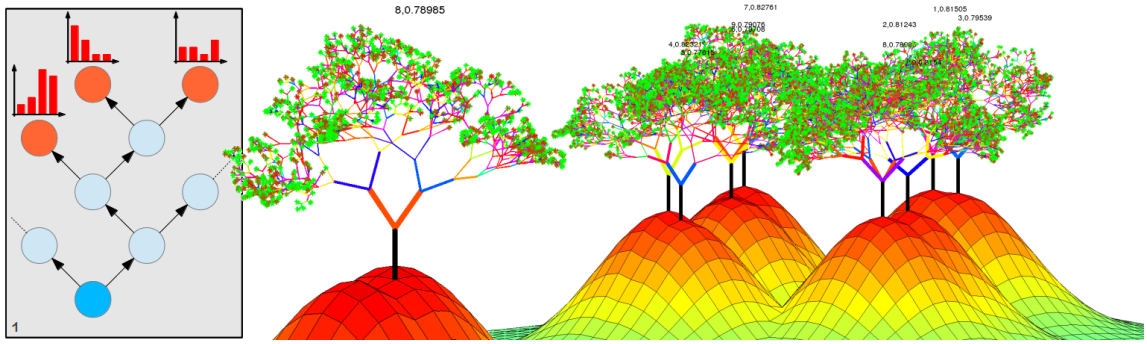

FIGURE *B.6* – *Forêt Aléatoire*

(branche). Un chemin suivi le long des branches se termine à une feuille où il n'y ainsi plus de bifurcation. A une bifurcation, le jeu de données est séparé à partir d'une variable (**V**) et une valeur associée ($\mathbf{V_m}$) de sorte à séparer les grossesses qui ont $\mathbf{V} < \mathbf{V_m}$ de celles qui ont $\mathbf{V} > \mathbf{V_m}$. La variable **V** est choisie de façon à avoir des sous-jeux de données avec un minimum de variance vis-à-vis de la variable d'intérêt $\mathbf{z_m}$. On ne fait plus de bifurcation si les variances des sous-jeux de données ne sont pas plus faibles que le jeu de données amont.

A chaque grossesse est donc associée une feuille dans l'arbre, la valeur de $\mathbf{z_m}$ prédite pour cette grossesse est la modalité de $\mathbf{z_m}$ la plus présente pour les grossesses appartenant à la même feuille. On note : $x \mapsto h(x)$ Une fonction qui associe les probabilités prédites de $\mathbf{z_m}$ pour une grossesse **x** selon un arbre de décisions binaires. On a alors :

$$g_{RF}(x) = arg\max(\sum_{i=1}^{N} \gamma_i h_i(x))$$

avec N le nombre d'arbres de décisions binaires dans la forêt et γ_i une pondération de la valeur renvoyée par la fonction h_i.

On construit itérativement le modèle $F_m()$ à partir du modèle de la forêt aléatoire contenant m-1 arbres : $F_{m-1}()$ et un nouvel arbre de décisions binaires h_m grâce à la méthode de descente du gradient via l'algorithme suivant :

$$\begin{cases} F_m(x) & = F_{m-1}(x) - \gamma_m \sum_{i=1}^{\mathbf{n_{LS}}} \nabla_F L(y_i, F_{m-1}(x_i)) \\ \gamma_m & = arg\min_{\gamma} \sum_{i=1}^{\mathbf{n_{LS}}} L(y_i, F_{m-1}(x_i) - \gamma \frac{\partial L(y_i, F_{m-1}(x_i))}{\partial F_{m-1}(x_i)}) \end{cases}$$

Avec L correspondant à la fonction de perte c'est-à-dire la fonction des pertes exponentielles.

BOOSTING

Un Boosting est un modèle additif qui se construit comme étant une somme pondérée d'arbres de décisions binaires.

Un arbre de décisions binaires correspond à une structure qui permet à partir d'une racine de scinder un jeu de données amont en deux sous-jeux de données à chaque bifurcation dans l'arbre (branche). Un chemin suivi le long des branches se termine à une feuille où il n'y ainsi plus de bifurcation. A une bifurcation, le jeu de données est séparé à partir d'une variable (**V**) et une valeur associée ($\mathbf{V_m}$) de sorte à séparer les grossesses qui ont $\mathbf{V} < \mathbf{V_m}$ de celles qui ont $\mathbf{V} > \mathbf{V_m}$. La variable **V** est choisie de façon à avoir des sous-jeux de données avec un minimum de variance vis-à-vis de la variable d'intérêt $\mathbf{z_m}$. Dans le Boosting, tous les arbres ont la même profondeur.

A chaque grossesse est donc associée une feuille dans l'arbre, la valeur de $\mathbf{z_m}$ prédite pour cette grossesse est la modalité de $\mathbf{z_m}$ la plus présente pour les grossesses appartenant à la même feuille. On note : $x \mapsto h(x)$ une fonction qui associe la valeur prédite de $\mathbf{z_m}$ pour une grossesse **x** selon un arbre de décisions binaires. On a alors :

$$g_{GB}(x) = arg\max(\sum_{i=1}^{N} \gamma_i h_i(x))$$

avec N le nombre d'arbres de décisions binaires utilisés dans la méthode et γ_i une pondération de la valeur renvoyée par la fonction h_i.

Avec L correspondant à la fonction de perte c'est-à-dire la fonction des pertes exponentielles.

PLUS PROCHES VOISINS

On va trouver les k individus qui sont les plus proches voisins géométriquement d'une grossesse test x. On choisit ainsi la modalité de la grossesse test en prenant la modalité la plus fréquente dans ce voisinage.

ANNEXE C

MÉTHODES DE RÉDUCTION DE DIMENSION

Annotations

- $n \in \mathbb{N}^*$: nombre total de grossesses
- $p \in \mathbb{N}^*$: nombre de variables renseignées par grossesse
- $d \in \mathbb{N}^* \leq p$: nombre de variables après réduction

Une méthode de réduction de dimension consiste à trouver une fonction $h : \mathbb{R}^p \to \mathbb{R}^d$. Le but d'une telle méthode est de pouvoir extraire les informations du jeu de données initial en synthétisant les **p** variables en **d** variables. Cela permet aussi de réduire les temps de calcul algorithmique nécessaire pour effectuer les modèles.

Il y a plusieurs façons de raisonner pour réduire les dimensions : raisonner sur la variance du jeu initial, raisonner sur la structure géométrique initiale du jeu de données et raisonner sur la conservation des voisinages.

FAMD : ANALYSE FACTORIELLE DE DONNÉES MIXTES

La FAMD est une méthode linéaire de réduction de données, c'est-à-dire que l'on va créer la matrice W=$(W_1, W_2, ..., W_k)$ où toute colonne de cette matrice s'exprime comme une combinaison linéaire des colonnes de X.

L'algorithme permettant la FAMD travaille de manière itérative en trouvant la meilleure famille orthonormale de vecteurs qui maximise la variance du jeu de données.

J'ai deux façons pour choisir le nombre optimal de vecteurs pour composer la matrice W qui repose sur le calcul de $\lambda_i = \max(\sum_{d \in quanti} r^2(W_i, X_{,d}) + \sum_{d' \in catego} \eta^2(W_i, X_{,d'}))$:

1. règle de Kaiser-Guttman : on complète la matrice W par W_i si $\lambda_i > 1$
2. règle de Karlis-Saporta-Spinaki : on complète la matrice W par W_i si $\lambda_i > 1 + 2\sqrt{\frac{p-1}{n-1}}$

kPCA : Analyse en Composantes Principales à Noyau

C'est une méthode de réduction non-linéaire mais uniquement de variables quantitatives. On utilise une méthode à noyau, c'est-à-dire que l'on va changer la représentation de notre matrice X initiale par le passage par la matrice noyau K.

Une méthode noyau est définie par une fonction $\Phi : \mathbb{R}^n \to F$, avec F un espace d'arrivée de n'importe quelle dimension (même infinie). La matrice K est alors définie par :

$$K_{i,j} = K(X_{,i}, X_{,j}) = < (\Phi(X_{,i}) | \Phi(X_{,j})) >$$

J'ai choisi personnellement d'utiliser trois méthodes à noyaux usuelles :

1. Gaussien : $k(x,x') = exp^{-\frac{<x-x'|x-x'>^2}{2\sigma^2}}$
2. Polynomial : $k(x,x') = (<x|x'> +1)^d$
3. Sigmoïde : $k(x,x') = \tanh(\alpha x^T x' + c)$

De manière analogue à la FAMD on va construire la matrice W en maximisant la covariance, mais cette fois-ci celle de la matrice K.

On détermine le nombre optimal de variables à créer en suivant la règle de Kaiser-Guttman ou celle de Karlis-Saporta-Spinaki :

1. règle de Kaiser-Guttman : on complète la matrice W par W_i si $\lambda_i > 1$
2. règle de Karlis-Saporta-Spinaki : on complète la matrice W par W_i si $\lambda_i > 1 + 2\sqrt{\frac{p-1}{n-1}}$

MDS : Echelle multidimensionnelle

C'est une méthode pour réduire notre jeu de données qui essaie de conserver la notion de distance (via la notion de norme) qu'il y a entre chaque individu dans l'espace initial des données vers l'espace d'arrivée.

On définit donc notre jeu de données initial par une matrice de distance D et une matrice de proximité P. On cherche alors à trouver une fonction $f : \mathbb{R}^p \to \mathbb{R}^d$ qui minimise le *Stress* :

$$\begin{cases} Stress &= \sum_{i,j}(f(P_{i,j}) - D_{i,j})^2 \\ f &= \underset{f}{arg\min}(Stress) \end{cases}$$

Pour trouver la dimension optimale de notre espace d'arrivée, on doit cette fois-ci déterminer la dimension à partir de laquelle j'obtiens un score *Stress* excellent, sachant que :

— $Stress > 0.20$: mauvais

— $0.10 < Stress < 0.20$: passable

— $0.025 < Stress < 0.05$: bien

— $Stress < 0.025$: excellent

La fonction f est linéaire dans le cas de la MDS métrique.
La fonction f est monotone dans le cas de la MDS non-métrique.

ISOMAP

La méthode Isomap cherche à conserver la notion de distance mais avec une approche de distance différente de celle de la norme : la distance géodésique. Cette distance prend en compte le fait que pour aller d'un point A à un point B, on doit emprunter un chemin qui n'est pas direct mais qu'on doit passer par là où sont les autres points. Pour faire un parallèle avec la réalité la distance euclidienne correspond à la distance à vol d'oiseau alors que la distance géodésique prend en compte que l'on doit suivre une route et que la distance peut alors être beaucoup plus longue.

Une fois que la matrice de distance D et de proximité P sont obtenues, on cherche alors à trouver une fonction $f : \mathbb{R}^p \to \mathbb{R}^d$ qui minimise le $Stress$:

$$\begin{cases} Stress &= \sum_{i,j}(f(P_{i,j}) - D_{i,j})^2 \\ f &= \underset{f}{arg\,\min}(Stress) \end{cases}$$

Pour trouver la dimension de notre espace d'arrivée optimale, on doit cette fois-ci déterminer la dimension à partir de laquelle j'obtiens un score $Stress$ excellent, sachant que :

— $Stress > 0.20$: mauvais

— $0.10 < Stress < 0.20$: passable

— $0.025 < Stress < 0.05$: bien

— $Stress < 0.025$: excellent

LLE : PLONGEMENT LOCALE LINÉAIRE

C'est une méthode qui permet de conserver l'allure locale du jeu de données initial. On cherche ainsi à ce qu'un individu dans la géométrie initiale se retrouve avec les mêmes individus voisins dans la géométrie finale. Un individu n'est alors uniquement impacté par les individus présents dans son voisinage et non plus par l'ensemble des individus.

On doit alors choisir arbitrairement le nombre d'individus qui définit un voisinage. On cherche alors pour chaque individu à minimiser la matrice de covariance locale qu'il a avec ses voisins puis

à minimiser l'erreur de reconstruction dans l'espace d'arrivée :

$$\begin{cases} C & = & \underset{\|C\|=1}{} \|X - C\Gamma(X)\| \\ W & = & \underset{\|W\|=1}{} \|W - CW\| \end{cases}$$

avec $\Gamma \in M_{n,p}$ la matrice des voisins de X, $C \in M_{n,n}$

T-SNE : PLONGEMENT STOCHASTIQUE DES VOISINS T-DISTRIBUÉS

Cette méthode repose aussi sur la conservation du voisinage mais en passant par des probabilités conditionnelles. On calcule une matrice de distance D entre les individus, puis on crée une matrice de similarité P entre chaque paire d'individus pour chercher la matrice W telle que :

$$\begin{cases} P_{i,j} = p(j|i) = \dfrac{e^{-\frac{D_{i,j}^2}{2\sigma_i^2}}}{\sum_{k \neq i} e^{-\frac{D_{i,k}^2}{2\sigma_i^2}}} \approx Q_{i,j} \\ Q_{i,j} = q(j|i) = \dfrac{e^{-\|W_{,i} - W_{,k}\|^2}}{\sum_{k \neq i} e^{-\|W_{,i} - W_{,k}\|^2}} \end{cases}$$

σ_i est un coefficient déterminé par l'algorithme pour s'approcher de la perplexité mathématique souhaitée.

On cherche à minimiser la divergence de Kullback-Leiber : $C = \sum_{i,j} P_{i,j} \log(\frac{P_{i,j}}{Q_{i,j}})$.

BIBLIOGRAPHIE

[**1**] FRANÇOISE VENDITTELLI, OLIVIER RIVIÈRE, GÉRARD BRÉART, *Is prenatal identification of fetal macrosomia useful ?*, 2012.

[**2**] FRANÇOISE VENDITTELLI, OLIVIER RIVIÈRE, BRIGITTE NEVEU, DIDIER, LÉMERY, *Does induction of labor for constitutionally large-for-gestational-age fetuses identified in utero reduce maternal morbidity ?*, 2014.

[**3**] A. EGO, C. PRUNET, B. BLONDEL, M. KAMINSKI, F. GOFFINET, J. ZEITLIN, *Courbes de croissance in utero ajustées et non ajustées adaptées à la population française. II - Comparaison à des courbes existantes et apport de l'ajustement*, 2015.

[**4**] J. GARDOSI, A. CHANG, B. KALYAN, D. SAHOTA, E. M. SYMONDS, *Customised antenatal growth charts*, 1992.

[**5**] FRANK P. HADLOCK, RUSSELT L. DETER, RONALD B. HARRIST, SEUNG K. PARK, *Fetal Biparietal Diameter : A critical Re-evaluation of the Relation to Menstrual Age by Means of Real-time Ultrasound*, 1982.

[**6**] CORY LESMEISTER, *Mastering Machine Learning with R*, 2015.

[**7**] BRETT LANTZ, *Machine Learning with R*, 2013.