HAL open science

Contributions à la détection de marqueurs et à l'analyse de survie en oncologie

Mathilde Sautreuil

▶ **To cite this version:**

Mathilde Sautreuil. Contributions à la détection de marqueurs et à l'analyse de survie en oncologie. Machine Learning [stat.ML]. Université Paris-Saclay, 2021. Français. NNT : 2021UPAST005 . tel-03278955

HAL Id: tel-03278955

https://theses.hal.science/tel-03278955v1

Submitted on 6 Jul 2021

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

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

Contributions à la détection de marqueurs et à l'analyse de survie en oncologie

Thèse de doctorat de l'Université Paris-Saclay

École doctorale n° 573, Interfaces
Spécialité de doctorat : Mathématiques appliquées
Unité de recherche : Université Paris-Saclay, CentraleSupélec, Mathématiques et Informatique pour la Complexité et les Systèmes, 91190, Gif-sur-Yvette, France.
Référent : CentraleSupélec

Thèse présentée et soutenue à Gif sur Yvette, le 12 février 2021, par

Mathilde SAUTREUIL

Composition du jury :

Daniel Gautheret Professeur, I2BC, Université Paris-Saclay	Président
Rodrigue Allodji Chargé de recherche, Épidémiologie des radiations, Gustave Roussy, HDR	Rapporteur & Examinateur
Vlad Stefan Barbu Maître de conférences, LMRS, Université de Rouen, HDR	Rapporteur & Examinateur
Nathalie Vialaneix Directrice de recherche, Unité MIA-T, INRAE de Toulouse	Rapporteur & Examinatrice
Angelina Roche Maître de conférences, CEREMADE, Université Paris Dauphine	Examinatrice
Paul-Henry Cournède Professeur, MICS, CentraleSupélec, Université Paris-Saclay	Directeur
Sarah Lemler Maître de conférences, MICS, CentraleSupélec, Université Paris-Saclay	Co-encadrante
Stefan Michiels Directeur de recherche, Oncostat, Gustave Roussy	Invité

Thèse de doctorat

NNT : 2021UPAST005

Thèse de doctorat de l'Université Paris-Saclay
préparée à CentraleSupélec

École doctorale n°573 Interfaces
Spécialité de doctorat : Mathématiques appliquées

Mathilde SAUTREUIL

Contributions à la détection de marqueurs et à l'analyse de survie en oncologie

Thèse présentée et soutenue à Gif sur Yvette, le 12 février 2021

Composition du jury :

Daniel Gautheret I2BC, Université Paris-Saclay	Président
Rodrigue Allodji Épidémiologie des radiations, Gustave Roussy	Rapporteur & Examinateur
Vlad Stefan Barbu LMRS, Université de Rouen	Rapporteur & Examinateur
Nathalie Vialaneix Unité MIA-T, INRAE de Toulouse	Rapporteur & Examinatrice
Angelina Roche CEREMADE, Université Paris Dauphine	Examinatrice
Paul-Henry Cournède MICS, CentraleSupélec, Université Paris-Saclay	Directeur
Sarah Lemler MICS, CentraleSupélec, Université Paris-Saclay	Co-encadrante
Stefan Michiels Oncostat, Gustave Roussy	Invité

université PARIS-SACLAY | ÉCOLE DOCTORALE Interfaces: matériaux, systèmes, usages | CentraleSupélec

Remerciements

Tout naturellement, je tiens, pour commencer, à remercier mes directeur.rices de thèse. Merci Paul-Henry de m'avoir donné l'opportunité de réaliser cette thèse et d'avoir su créer un environnement agréable au sein de l'équipe *biomathematics*. Merci Sarah, pour ta grande disponibilité durant ces trois années de thèse et de ton soutien dans les moments plus difficiles.

Je voudrais remercier chaleureusement les différents membres du jury d'avoir accepté de juger mes travaux de thèse. Je suis reconnaissante envers mes rapporteurs Vlad Barbu, Nathalie Vialaneix et Rodrigue Allodji pour le temps consacré à la relecture de mon manuscrit ainsi que vos retours et suggestions d'amélioration. Un merci particulier à Vlad, de m'avoir encouragé dès ma L3 de mathématiques et de continuer à le faire aujourd'hui. Merci à Nathalie Vialaneix d'avoir accepté de rapporter ma thèse dans un délai très court. Je remercie Daniel Gautheret et Angelina Roche d'avoir accepté de participer à mon jury de thèse. Angelina, merci de ta bienveillance lorsqu'on se croisait à Rouen et lors des JdS.

Ce paragraphe est dédié à remercier l'ensemble de l'équipe *biomathematics* pour l'atmosphère agréable qui font régner dans l'open-space. Commençons par une petite dédicace pour la team Kwak : Mahmoud et Elvire, vous avez rendu les afterworks plus intéressants. Pour éviter un paragraphe trop long, je vais tout d'abord remercier, Antonin pour ta personnalité, toutes tes suggestions culturelles et de tes appels pour prendre des nouvelles, Gautier et Sylvain , pour les discussions concernant la Normandie, Gurvan, pour les différentes séances de sports, Walid, pour ton aide en keras, Yoann pour le bon moment de codeNames, Stefania, pour ta gentillesse, à Andreas pour les bons moments passés au laboratoire et en dehors. Pour ceux qui ont fait un passage rapide au sein de l'équipe : Dimitri, Blandine et Julie. Merci à Andreas, Elvire, Julie, Dimitri et Jun pour les très bons moments à Corfou, je garde un très bon souvenir de ce séminaire d'équipe. Merci aux nouvelles arrivantes : Laura et Agathe. Enfin, j'aimerais également remercier Véronique pour sa bienveillance tout le long de ma thèse.

Le paragraphe concernant l'open-space terminé, je vais passer à l'autre côté du laboratoire. Merci à Rémi et Guillaume, qui ont fait tout leur possible pour la réparation de fusion, dans ma phase critique de thèse et qui ont toujours été d'une grande rapidité pour répondre à mes questions. Un petit paragraphe va aux assistantes du laboratoire dont la vie du labo ne serait pas la même sans elles. Je voudrais remercier Fabienne pour les pauses discussions, qui me permettaient de me changer les idées et de respirer. Et un immense merci à ma Sylvie, d'avoir été là au moment les plus compliqués, à m'écouter et à me changer les idées. Je sais que tu passes par un moment assez compliqué et je suis de tout cœur avec toi. Merci à Ludovic pour ta gentillesse et ta bienveillance. Merci également à Jad pour ta gentillesse. Merci à vous deux de m'avoir tenu compagnie durant l'écriture de mon manuscrit cet été.

Merci à Chloé, pour les bons moments passés au sein du laboratoire, comme par exemple, les petites discussions lorsque l'open-space jouait au baby-foot, ainsi que ceux passés à l'extérieur, l'adada en est un exemple et de m'avoir donné le goût du running.

Un merci tout particulier à Myriam, Jun et Brice pour tous les bons moments partagés au laboratoire comme à l'extérieur. Jun, merci de m'avoir fait découvrir les bons restaurants asiatiques du coin et de Paris. Merci à

toi et Weichao pour les séances d'escalade et espérons qu'on pourra un jour courir le semi de Paris ensemble. Myriam, merci d'avoir été là, on est vite devenu plus que des collègues. J'espère qu'on pourra continuer longtemps nos footings blabla. Et je vais enfin avoir plus de temps pour faire les activités qu'on prévoit depuis des mois (couture, pâtisseries, shopping, etc). Merci également à Manu de nous accompagner lors des footings et ainsi éviter qu'on se perde dans la forêt. Merci à toi, Brice, pour toutes les parties de jeux toujours couplés avec de la bonne nourriture.

Je tenais également à remercier l'équipe Ginette, pour leur accueil si chaleureux ainsi que leur compréhension pour la finalisation de ce travail de thèse. Je m'y suis sentie très rapidement à l'aise grâce à votre bonne humeur et votre humour.

Je voudrais remercier les deux équipes de Rouen qui m'ont accueillis pendant mon alternance/stage et qui ont fait part d'encouragements lors de mon choix de me diriger vers une thèse. Parmi ces personnes, je voudrais particulièrement remercier Caroline et Nicolas. Tout d'abord, Caroline de m'avoir donné l'envie de continuer vers une thèse et surtout m'avoir fait comprendre que c'était possible. Merci à Nicolas de m'avoir toujours encourager et de m'avoir poussé à prendre plus confiance en moi. Je garde de très bons souvenirs de mon passage à Rouen. Vous m'avez également permis d'acquérir mes premières expériences d'enseignement et l'opportunité d'intervenir dans le master qui m'a formé.

Enfin, je remercie mes parents et mes deux soeurs qui ont contribué à la personne que je suis. Tout d'abord, Camille, ma soeur, dont le parcours a été semée d'embûches, Malgré tout cela, elle a réussi à soutenir brillamment sa thèse renforçant encore mon admiration. Merci à Ophélie, qui m'a montré que la différence est un atout. Je remercie, mes parents, de m'avoir laisser respirer quand j'en ai eu besoin. Je tiens aussi à remercier Nadine pour son soutien.

Enfin fini, tout commence. Merci à tous !

Résumé

La médecine personnalisée a une grande importance en oncologie, elle permet de personnaliser des traitements ou de prédire la réponse à ceux-ci. Elle nécessite la mise en œuvre des méthodes développées par les statisticiens, permettant au médecin de spécifier les caractéristiques du patient afin de proposer à celui-ci le parcours de soin adapté. Depuis l'arrivée du séquençage à haut-débit, les données permettant de caractériser les patients sont plus nombreuses, offrant un portrait moléculaire de ceux-ci. Les méthodes développées doivent donc maintenant prendre en considération le cadre de la grande dimension, quand le nombre de variables est supérieur au nombre de patients. L'objectif de cette thèse est d'étudier et développer des méthodes adaptées à la grande dimension pour la détection de marqueurs et l'analyse de survie en oncologie.

Cette thèse se divise en deux parties. Dans une première partie, nous nous intéressons à la détection de marqueurs en oncologie avec deux objectifs différents. Le premier objectif consiste à identifier les gènes signatures du cancer du rein à cellules claires (ccRCC). Celui-ci a fait l'objet d'une collaboration avec l'équipe du Dr Diana Tronik Le Roux. Pour cet objectif, nous utilisons deux démarches différentes. La première, plus classique, consiste à réaliser une analyse différentielle. La seconde, plus originale, consiste à coupler une sélection de variables à partir de l'analyse différentielle avec une méthode d'apprentissage. Le second objectif est d'étudier les méthodes de régularisation et de *Screening* pour mettre en évidence les gènes influençant la survie des patients. La stabilité de ces méthodes est également étudiée à partir d'un indice de similarité.

Dans la seconde partie de cette thèse, nous nous intéressons à la prédiction de la survie en grande dimension. Pour cela, nous étudions l'apport des réseaux de neurones pour prédire la durée de survie dans un cadre de grande dimension. Ces méthodes ont fait récemment leur ré-apparition en analyse de survie mais elles ont été peu étudiées en grande dimension. Nous distinguons deux approches de réseaux de neurones. La première est basée sur la modélisation de Cox. La seconde approche est basée sur une modélisation à temps discret. Cette dernière ayant été peu étudiée, nous nous concentrons donc sur celle-ci en adaptant les réseaux de neurones à la grande dimension. Nous comparons, dans cette thèse, un réseau de neurones basé sur le modèle de Cox, appelé Cox-nnet, avec ceux basés sur un modèle à temps discret que nous avons adaptés à la grande dimension. Nous prenons comme référence la prédiction à partir du modèle de Cox avec une procédure d'estimation de type Lasso. Nous avons créé un plan de simulations pour comparer les performances de ces méthodes sur la prédiction de la survie. Les données sont simulées à partir de différents modèles de survie (Cox, AFT et AH) pour avoir des données avec différents niveaux de complexité et en faisant varier la taille d'échantillon et le nombre de variables. La sparsité et la censure sont également prises en compte. Les différentes méthodes sont appliquées en utilisant les données ainsi simulées et nous illustrons également les performances des réseaux de neurones sur des jeux de données réelles. Nous pouvons conclure que dans la plupart des situations, le meilleur réseau de neurones est celui basé sur la modélisation de Cox (Cox-nnet). L'utilisation du réseau de neurones dans cette modélisation permet de gérer les effets non-linéaires ainsi que les interactions. Cependant, nous montrons que dans les cas les plus complexes, notamment en présence des risques non-proportionnels et de courbes de survie se croisant, le réseau de neurones à plusieurs couches, basé sur une modélisation à temps discret, est le plus performant.

Mots-clés : Analyse de survie, détection de marqueurs, réseaux de neurones, régularisation, *Screening*, modèle de Cox, modèle AFT, modèle AH, grande dimension, cancer, ccRCC, *Immune-Checkpoints*.

Abstract

Personalized medicine plays an important role in oncology. It enables to personalize treatments or to predict the response to treatments. This type of medicine needs the development of methods by statisticians, in which doctors can specify each patient's characteristics to propose the adapted care pathway. With the advent of high-throughput sequencing, the data enabling to characterize the patients are more voluminous and offer molecular portraits of patients. The models developed must now consider the high-dimensional context when the number of covariates is larger than the number of patients. This thesis's objective consists of studying and developing methods adapted to the high-dimension for marker detection and survival analysis in oncology.

This thesis is divided into two parts. In the first one, we are interested in marker detection in oncology with two different objectives. The first one consists in identifying the genes responsible for Clear Cell Renal Cell Carcinoma (ccRCC). We use two approaches to respond to this objective. For the first approach, which is more classic, we realize a differential analysis. For the second one, more original, we couple variable selection from differential analysis with a machine learning method. Then, the second objective of this part consists in studying regularization and some screening methods to underline the genes with strong influence on the survival of patients. The stability of these methods is also evaluated with a similarity index.

In the second part of this thesis, we are interested in predicting survival in high-dimension. For this goal, we study the potential of neural networks to predict the survival duration in a high-dimensional context. These methods make a come-back in survival analysis but few of them is studied in high-dimension. We distinguish two strategies for neural networks in survival analysis and study them in various scenarios. The first one is based on the Cox model and the second one is based on a discrete-time model. This second approach is less studied than the one based on the Cox model, and we propose several adaptations to the high-dimensional setting. We present a comparison study. We compare a neural network based on the Cox model called Cox-nnet with those based on a discrete-time model adapted to the high-dimension. The Lasso procedure using the Cox partial log-likelihood is used as a benchmark. We create a simulation plan to make this comparison more relevant. The data are simulated from different survival models (Cox, AFT, and AH) to have data of different complexity levels with various sample sizes and numbers of covariates. We also study the effect of censorship and sparsity. We consider the Concordance index and the Integrated Brier Score to compare the performances of the different procedures. Finally, we use two classical real datasets for comparison. We conclude that in most situations, the best method is the one based on the Cox framework in which the neural network is used to handle nonlinear effects as well as interactions (Cox-nnet). However, the model in which the neural network directly predicts the discrete risks, with several hidden layers, proves superior in the most complex situations, notably with non-proportional risks and crossing survival curves.

Keywords: Survival analysis, marker detection, neural networks, regularization, Screening, Cox model, AFT model, AH model, high-dimension, cancer.

Table des matières

Remerciements **i**

Résumé **v**

Abstract **ix**

Introduction **1**

1 Analyse de survie **11**
1.1 Contexte et objectifs 11
1.2 Modèles 13
1.2.1 Notations 13
1.2.2 Estimateurs non-paramétriques 15
1.2.3 Modèle de Cox 17
1.2.4 Modèle de vie accélérée (AFT) 20
1.2.5 Modèle des risques accélérés (AH) 21
1.3 Métriques 22
1.3.1 Indices de concordance 23
1.3.2 Score de Brier Intégré 26

2 Sélection de variables **29**
2.1 Sélection de variables pour la détection de marqueurs 29
2.1.1 Contexte de l'étude 29
2.1.2 Analyse différentielle 30
2.1.3 Recursive Feature Elimination 32
2.1.4 Résultats de l'étude 34
2.2 Sélection de variables pour la survie 37
2.2.1 Méthodes de régularisation 37
2.2.2 Méthodes de Screening 41
2.2.3 Applications 44

3 Étude des réseaux de neurones pour la prédiction de survie **57**
3.1 Introduction 57
3.2 Réseaux de neurones 58
3.2.1 Cox-nnet 60
3.2.2 Réseau de neurones à partir d'un modèle à temps discret (approche 1) 62
3.2.3 Réseau de neurones à partir d'un modèle à temps discret (approche 2) 63
3.3 Simulations 66

3.3.1 Génération de données . . . 66
3.3.2 Plan de simulations . . . 71
3.3.3 Comportement des données simulées . . . 78
3.4 Résultats sur les données simulées . . . 83
3.4.1 Configuration 1 . . . 84
3.4.2 Configuration 2 : effet de la censure . . . 94
3.4.3 Configuration 3 : effet de la sparsité . . . 96
3.4.4 Discussion des résultats sur les données simulées . . . 99
3.5 Illustration sur données réelles . . . 99
3.5.1 Jeu de données du cancer du rein à cellules claires . . . 99
3.5.2 Jeu de données du cancer du sein . . . 102

Conclusion et perspectives **105**

Annexes **111**

A Résultats détaillés de l'analyse différentielle **111**

B Présentation des indices de similarité **113**
B.1 Diversité β . . . 113
B.2 Indices de similarité en écologie . . . 113
B.3 Indice de Jaccard . . . 115

C Résultats des méthodes de sélection de variables pour la survie **117**
C.1 Méthodes de régularisation . . . 117
C.1.1 Sur l'ensemble des gènes . . . 117
C.2 Méthodes de *Screening* . . . 119
C.2.1 Sur l'ensemble des gènes . . . 119

D Présentation de l'outil Gimli **123**

E Vérifier l'hypothèse des risques proportionnels à partir des résidus de Schoenfeld **125**
E.1 Introduction . . . 125
E.2 Définition des résidus (Schoenfeld, 1982 ; Schoenfeld, 1980) . . . 125
E.3 Définition et amélioration par Grambsch et al. (1994) . . . 126
E.4 Limites . . . 128
E.5 Résultats du test des risques proportionnels pour les différentes simulations . . . 129
E.5.1 Résumé des résultats . . . 130
E.5.2 Détails des résultats pour la simulation Cox/Weibull . . . 130

F Résultats détaillés de la prédiction de la survie en grande dimension **137**
F.1 Comportement des données simulées . . . 137
F.1.1 Distribution des données simulées . . . 137
F.1.2 Courbes de survie des données simulées . . . 138

G Illustration de la performance des réseaux de neurones pour la prédiction de la survie sur des données du cancer du sein **141**

Références **143**

Table des figures

1 Les différentes étapes du RNA-seq 2

2.1 Heatmap de l'expression des gènes dans l'étude du ccRCC 34
2.2 Barplot des résultats d'analyse différentielle (DESeq2) dans l'étude du ccRCC 35
2.3 Résultats de la méthode RFE avec un modèle SVM-linéaire sur les IC différentiellement exprimés 36
2.4 Illustration de la pénalisation l_1 38

3.1 Schéma d'un réseau de neurones de type perceptron multi-couches (MLP) 60
3.2 Structure du réseau de neurones basé sur un modèle à temps discret proposée par Biganzoli et al. (1998) 62
3.3 Structure du réseau de neurones basé sur un modèle à temps discret avec une sortie multivariée 65
3.4 Distribution des temps de survie des données réelles du cancer du sein 73
3.5 Distribution des temps de survie simulés par un modèle Cox/Weibull 74
3.6 Distribution des temps de survie simulés par un modèle AFT/Log-normale 75
3.7 Distribution des temps de survie simulés par un modèle AH/Log-normale 76
3.8 Distribution des temps de survie simulés par un modèle AFT/Log-normale avec un ajout d'un terme $\phi_2(X)$ 77
3.9 Résidus standardisés pour 1000 variables 80
3.10 Courbes de survie de différents individus pour la simulation Cox/Weibull 82
3.11 Courbes de survie de différents individus pour la simulation AFT/Log-normale 82
3.12 Courbes de survie de différents individus pour la simulation AFT/Log-normale modifiée . 83
3.13 Courbes de survie de différents individus pour la simulation AH/Log-normale 84
3.14 Courbes de survie obtenues à partir des méthodes pour la simulation Cox/Weibull 87
3.15 Courbes de survie obtenues à partir des méthodes pour la simulation AFT/Log-normale . . 89
3.16 Courbes de survie obtenues à partir des méthodes pour la simulation AH/Log-normale . . 91
3.17 Courbes de survie obtenues à partir des méthodes pour la simulation AFT/Log-normale modifiée 93
3.18 Courbes de survie obtenues à partir des méthodes pour la simulation AFT/Log-normale censurée 95
3.19 Courbes de survie obtenues à partir des méthodes pour la simulation AFT/Log-normale sparse 98
3.20 Courbes de survie obtenues à partir des méthodes sur le jeu de données du cancer du rein à cellules claires (ccRCC/KIRC) 100
3.21 Résultats de l'ensemble des méthodes étudiées pour la prédiction de la survie en grande dimension sur les données du cancer du rein à cellules claires avec 2 sélections différentes . . 102
3.22 Courbes de survie obtenues à partir des méthodes sur le jeu de données du cancer du sein provenant du projet METABRIC 104

B.1 Schéma de la région R quand 2 sites sont considérés $S = 2$ 114

F.1 Distribution des temps de survie simulés par un modèle AFT/Log-normale censuré 137
F.2 Distribution des temps de survie simulés par un modèle AFT/Log-normale sparse 138
F.3 Courbes de survie des individus simulés par un modèle AFT/Log-normale censuré 139
F.4 Courbes de survie des individus simulés par un modèle AFT/Log-normale sparse 139

G.1 Courbes de survie obtenues à partir des méthodes sur le jeu de données du cancer du sein (Breast) 142

Liste des tableaux

2.1 Résultats de stabilité pour l'ensemble des méthodes de régularisation 46
2.2 Résultats des méthodes de régularisation sur les checkpoints 47
2.3 Résultats des méthodes de régularisation sur les gènes différentiellement exprimés 49
2.4 Résultats de stabilité pour l'ensemble des méthodes de *Screening* 50
2.5 Résultats des méthodes de *Screening* sur les checkpoints 51
2.6 Résultats des méthodes de *Screening* sur les gènes différentiellement exprimés 53

3.1 Définition des fonctions utilisées en analyse de survie pour les modèles Cox, AFT et AH . . 67
3.2 Caractéristiques des lois utilisées pour la simulation des données de survie 67
3.3 Expressions de la fonction de survie et des temps de survie pour les modèles de survie . . . 68
3.4 Résultats du test global de Grambsch et al. (1994) sur les données simulées avec 100 variables 81
3.5 Résultats pour l'ensemble des méthodes sur la simulation Cox/Weibull 85
3.6 Résultats pour l'ensemble des méthodes sur la simulation AFT/Log-normale 88
3.7 Résultats pour l'ensemble des méthodes sur la simulation AH/Log-normale 90
3.8 Résultats pour l'ensemble des méthodes sur la simulation modifiée AFT/Log-normale . . . 94
3.9 Résultats pour l'ensemble des méthodes sur la simulation modifiée AFT/Log-normale censurée 96
3.10 Résultats pour l'ensemble des méthodes sur la simulation modifiée AFT/Log-normale sparse 97
3.11 Résultats sur les données du cancer du rein à cellules claires 100
3.12 Résultats des méthodes sur les données du cancer du rein à cellules claire avec pré-sélection . 101
3.13 Résultats des différentes méthodes sur le jeu de données du cancer du sein (METABRIC) . 103

A.1 Résultats de DESeq2 concernant les niveaux d'expression des 44 ICs 112

B.1 Table schématisant l'utilisation de l'information pour les indices de similarité 114
B.2 Résultats de l'indice de Jaccard pour les méthodes de régularisation 116
B.3 Résultats de l'indice de Jaccard pour les méthodes de *Screening* 116

C.1 Résultats des méthodes de régularisation sur l'ensemble de gènes 119
C.2 Résultats des méthodes de *Screening* sur l'ensemble de gènes 121

E.1 Résultats du test global de Grambsch et al. (1994) sur les données simulées avec 10 variables 130
E.2 Résultats du test global de Grambsch et al. (1994) sur les données simulées avec 1000 variables 130
E.3 Résultats du test des risques proportionnels pour la simulation Cox/Weibull avec 10 variables 130
E.4 Résultats du test des risques proportionnels pour la simulation Cox/Weibull avec 100 variables 133
E.5 Résultats du test des risques proportionnels pour la simulation Cox/Weibull avec 1000 variables 135

G.1 Résultats des différentes méthodes sur des jeux de données du cancer du sein 142

Introduction

Contexte

Médecine de précision

La médecine de précision n'est pas un concept nouveau. Depuis des siècles, les praticiens adaptent le traitement donné au patient à partir de ses caractéristiques cliniques (âge, sexe, antécédents). Cependant, la découverte de la structure en double hélice de l'ADN dans les années 50 a été le point de départ de l'évolution de la médecine de précision. Cette médecine adapte maintenant le traitement aux caractéristiques moléculaires du patient et de sa maladie. La mutation de la médecine de précision a pu être réalisée grâce à l'apparition de différentes avancées technologiques. À la fin des années 70, une première méthode développée par Sanger et al. (1977) permet de séquencer l'ADN. À partir de cette date, les technologies ne cesseront d'évoluer. En effet, les puces à ADN (Schena et al., 1995) font leur apparition dans les années 90. Elles constituent la première technologie à grande échelle en biologie donnant accès au profil d'expression des gènes. Elles permettront le séquençage du génome humain quelques années plus tard (Venter et al., 2001; Lander et al., 2001). Dans les années 2000, une nouvelle technologie fait son apparition : la technique de séquençage d'ADN à haut-débit (Margulies et al., 2005). Le coût de cette technique a rapidement diminué en quelques années et celle-ci s'est largement répandue. La quantité d'informations moléculaires produite par ces technologies a amené de nouvelles problématiques et le besoin d'analyses statistiques et de traitements bioinformatiques adaptés est apparu (Searls, 2000).

La médecine de précision joue un rôle important en oncologie. Elle aide au diagnostic (type de cancers), au pronostic (survie, présence de métastases) et également à la prédiction de la réponse d'un patient à un traitement. Tout d'abord, il est possible de déterminer une prédisposition génétique à un cancer. Prenons l'exemple du cancer du sein, il a été établi que les personnes ayant une expression importante des gènes BRCA1/BRCA2 avaient plus de risques de le développer (Antoniou et al., 2002). Pour les patients porteurs de cette prédisposition, des examens de dépistage seront donc mis en place. Deuxièmement, il est possible de donner un diagnostic et/ou pronostic à des patients. En effet, l'absence ou la présence de mutations dans la cellule tumorale peut indiquer le type et l'agressivité de la tumeur. De plus, savoir si une mutation est présente dans une cellule tumorale peut permettre de prédire l'efficacité d'un traitement. Par exemple, il a été montré que 15% des patients atteints du cancer du sein avec une sur-expression de HER2 ne répondaient pas aux traitements classiques (Arteaga et al., 2012). Cela nous amène à notre dernier point concernant la tolérance aux médicaments. En effet, les thérapies proposées pour combattre le cancer sont souvent lourdes avec des effets secondaires importants. Des thérapies ciblant les récepteurs HER2 ont été mises en place permettant aux patients de gagner en confort de vie. Enfin, il est également possible que certains traitements occasionnent une toxicité chez le patient provoquée par une interaction entre la molécule utilisée dans le traitement et la quantité d'enzyme produite par le patient. À titre d'illustration, certains patients avec un déficit de l'enzyme dihydropyrimidine déshydrogénase (DPD) ont été intoxiqués avec le traitement 5-Fluoro-Uracile (5-FU) (Gamelin et al., 2014). Un test existe pour dépister la présence de l'enzyme chez le

patient. Connaître l'expression des gènes des cellules germinales (patrimoine génétique hérité de nos parents) et/ou des cellules tumorales (code génétique modifié) est important pour aider les médecins à personnaliser la prise en charge de leurs patients. C'est dans ce contexte que le développement des outils bioinfomatiques et statistiques adaptés est devenu un enjeu majeur pour mieux comprendre la biologie moléculaire.

Détection de marqueurs à partir des données d'expression

Données d'expression

Différentes techniques permettent d'obtenir des données d'expression de gènes afin d'étudier le transcriptome d'une cellule ou d'un tissu donnés. La première technique est celle des puces à ADN qui est aussi appelée *microarrays* (Southern et al., 1999). Elle est basée sur le principe d'hybridation. Les cibles, correspondant aux ARNm (ARN messagers) que l'on cherche à identifier, sont transformées en ADNc (ADN complémentaires) et sont marqués par fluorescence. Ces derniers sont mis en contact avec la puce à ADN qui portent l'ensemble des sondes à sa surface. Les sondes sont les fragments d'ADN synthétique représentant les gènes dont on cherche à quantifier l'expression. Chaque brin d'ADNc va donc s'hybrider aux sondes qui lui sont complémentaires pour reformer le double brin. Le niveau d'hybridation est alors calculé par fluorescence et une valeur d'intensité pour chaque sonde est donc calculée. La mesure acquise par cette technologie est continue. La seconde technologie, appelée RNA-seq, est apparue quelques années plus tard et contrairement aux puces à ADN, n'est pas basée sur l'hybridation de l'ADNc mais sur son séquençage. Les différentes étapes pour l'acquisition des données d'expression à partir de la technologie RNA-seq sont résumées sur la Figure 1 et les données obtenues sont discrètes.

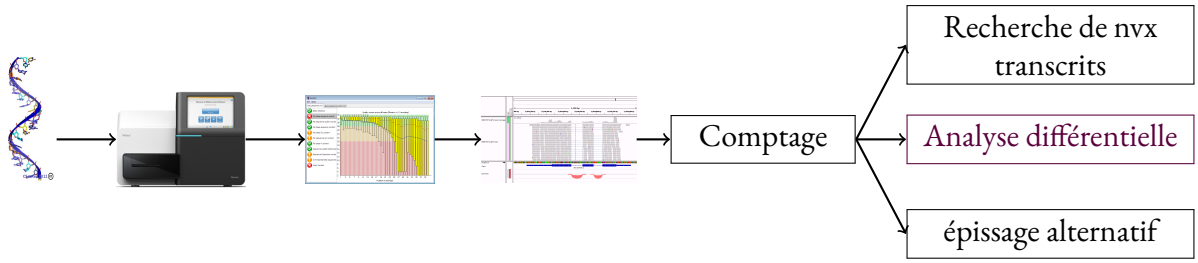


Figure 1 – Les différentes étapes du RNA-seq : L'ARN est retranscrit en ADNc. Cet ADNc est ensuite séquencé. À la suite de ce séquençage, on obtient des lecture de brins d'ADNc, appelées *reads*. Ensuite, on contrôle la qualité des *reads*. Ces *reads* vont être alignés sur le génome de référence et on va compter le nombre de *reads* alignés sur chaque gène. À la suite de cette expérience de RNA-seq, plusieurs objectifs peuvent être effectués comme la recherche de nouveaux transcrits, l'analyse différentielle ou l'épissage alternatif.

Plusieurs objectifs peuvent être réalisés à partir de la technologie RNA-seq comme la recherche de nouveaux transcrits, l'épissage alternatif ou encore l'analyse différentielle. Dans cette thèse, nous nous sommes intéressés aux données issues soit des puces à ADN soit du RNA-seq, et nous les avons utilisé pour l'analyse différentielle ou comme entrées de modèles, notamment de survie.

Analyse différentielle

L'analyse d'expression différentielle consiste en la recherche de gènes différentiellement exprimés, c'est-à-dire des gènes avec un niveau d'expression significativement différent entre deux conditions. De nombreuses méthodes d'analyse différentielle ont été développées pour la comparaison de deux conditions, initialement

proposées pour les puces à ADN, puis mises à jour pour l'analyse de données RNA-Seq. Ces méthodes utilisent principalement des tests statistiques pour la recherche de gènes différentiellement exprimés. Les données RNA-seq demandent un soin particulier avant de faire l'analyse différentielle. En effet, elles amènent de nombreuses problématiques comme la normalisation et le choix de la modélisation des données. Tout d'abord, il est nécessaire de prendre en compte la profondeur de séquençage par échantillon. Le nombre de *reads* alignés pour un gène et un échantillon donné dépend du nombre de total de fragments d'ADN alignés pour l'échantillon. Pour que l'expression d'un gène soit comparable entre plusieurs échantillons, il convient de prendre en compte le nombre total de *reads* alignés pour chaque échantillon, aussi appelé profondeur de séquençage par échantillon. Law et al. (2014) a utilisé le log-comptage par million (*log-count per million log-cpm*) du gène donné pour l'échantillon et cela correspond au logarithme du nombre de *reads* alignés pour le gène donné dans l'échantillon considéré normalisé par la profondeur de séquençage exprimé en million. Il est également nécessaire de prendre en compte la longueur du gène. En effet, la longueur d'un gène peut varier de 400 à 2 millions de paires de base. La méthode de normalisation la plus simple pour la comparaison de mesure d'expression entre deux gènes consiste à diviser chaque comptage par la longueur du gène correspondante. De plus, les données RNA-seq ont été, dans un premier temps, modélisées par une loi de Poisson. Cependant, cette loi modélise mal la grande variabilité inter-échantillons présente dans les données RNA-seq. Pour prendre en compte cette variabilité, des modèles basés sur la loi de Poissson surdispersée ou sur la loi Binomiale Négative ont été proposés comme solution alternative (Anders et al., 2010; McCarthy et al., 2012). Les paramètres de la loi doivent prendre en compte les biais techniques de la technologie RNA-seq détaillés précédemment. Deux approches sont donc possibles pour modéliser les données RNA-seq. La première utilise des modèles discrets (loi de Poisson ou loi Binomiale Négative) modélisant directement les comptages et prenant en compte la normalisation dans l'écriture des paramètres du modèle. La seconde approche utilise des modèles gaussiens modélisant les comptages normalisés. Il existe beaucoup de méthodes différentes issues des deux approches pour l'analyse de l'expression différentielle des données RNA-Seq, et les articles de comparaison de méthodes (Soneson et al., 2013; Robles et al., 2012; Rapaport et al., 2013) ne réussissent pas à déterminer une méthode performante dans tous les cas. Une des méthodes la plus utilisée et avec les meilleures performances est DESeq2 (Anders et al., 2010). Cette méthode est basée sur la seconde approche, c'est-à-dire qu'elle utilise un modèle Binomial Négatif et la normalisation des données est prise en compte dans l'écriture du modèle. Enfin, l'analyse différentielle consiste à tester si le niveau d'expression de chaque gène diffère entre deux conditions. Le test d'hypothèse est donc effectué pour chaque gène et puisque le nombre de gènes est élevé, il convient de contrôler correctement le nombre de gènes détectés différentiellement exprimés à tort, aussi appelé le nombre de faux positifs (*False Discovery Rate (FDR)*). DESeq2 (Anders et al., 2010) utilise un test de Wald pour trouver les gènes différentiellement exprimés entre les deux conditions et une procédure de Benjamini-Hocherg (Benjamini et al., 1995) est utilisée pour prendre en compte le nombre de faux positifs.

Autres modèles pour la détection de marqueurs à partir des données d'expression

La question de la modélisation des données d'expression est majeure dans le but de détecter des marqueurs. Plusieurs travaux ont été proposés. Les modèles de mélange ont été développés avec une loi de Poisson par Rau et al. (2015) et pour une loi Binomiale Négative par Si et al. (2014) pour détecter les gènes co-exprimés. Cela est utile pour mettre en évidence par exemple les gènes qui partagent des fonctions similaires. Pour répondre à ce même objectif, quelques travaux sur l'inférence des réseaux ont été proposés mais la plupart se base sur l'inférence de réseaux gaussiens en transformant les données préalablement. Dernièrement, Gallopin (2015) a proposé un modèle Poisson Log-normal hiérarchique pour l'inférence des réseaux. Mais ces méthodes d'inférence de réseaux souffrent de la faible taille d'échantillons. D'autres types de méthodes ont également été utilisés dans le contexte de la détection de marqueurs dont leur principal avantage est leur facilité à gérer des données avec une taille d'échantillon faible. Ce sont les méthodes d'apprentissage

statistique. Les forêts aléatoires (*Random Forest* RF) (Breiman, 2001) ou les machines à vecteurs supports (*Support Vector Machine* SVM) (Vapnik, 1998) font notamment parties des méthodes d'apprentissage. Pour les forêts aléatoires, Boulesteix et al. a réalisé en Boulesteix et al. (2012) une synthèse de l'utilisation des forêts aléatoires en bioinformatique. Dernièrement, une revue de Huang et al. (2018) a montré les atouts de l'application des SVM dans l'étude génomique du cancer. Ces deux revues montrent la possibilité d'utiliser ces méthodes pour faire de la détection de marqueurs, c'est-à-dire obtenir les variables importantes pour la classification (malade/sain). Huang et al. (2018) évoque l'exemple de SVM *recursive feature elimination* (SVM-RFE) (Guyon et al., 2002; Guyon et al., 2003). L'idée de la méthode SVM-RFE est d'entraîner un SVM sur les données et d'appliquer un critère pour évaluer l'importance de chaque variable. Les variables considérées comme moins importantes selon le critère sont supprimées. Enfin, ces méthodes peuvent également être utilisées dans le but de trouver les interactions gène-gène.

Analyse de survie en grande dimension

La partie précédente présente la détection de marqueurs pour caractériser le cancer à partir de données d'expression, mais la détection de marqueurs peut également permettre de répondre à d'autres questions biologiques. Nous pouvons également faire de la sélection de variables à partir des modèles de survie et dans ce cadre nous répondons à quels sont les marqueurs influençant la survie des patients.

Analyse de survie

Dans le contexte de la médecine de précision, l'analyse de survie est un des cadres de travail classique permettant de relier la durée de survie d'un individu à des variables explicatives. La durée de survie est définie comme le temps écoulé jusqu'à la survenue d'un événement d'intérêt. Cet événement peut être le décès de l'individu mais il peut également correspondre à la rémission ou à la rechute de l'individu. Le modèle classiquement utilisé en analyse de survie est le modèle de Cox (Cox, 1972). Celui-ci permet de relier la survie des patients à ses variables explicatives. Dans le domaine de l'analyse de survie, deux grandes questions sont possibles :

1. Quelles sont les variables explicatives qui permettent d'expliquer la survie des patients ?
2. Est-il possible de prédire la durée de survie d'un individu à partir de ses variables explicatives ?

Pour connaître l'influence des variables sur la survie des patients, il est nécessaire d'estimer les paramètres du modèle considéré. Dans le modèle de Cox, la durée de survie T_i d'un individu i est reliée à ses variables $X_{i.}$ à partir du risque instantané de décès conditionnel qui s'exprime de la façon suivante :

$$\lambda(t|X_{i.}) = \alpha_0(t)\exp(\beta^T X_{i.})$$

avec $\alpha_0(t)$ le risque de base et β les paramètres du modèle traduisant l'importance des variables. Pour répondre à la première question, nous avons besoin d'estimer les paramètres β du modèle. Cette estimation est réalisée à l'aide de la vraisemblance partielle de Cox (Cox, 1975) qui est une partie de la vraisemblance totale. Mais pour répondre à la seconde question, il est nécessaire d'estimer le risque instantané en entier car la fonction de survie s'exprime à partir du risque instantané de décès conditionnel. Le risque de base $\alpha_0(t)$ doit donc être estimé et cela est réalisé lors d'une seconde étape avec un estimateur non-paramétrique (par exemple l'estimateur à noyau de Ramlau-Hansen (1983)) dans lequel on plugge l'estimateur obtenu à la première étape pour β.

Grande dimension

Comme je l'ai évoqué ci-dessus, de nombreuses technologies sont apparues dans les années 90 permettant l'accès à un grand nombre de données moléculaires d'un individu. Cet afflux de nouvelles données a toutefois

amené de nouvelles problématiques. Celles-ci ont poussé les statisticiens et les bioinformaticiens à adapter et à développer de nouvelles méthodes afin de traiter et d'extraire la connaissance de ces données. Une des problématiques importantes dans la recherche clinique est la faible quantité d'échantillons comparée à celle des variables moléculaires et cliniques auxquelles les chercheurs/médecins ont accès. En effet, il est difficile de recruter des milliers de patients pour une étude clinique. Les statisticiens récupèrent donc des études avec une centaine de patients et des milliers voire des dizaines de milliers ou même des millions dans le cas des k-mers (Audoux et al., 2017; Marchet et al., 2020) de variables moléculaires pour chaque patient à partir desquelles ils doivent réussir à extraire de l'information. Le statisticien est donc dans un nouveau cadre : la grande dimension. On parle de grande dimension quand le nombre de variables est largement supérieur à la taille de l'échantillon. Dans ce nouveau cadre, les méthodes statistiques classiques utilisées jusqu'à présent ne peuvent plus être appliquées et le développement de nouvelles méthodes est nécessaire.

Détection de marqueurs

Une des premières solutions proposée par les statisticiens concerne la réduction de la dimension. Cette réduction va permettre de représenter les données dans un espace plus facilement interprétable par les méthodes classiques. Dans cette thèse, nous considérons comme réduction de dimension seulement celle obtenue par sélection de variables, notre objectif étant de mettre en évidence les variables les plus pertinentes. Les méthodes de réduction de dimension par représentation des données initiales dans un espace latent (apprentissage de représentation) (Bengio et al., 2013) ne seront pas considérées. Plusieurs approches de sélection de variables ont été mises en place pour réduire la dimension. La première approche correspond à celle des modèles pénalisés. Ces modèles ont, dans un premier temps, été développés dans le cadre des modèles linéaires et adaptés par la suite au cadre de l'analyse de survie. Il s'agit d'estimer les paramètres avec une régression pénalisée, c'est-à-dire qu'un terme de pénalisation est ajouté lors de l'estimation des paramètres. Ce terme de pénalisation peut par exemple rendre nul tous les paramètres des variables non pertinentes. Plusieurs modèles pénalisés existent. La régression pénalisée la plus connue est la procédure Lasso, développée par Tibshirani (1996). Il pénalise son critére d'estimation (les moindres carrés dans le cas de la régression linéaire) avec une norme L1 ce qui permet d'annuler un certain nombre de paramètres du modèle et de mettre ainsi en évidence les variables pertinentes. Un autre modèle pénalisé existe, la pénalité ridge proposée par Hoerl et al. en 1970. La pénalisation utilisée est la norme L2. Celle-ci ne va pas mettre à zéro les paramètres des variables non informatives, mais va les réduire et les faire tendre vers zéro. Une autre procédure de pénalisation est l'Elastic-net (Zou et al., 2005) développée par Zou et al. en 2005. Le concept de cette procédure pénalisée est de garder l'avantage de la parcimonie amenée avec la norme L1 de la pénalisation Lasso couplée avec une pénalisation ridge. La pénalisation utilisée combine à la fois les normes L1 et L2. Ces trois modèles pénalisés sont les plus connus et utilisés, mais d'autres existent, comme par exemple, les modèles pénalisés Fused-Lasso (Tibshirani et al., 2005), Adaptive-Lasso (Zou, 2006) ou Group-Lasso (Yuan et al., 2006). La pénalisation fused-Lasso pénalise à la fois les paramètres et leurs différences successives. Le principe de cette pénalisation est de réduire le nombre de variations entres des variables qui se suivent. La pénalisation Adaptive-Lasso (Zou, 2006) pénalise en pondérant la pénalisation des paramètres. Cette pénalisation a pour objectif de résoudre l'instabilité du Lasso en gérant des variables qui ont un effet sur la variable d'intérêt et qui sont fortement corrélées entre elles. Enfin, la pénalisation Group-Lasso pénalise sur un groupe de paramètres défini en amont. Cette pénalisation est utile quand une structure de groupe est déjà connue. Dans cette configuration, le Lasso (Tibshirani, 1996) risque d'exclure un certain nombre de variables et d'en conserver d'autres d'un même groupe alors qu'on souhaiterait plutôt garder soit toutes les variables soit les exclure toutes, ce que permet de faire le Group-Lasso. Ces différentes types de pénalisations ont par la suite été étendus au cadre de l'analyse de survie (Tibshirani, 1997; Verweij et al., 1994; Zhang et al., 2007; Wu, 2012).

Cependant, en grande dimension des problèmes d'instabilité ont été mis en évidence pour la régularisation par Fan et al. (2010a). Des méthodes appelées *Screening* ont donc été développées pour les modèles linéaires et adaptées par la suite au cadre des modèles de survie en utilisant la log-vraisemblance partielle de Cox. La première méthode est appelée *Sure Independance Screening* (SIS) (Fan et al., 2008). Le principe de cette méthode peut se résumer en deux étapes. La première étape consiste à calculer un score pour chaque variable. Les variables sont ensuite sélectionnées selon leur classement et ce classement est obtenu à partir de la valeur de leur score. La seconde étape consiste à appliquer la procédure Lasso sur les variables sélectionnées lors de la première étape. D'autres méthodes ont par la suite été développées comme ISIS, PSIS et coxCS. La méthode ISIS *(Iteratice Sure Independence Screening)* (Fan et al., 2010a) développée par Fan et al. (2010a) est une version itérative de la méthode SIS. La première étape d'ISIS consiste à appliquer la procédure SIS sur l'ensemble des variables. Ensuite, la procédure SIS est de nouveau exécutée pour chaque variable non-sélectionnée conditionnellement aux variables sélectionnées. Cette étape est répétée jusqu'à convergence. La méthode PSIS *(Principle Sure Independence Screening)* (Zhao et al., 2012) diffère de la méthode SIS dans le calcul du score dans la première étape. Enfin, la méthode CoxCS (Barut et al., 2016) utilise la connaissance biologique pour faire une première sélection des variables. La seconde étape consiste à appliquer une procédure SIS conditionnellement aux variables sélectionnées à la première étape.

Prédiction de la survie à l'aide des réseaux de neurones

Enfin, les statisticiens s'intéressent depuis ces dernières années à un type de méthodes appartenant à l'apprentissage automatique : les réseaux de neurones. Les réseaux de neurones sont des méthodes développées dans les années 50 et qui ont explosé ces dernières années en raison notamment de l'évolution des capacités de calculs. Les réseaux de neurones sont inspirés du fonctionnement des neurones biologiques. L'idée de cette démarche était de reproduire les capacités du cerveau comme l'apprentissage, la mémorisation de l'information et le traitement d'informations incomplètes (McCulloch et al., 1943). Le premier système artificiel capable d'apprendre par expérience, appelé perceptron (Rosenblatt, 1958), a été développé par Rosenblatt en 1958. La publication de Minsky et al. en 1969 a provoqué une désillusion dans le domaine de la recherche neuronal (Minsky et al., 1969). En effet, Minsky et al. (1969) a montré certaines limites de ces méthodes comme l'impossibilité de traiter des problèmes non-linéaires et de connexité. Après des années de disgrâce, une nouvelle génération de réseaux de neurones apparaît s'affranchissant des limites évoquées par Minsky et al. (1969). Ce nouveau type de réseaux de neurones est appelé perceptron multi-couches (Rumelhart et al., 1986 ; Lecun, 1985). Son succès peut s'expliquer par la flexibilité des modèles et sa méthode d'apprentissage efficace : la rétropropagation du gradient de l'erreur entre les couches. La popularité des réseaux de neurones n'a pas cessé d'augmenter depuis la fin des années 80 et les données considérées sont de plus en plus complexes, elles vont des données temporelles aux structures de données comme les graphes ou les fonctions. Par exemple, Cottrell et al. (2012) montre la flexibilité des réseaux de neurones à s'adapter à ces nouvelles données. De plus, nous les trouvons dans tous les domaines applicatifs : physique, finance, santé... Le domaine biomédical n'est pas en reste. L'une des applications des réseaux de neurones dans le domaine biomédical est l'imagerie qui a montré de nombreux résultats. En analyse de survie, des réseaux de neurones ont été développés dans les années 90 pour prédire la survie. Cependant, ces réseaux de neurones ont seulement été appliqués sur des données cliniques. Le comportement de la plupart des réseaux de neurones n'a jamais été étudié dans le cadre de la grande dimension jusqu'à présent. La capacité de calculs dans les années 90 ne le permettait pas. À noter que d'autres méthodes à apprentissage statistique ont été adaptées à la survie, notamment les *Random Survival Forests* (Ishwaran et al., 2008), mais n'ont pas été étudiées dans cette thèse.

Problématiques

Cette thèse est consacrée à l'identification des marqueurs biologiques et à la prédiction de la durée de survie en oncologie dans le contexte de la grande dimension. La grande dimension est devenue la problématique majeure chez les statisticiens qui tentent d'extraire de la connaissance des nouvelles données biologiques.

1. Le premier objectif de cette thèse est de mettre en évidence les gènes jouant un rôle dans l'évolution du cancer du rein à cellules claires (ccRCC). La grande dimension est devenue cruciale dans la détection de marqueurs biologiques en oncologie et a donc amené au développement de nouvelles méthodes statistiques depuis les 20 dernières années. Tout d'abord, les méthodes les plus couramment utilisées en analyse différentielle sont basées sur les tests statistiques. Cependant, de nombreux faux positifs sont détectés à cause du grand nombre de gènes. C'est pour cette raison que d'autres méthodes ont été développées mais des précautions particulières doivent être prises en compte dans ces méthodes pour modéliser au mieux les données d'expression RNA-seq. Nous étudierons dans cette thèse les méthodes d'apprentissage couplées avec celles d'analyse différentielle pour évaluer l'amélioration possible des performances de celles-ci.

   Dans le cadre de la détection de marqueurs pour la survie, de nombreuses méthodes ont été développées comme les méthodes de régularisation et les méthodes de *Screening*. Des problèmes d'instabilité ont été montrés pour ces méthodes en grande dimension, mais cela a été peu étudié et quantifié. Nous allons comparer ces différentes méthodes dans le cadre de la survie sur les données réelles du cancer du rein à cellules claires.

2. Le second objectif de cette thèse est d'étudier des méthodes basées sur des réseaux de neurones pour prédire la survie des patients en oncologie. La grande dimension joue aussi un rôle majeur dans la prédiction de la survie car celle-ci ne se fait plus seulement sur les variables cliniques du patient, mais également sur ses variables génomiques. La méthode la plus connue avant l'arrivée des données génomiques était le modèle de Cox pour traiter des données de survie. En grande dimension, la maximisation de la log-vraisemblance partielle de Cox ne fonctionne plus. Les méthodes de régularisation ont donc été proposées pour réduire la dimension, mais elles amènent également d'autres problématiques comme l'instabilité. Les réseaux de neurones sont devenus des outils très populaires dans de nombreux domaines de recherche et récemment en analyse de survie. Cependant, le comportement et le potentiel de ceux-ci ont été peu étudiés en grande dimension. Nous proposons de comparer les performances de certains modèles de survie basés sur des réseaux de neurones dans ce manuscrit.

Contributions

Les contributions réalisées dans cette thèse répondent aux deux problématiques exposées ci-dessus :

1. Pour répondre à la première problématique, notre travail va avoir deux directions. La première est axée sur le compréhension des processus biologiques. L'équipe de l'hôpital Saint-Louis voulait connaître les processus immunitaires impliqués dans le cas du cancer du rein à cellules claires. Dans ce but, nous avons commencé par faire une analyse différentielle sur les données d'expression de gènes de ce cancer. Cette partie a permis de mettre en évidence des gènes impactant le système immunitaire. Notre deuxième contribution a été plus originale, elle permet de souligner l'importance des variables sélectionnées. Cette démarche a consisté à utiliser une méthode d'apprentissage SVM couplée avec l'algorithme de sélection de variables RFE (Guyon et al., 2002; Guyon et al., 2003), sur des gènes pré-sélectionnés. La pré-sélection a été obtenue par analyse différentielle. Cette étude a permis de mettre en évidence les gènes impactant le cancer du rein à cellules claires et a donné lieu à une publication en collaboration avec l'équipe de l'hôpital Saint-Louis (Tronik-Le Roux et al., 2020).

La deuxième direction de notre travail concerne la détection de marqueurs impactant la durée de survie des patients. Pour étudier la survie des patients, le modèle le plus utilisé est le modèle de Cox. En grande dimension, la procédure classique d'estimation consistant à maximiser la log-vraisemblance partielle de Cox (qui est une partie de la log-vraisemblance totale où seul les coefficients de régression du modèle apparaissent) ne fonctionne plus. C'est pour cette raison que d'autres méthodes ont été développées dont notamment les méthodes de régularisation et de *Screening*. La deuxième direction de ce travail se divise donc en deux parties. La première concerne une étude des gènes impactant la survie des patients dans le cadre du cancer du rein à cellules claires. Pour cela, nous avons utilisé les méthodes de régularisation et de *Screening*. Une des méthodes de *Screening* utilisée est basée sur la connaissance biologique. Nous avons donc proposé une démarche consistant à utiliser une méthode d'apprentissage supervisé pour la reconnaissance de termes biomédicaux à partir d'*abstracts*. Les gènes mis en évidence par cette méthode ont été utilisés comme connaissance biologique. Les méthodes appliquées sur le cancer du rein à cellules claires a permis de mettre en évidence le potentiel de certains gènes pouvant expliquer la survie des patients. Cependant, ces méthodes sont réputées pour être instables lorsque la dimension des données augmente. La seconde partie de cette direction concerne donc l'étude de stabilité des méthodes de régularisation et de *Screening* sur notre jeu de données. Pour mesurer le niveau de stabilité de la méthode, nous avons exécuté un certain nombre de fois les méthodes sur le même jeu de données. Nous avons également fait varier la dimension de l'ensemble de gènes donné en entrées afin de voir l'impact de celle-ci sur la stabilité des méthodes. Enfin, nous avons proposé d'utiliser un indice de similarité qui permet de mesurer la variation de la composition des gènes sélectionnés entre les différentes exécutions. C'est un indice couramment utilisé en écologie (Baselga, 2013 ; Baselga et al., 2012 ; Arita et al., 2008 ; Arita, 2017) mais qui a été peu appliqué à d'autres domaines et en particulier en sélection de variables.

2. Pour répondre à la seconde problématique, notre contribution a été d'étudier le potentiel des réseaux de neurones pour prédire la survie en grande dimension. Deux approches différentes existent pour les réseaux de neurones en analyse de survie. La première est basée sur le modèle de Cox et la seconde est basée sur un modèle à temps discret. La seconde approche est moins courante et n'avait pas été étudiée en grande dimension. Par exemple, Biganzoli et al. (1998) et Lee et al. (2018) ont proposé respectivement un réseau de neurones perceptron à multi-couches et un réseau de neurones multi-tâche basés sur cette approche, mais il l'ont seulement étudié à partir de variables cliniques. Nous nous sommes plus particulièrement concentrés sur la seconde approche basée sur un modèle à temps discret en étudiant le réseau de neurones de Biganzoli et al. (1998) (appelé NNsurv) et en l'adaptant à la grande dimension. Nous avons proposé un nouveau réseau de neurones en modifiant ce dernier. Tout d'abord, nous avons considéré un réseau de neurones avec une architecture plus profonde en y ajoutant une couche cachée, celui-ci est appelé NNsurv deep. Ensuite, nous avons développé un nouveau réseau de neurones (appelé NNsuvrK) et il diffère de NNsurv par la configuration de sa sortie (plusieurs sorties au lieu d'une seule) ainsi que par la pénalisation (fused-Lasso) et l'indicateur de censure (basé sur l'estimateur de Kaplan-Meier) utilisés. Afin d'étudier le potentiel des réseaux de neurones pour la prédiction de la survie en grande dimension, nous avons comparé les réseaux de neurones basés sur un modèle à temps discret que nous avons adapté et développé pour la grande dimension (NNsurv, NNsurv deep et NNsurvK) avec le réseau de neurones Cox-nnet (Ching et al., 2018). Cox-nnet est un réseau de neurones basé sur la log-vraisemblance partielle de Cox et adapté à la grande dimension. La procédure Lasso utilisant la log-vraisemblance partielle de Cox est prise comme référence. Cette comparaison de méthodes a été effectuée à partir d'un plan de simulation. La génération des données a été réalisée à partir de différents modèles de survie (Cox, AFT et AH) afin d'avoir des données de survie avec différents niveaux de complexité notamment pour tester l'effet des risques non proportionels et les cas où les courbes de survie individuelle se croisent. Nous avons

fait varier un certain nombre de paramètres comme la taille de l'échantillon et le nombre total de variables. Nous avons également fait varier les paramètres des données de survie pour prendre en compte l'effet de la censure et la sparsité. Enfin, nous avons également illustré cette comparaison sur des jeux de données réelles.

Organisation du manuscrit

Le manuscrit est divisé en trois chapitres :

1. Le chapitre 1 est une présentation de l'analyse de survie. Nous y exposons les objectifs de l'analyse de survie et nous introduisons par la suite les notations utilisées. Pour chacun des objectifs, nous présentons les méthodes couramment utilisées en analyse de survie. Enfin, nous introduisons les métriques utilisées dans ce manuscrit.
2. Le chapitre 2 se divise en deux parties. La première concerne la détection de marqueurs en oncologie expliquant les processus immunitaires pour le cancer du rein à cellules claires. Nous commençons donc par introduire le contexte de l'étude, nous présentons ensuite les deux démarches utilisées pour répondre à ce premier objectif et nous terminons en détaillant les résultats de l'étude. La seconde partie concerne la détection de marqueurs impactant la survie des patients pour le cancer du rein à cellules claires. Nous présentons tout d'abord les deux types de méthodes utilisés et nous montrons les résultats de ces méthodes sur les données réelles.
3. Le chapitre 3 concerne l'étude des réseaux de neurones pour prédire la durée de survie en grande dimension. Nous débutons ce chapitre en introduisant le contexte de l'étude. Nous présentons ensuite les différents réseaux de neurones étudiés et développés pour prédire la survie des patients. Ensuite, nous exposons dans ce chapitre le plan de simulation mis en œuvre pour comparer les réseaux de neurones développés en analyse de survie pour la grande dimension. Nous présentons les résultats de ces modèles sur les données générées à partir du plan de simulation. Enfin, nous illustrons les performances des réseaux de neurones sur des jeux de données réelles.

Nous terminons le manuscrit par une discussion et en détaillant quelques perspectives de notre travail.

Chapitre 1

Analyse de survie

1.1 Contexte et objectifs

L'analyse de survie est l'étude du temps écoulé jusqu'à l'apparition d'un événement d'intérêt. Cet événement est appelé *décès* mais il ne correspond pas forcément à la mort de l'individu. Il peut s'agir d'une rechute ou d'une rémission de l'individu. En analyse de survie, deux objectifs sont étudiés. Le premier concerne la détection de facteurs influençant la durée de survie. Le second concerne la prédiction de la durée de survie pour un individu donné à partir des facteurs auxquels il est soumis. Ces facteurs, appelés variables explicatives ou covariables, peuvent être par exemple des données cliniques (âge, sexe,...) et/ou des données génomiques issues du séquençage à haut-débit. Une particularité des données de survie est qu'elles ne sont pas toujours observées. Il est possible que lors d'une étude, des individus sortent de celle-ci pour différentes raisons. Un déménagement, le décès de l'individu (qui n'est pas la cause de la maladie étudiée) peuvent amener à la disparition de l'individu dans l'étude. Ces données sont dites censurées. Plusieurs types de censures existent. La censure à laquelle nous sommes confrontés à partir de nos données est la censure à droite. On parle de censure à droite quand la durée de survie observée est plus petite que la durée de survie réelle de l'individu. Cette particularité demande donc un traitement particulier dans l'étude de la durée de survie. Considérons un modèle simple de régression de survie sous la forme :

$$Y_i \sim \mathbb{P}(y|\beta^T X_{i.}), \tag{1.1}$$

avec $X_{i.} = (X_{i1}, \ldots, X_{ip})^T$ l'ensemble des variables de l'individu i que l'on suppose standardisées et $\beta = (\beta_1, \ldots, \beta_p)^T$ les paramètres à estimer. Le paramètre β traduit les poids des variables sur la durée de survie. Après la définition du modèle de survie, trois termes sont importants pour l'analyse de survie : l'estimation, la sélection de variables et la prédiction de la durée de survie. Tout d'abord, la procédure classique d'estimation du paramètre β consiste à minimiser l'opposé de la log-vraisemblance $\mathcal{L}$:

$$\arg\min_{\beta} \{-\mathcal{L}(\beta)\}. \tag{1.2}$$

À partir de l'estimation des paramètres β, deux objectifs sont possibles : la sélection de variables et/ou la prédiction. La sélection de variables consiste à déterminer les variables impactant le plus la durée de survie, c'est-à-dire à sélectionner les variables dont les coefficients des paramètres estimés sont les plus grands. La prédiction consiste à estimer la durée de survie des individus à partir des variables auxquelles ils sont soumis, elle est obtenue en calculant l'équation (1.2) en y insérant les paramètres estimés et les variables de l'individu. En analyse de survie, les données génomiques sont de plus en plus utilisées. Le nombre de variables est de

plus en plus important, mais la taille des cohortes augmente peu comparativement. Nous sommes dans un cadre de grande dimension, car le nombre de variables est supérieur à la taille de l'échantillon. Dans ce cadre, la procédure classique d'estimation évoquée ci-dessus n'est plus valide et l'estimateur $\widehat{\beta}$ n'est plus consistant. Cela engendre donc de nouvelles problématiques pour les deux objectifs que sont la sélection de variables et la prédiction. Quand le nombre de variables est grand comparé à la taille de l'échantillon, il est intéressant de faire de la sélection de variables. En effet, cela revient à sélectionner les coefficients non-nuls dans β. Mais minimiser l'opposé de log-vraisemblance amènera toujours à choisir le plus grand modèle, autrement dit à choisir un estimateur dont tous les coefficients sont non-nuls. On peut parfois mettre en oeuvre des tests statistiques sur la nullité des coefficients, mais le grand nombre de variables et donc de paramètres rend la combinatoire des tests à réaliser insurmontable. Cet estimateur ne peut pas être défini en grande dimension et serait difficilement interprétable. Pour résoudre cela, une solution classique consiste à pénaliser le critère d'estimation, ce qui signifie minimiser l'opposé de la log-vraisemblance à laquelle un terme de pénalité est ajouté. L'idée intuitive derrière ce critère pénalisé d'estimation est l'incitation à choisir un modèle plus petit :

$$\widehat{\beta} = \arg\min \left\{ -\mathcal{L}(\beta) + pen(\beta) \right\}. \tag{1.3}$$

De nombreuses pénalités existent et donnent des estimateurs avec différentes propriétés d'interprétabilité, elles sont détaillées en Section 2.2.1 :

$$\begin{aligned} pen(\beta) &= \lambda ||\beta||_p \\ &= \lambda \left(\sum_{j=1}^{p} |\beta_j|^p \right)^{\frac{1}{p}}, \end{aligned} \tag{1.4}$$

où $0 \leq p \leq +\infty$ et $\lambda \in \mathbb{R}^+$ est un hyperparamètre de régularisation contrôlant l'ajustement entre l'adéquation du modèle aux données et sa complexité. Nous renvoyons à Bickel et al. (2006) pour plus de détails sur le concept de pénalisation. Comme pour la sélection de variables, la prédiction devient très compliquée quand le nombre de variables est grand comparé à la taille de l'échantillon. En effet, l'estimateur β n'est pas défini en grande dimension. Pour prédire la durée de survie, celui-ci doit être défini. La solution consiste également à utiliser le critère d'estimation pénalisée présenté ci-dessus pour l'estimation des paramètres β.

Dans cette thèse, nous nous intéressons aux deux objectifs de l'analyse de survie : la sélection de variables et la prédiction. Nous rappelons dans les deux paragraphes suivants les détails des deux objectifs en évoquant les méthodes utilisées et leurs importances en oncologie.

Le premier objectif de l'analyse de survie consiste à détecter les marqueurs impactant la durée de survie des patients. Obtenir l'information des gènes ou des groupes de gènes impactant la survie permet de mettre en place le parcours de soin approprié. Nous avons de plus en plus souvent accès à l'expression d'un grand nombre de gènes pour un individu. Par conséquent, le nombre de variables explicatives a explosé. Le nombre de gènes codants chez l'humain est de l'ordre de 20 mille (Harrow et al., 2012). En parallèle, la taille de l'échantillon n'a pas augmenté. Le modèle de Cox est le modèle le plus couramment utilisé en analyse de survie. Mais l'estimation classique des paramètres n'est plus consistante quand le nombre de variables est très important. Un travail assez important a déjà été effectué pour proposer des méthodes adaptées à la grande dimension en utilisant le critère d'estimation pénalisée présenté en Section 1.1. Ce sont les méthodes de régularisation et de *Screening*, qui ont tout d'abord été implémentées dans un cadre linéaire (Tibshirani, 1996; Zou, 2006; Verweij et al., 1994; Fan et al., 2001) et adaptées par la suite au cadre de la survie (Tibshirani, 1997; Zhang et al., 2007; Fan et al., 2010a; Fan et al., 2010b; Hong et al., 2018; Zhao et al., 2012). L'idée des méthodes de régularisation, la méthode Lasso et de ses dérivées, est de pénaliser la vraisemblance afin

que les variables non pertinentes soient mises à zéro. Cependant, des auteurs ont mis en évidence des problèmes d'instabilité de ces méthodes (Fan et al., 2008; Michiels et al., 2005). C'est pour cette raison que les méthodes de *Screening* ont été développées. Ces méthodes font une prés-sélection à l'aide d'un score et une procédure Lasso est appliquée sur cette pré-sélection afin de garder seulement les variables pertinentes (celles qui sont non-nulles). Les méthodes de régularisation et de *Screening* seront détaillées en Section 2.2.1 et Section 2.2.2.

Un second objectif de l'analyse de survie consiste à prédire la durée de survie des patients. La fonction de survie est la probabilité que la durée de survie d'un patient soit supérieure à un temps donné. À partir de l'estimation de cette fonction, on peut soit prédire la durée de survie soit prédire le moment où le risque de décès augmente de manière significative. En grande dimension, les modèles habituellement utilisés ne peuvent plus être appliqués. Le modèle de Cox qui est le modèle de référence en analyse de survie en est un exemple, sa procédure classique d'estimation n'étant plus adaptée. Dans ce cadre, deux choix sont possibles. Le premier choix concerne l'utilisation des méthodes de régularisation. Ces méthodes utilisent le critère d'estimation pénalisée, comme présenté dans le paragraphe précédent. La seconde possibilité concerne l'utilisation de méthodes d'apprentissage statistique. C'est pour cette raison que celles-ci se développent depuis quelques années et de nombreux auteurs s'intéressent sur le potentiel des méthodes de *machine learning*, comme les Forêts aléatoires de survie (Breiman, 2001; Boulesteix et al., 2012; Ishwaran et al., 2008), les machines à support vecteur (Vapnik, 1998) ou encore les réseaux de neurones (Ching et al., 2018; Katzman et al., 2018), pour la prédiction de survie en grande dimension. Dans cette thèse, nous nous intéressons au potentiel des ces derniers et cette étude est présentée dans le chapitre 3.

1.2 Modèles

1.2.1 Notations

Dans cette thèse, nous introduisons les notations suivantes : nous considérons un échantillon à n individus et T_i et C_i sont considérés indépendants et correspondent respectivement au temps de survie et au temps de censure de l'individu i. $X_{i.} = (X_{i1}, \ldots, X_{ip})^T \in \mathbb{R}^p$ va être l'ensemble des variables de l'individu i avec p le nombre de variables explicatives. Le temps observé noté $\widetilde{T}_i = \min(T_i, C_i)$ est le minimum entre le temps de l'événement d'intérêt et le temps de censure. Enfin, l'indicateur de censure est noté $\delta_i = \mathbb{1}_{\{T_i \leq C_i\}}$, il vaut 1 si l'événement d'intérêt est observé pour l'individu i.

Pour caractériser la distribution du temps de survie T, différentes fonctions du temps sont utilisées. Il est important de les définir et de montrer comment elles sont liées les unes aux autres (Klein et al., 1997). T représentant le temps de survie auquel a lieu l'événement d'intérêt est une variable aléatoire. Plusieurs fonctions caractérisent la distribution de T. Tout d'abord, il y a la fonction de survie correspondant à la probabilité qu'un individu i survive au-delà d'un temps t. Le risque instantané de décès est la probabilité que l'événement ait lieu pour l'individu i dans un intervalle de temps très court sachant qu'il n'avait pas encore eu lieu au début de l'intervalle et la fonction de densité correspond à la probabilité que l'événement ait lieu au temps t. Enfin, la fonction de risques cumulés correspond à la totalité des risques instantanés auxquels l'individu est exposé depuis le début de l'étude. Si l'on connaît une de ces fonctions, il est possible de déterminer les autres.

La première fonction utile pour modéliser les données de survie est la fonction de survie. Cette fonction est la probabilité qu'un individu i survive un temps plus long que t :

$$S(t) = \mathbb{P}(T \geq t). \tag{1.5}$$

Si T est une variable aléatoire continue alors $S(t)$ est une fonction continue et strictement décroissante. La fonction de survie est le complément de la fonction de répartition, c'est-à-dire $S(t) = 1 - F(t)$, où $F(t) = \mathbb{P}(T \leq t)$. De plus, la fonction de survie est l'intégrale de la densité f pour la queue de distrbution :

$$S(t) = \mathbb{P}(T > t) = \int_t^{+\infty} f(u)du.$$

On peut également obtenir des temps de survie discrets en analyse de survie, c'est par exemple le cas quand les temps de survie sont regroupés en intervalle. Supposons que T prend les valeurs t_l, $l = 1, \ldots, L$ avec probabilité $p(t_l) = \mathbb{P}(T = t_l)$, $l = 1, \ldots, L$, où $t_1 < t_2 < \ldots < t_L$, la fonction de survie pour une variable aléatoire discrète est :

$$S(t) = \mathbb{P}(T > t) = \sum_{t_l > t} p(t_l). \tag{1.6}$$

Le risque instantané est le plus souvent utilisé pour définir les modèles de survie. Le risque instantané est défini par :

$$\lambda(t) = \lim_{\Delta t \to 0} \frac{1}{\Delta t} \mathbb{P}(t \leq T < t + \Delta t | T > t), \tag{1.7}$$

et pour une variable aléatoire T continue, on peut écrire :

$$\lambda(t) = \frac{f(t)}{S(t)} = -\frac{d\left[\ln(S(t))\right]}{dt}. \tag{1.8}$$

Donc la fonction de risques cumulés est défini par :

$$\Lambda(t) = \int_0^t \lambda(u)du = -\ln(S(t)). \tag{1.9}$$

Ainsi, pour des durées de vie continues, on a :

$$S(t) = \exp[-\Lambda(t)] = \exp\left[-\int_0^t \lambda(u)du\right]. \tag{1.10}$$

Quand T est une variable aléatoire discrète, le risque instantané est donné par :

$$\lambda(t_l) = \mathbb{P}(T = t_l | T \geq t_l) = \frac{p(t_l)}{S(t_{l-1})}, \;\; l = 1, \ldots, L, \tag{1.11}$$

avec $S(t_0) = 1$. Comme $p(t_l) = S(t_{l-1}) - S(t_l)$ et d'après l'équation (1.11), le risque instantané peut aussi s'écrire : $\lambda(t_l) = 1 - \frac{S(t_l)}{S(t_{l-1})}$. On peut remarquer que dans le cas discret la fonction de survie peut être écrite comme le produit des probabilités conditionnelles de survie :

$$S(t) = \prod_{t_l \leq t} \frac{S(t_l)}{S(t_{l-1})}. \tag{1.12}$$

Ainsi, la fonction de survie est liée au risque instantané par :

$$S(t) = \prod_{t_l \leq t} \left(1 - \lambda(t_l)\right). \tag{1.13}$$

Nous venons de présenter les fonctions utilisées en analyse de survie, nous allons maintenant présenter les approches possibles pour estimer ces fonctions ou les paramètres de ces fonctions. L'estimation peut être soit paramétrique soit non-paramétrique soit semi-paramétrique selon la définition du modèle. L'estimation paramétrique consiste à estimer le ou les paramètres inconnus d'un modèle statistique, l'estimation non-paramétrique consiste à estimer une fonction inconnue et on parle d'estimation semi-paramétrique lorsqu'on doit estimer à la fois une fonction inconnue et des paramètres inconnus. L'estimateur de Kaplan-Meier (Kaplan et al., 1958), par exemple, est un estimateur non-paramétrique de la fonction de survie. Le modèle de Cox (Cox, 1972) est un modèle semi-paramétrique, c'est-à-dire qu'il se décompose à la fois à partir d'une fonction inconnue et de paramètres inconnus. L'estimation sera donc semi-paramétrique. Le modèle de durée de vie accélérée (Kalbfleisch et al., 2002) (AFT) et le modèle des risques accélérés (Chen et al., 2000) (AH) peuvent être soit paramétriques soit semi-paramétriques. Si une hypothèse peut être faite sur la distribution de la fonction du risque de base, celle-ci s'exprime à partir des paramètres d'une loi usuelle connue et on cherchera à estimer ces paramètres. L'estimation sera donc paramétrique. En revanche, si aucune hypothèse n'est faite sur la distribution de la fonction du risque de base alors le modèle de durée de vie accéléré et le modèle AH seront semi-paramétriques car on aura besoin d'estimer à la fois une fonction inconnue et des paramètres comme c'est le cas pour le modèle de Cox. Nous commençons par présenter les estimateurs non-paramétriques (Kaplan et al., 1958 ; Nelson, 1972 ; Aalen, 1978) en Section 1.2.2. Nous présentons ensuite le modèle de Cox (Cox, 1972) qui est un modèle semi-paramétrique à risques proportionnels en Section 1.2.3. Nous introduisons en Section 1.2.4 le modèle de durée de vie accéléré (AFT) (Kalbfleisch et al., 2002) pouvant être paramétrique ou semi-paramétrique et dont l'effet des variables va accélérer ou décélérer la survie des individus. Nous finirons par introduire le modèle AH (Chen et al., 2000) qui peut également être paramétrique ou semi-paramétrique et dont l'effet des variables va augmenter le risque des individus.

1.2.2 Estimateurs non-paramétriques

Pour estimer les quantités présentées dans le paragraphe précédent, plusieurs estimateurs non-paramétriques existent. Par exemple, l'estimateur Kaplan-Meier (Kaplan et al., 1958) est utilisé pour estimer la fonction de survie $\hat{S}(t)$. L'estimateur de Nelson-Aalen (Nelson, 1972 ; Aalen, 1978) est utilisé pour estimer le risque cumulé $\hat{H}(t)$. Dans cette section, nous supposons n temps distincts $t_1 < t_2 < \ldots < t_n$ et au temps t_i d_i événements (décès) se sont produits. Y_i est le nombre d'individus à risque au temps t_i, c'est-à-dire le nombre d'individus pour lesquels l'événement d'intérêt ne s'est pas encore produit en t_{i-1}. La quantité $\frac{d_i}{Y_i}$ donne une estimation de la probabilité conditionnelle qu'un individu i ait survécu jusqu'au temps t_i décède au temps t_i. La construction de l'estimateur de la fonction de survie et du risque cumulé s'effectue à partir de cette quantité.

1 Estimateur de Kaplan-Meier

L'estimateur de la fonction de survie, proposé par Kaplan et al. (1958), est également appelé produit-limite car il s'obtient comme la limite d'un produit. L'idée ce cet estimateur provient de la remarque suivante : survivre après un temps t signifie être en vie juste avant le temps t et ne pas mourir au temps t, autrement dit si $t'' < t' < t$:

$$\begin{aligned} P(T > t) &= \mathbb{P}(T > t'; T > t) \\ &= \mathbb{P}(T > t | T > t')\mathbb{P}(T > t') \\ &= \mathbb{P}(T > t | T > t')\mathbb{P}(T > t' | T > t'')\mathbb{P}(T > t'') \end{aligned}$$

En considérant les temps d'événements $t_1 < t_2 < \ldots < t_n$, on obtient :

$$\mathbb{P}(T > t_k) = \prod_{i=1}^{k} \mathbb{P}(T > t_k | T > t_{k-1}), \tag{1.14}$$

avec $t_0 = 0$ et $\mathbb{P}(T > t_0) = 1$. Nous rappelons que Y_i est le nombre d'individus à risque de décéder avant le temps t_i et d_i le nombre de décès en t_i. La probabilité de mourir dans l'intervalle $]t_{i-1}, t_i]$ sachant que l'on était vivant en t_{i-1}, *i.e.* $p_i = \mathbb{P}(T \leq t_i | T > t_{i-1})$ peut être estimée par :

$$\hat{p}_i = \frac{d_i}{Y_i}.$$

Si les temps d'événements sont supposés distincts, on a :

$$\begin{aligned} d_i &= 0 \text{ en cas de censure en } t_i, \text{ quand } \delta_i = 0 \\ d_i &= 1 \text{ en cas de décès en } t_i, \text{ quand } \delta_i = 1. \end{aligned}$$

On obtient alors l'estimateur de Kaplan-Meier :

$$\begin{aligned} \hat{S}(t) &= \prod_{t_i \leq t} \left(1 - \frac{\delta_i}{Y_i}\right) \\ &= \prod_{t_i \leq t} \left(1 - \frac{\delta_i}{n - (i-1)}\right) \\ &= \prod_{t_i \leq t} \left(\frac{n-i}{n-i+1}\right)^{\delta_i}. \end{aligned}$$

Il existe également une forme explicite de l'estimateur de Kaplan-Meier dans le cas d'*ex-aequo*, nous renvoyons à Kaplan et al. (1958) pour plus de détails. $\hat{S}(t)$ est une fonction en escalier décroissante et continue à droite. Cet estimateur fournit une estimation moyenne de la fonction de survie efficace pour les données censurées à droite. Il peut également être utilisé pour estimer le risque cumulé $H(t) = -\log(S(t))$. L'estimateur du risque cumulé est : $\hat{H}(t) = -\log[\hat{S}(t)]$ et il est appelé estimateur de Breslow.

2 Estimateur de Nelson-Aalen

Pour estimer le risque cumulé, une autre alternative existe l'estimateur de Nelson-Aalen. Celui-ci a de meilleurs performances sur les données dont la taille de l'échantillon est petite. Cet estimateur de Nelson-Aalen a tout d'abord été suggéré par Nelson (1972) dans un contexte de fiabilité. Il a ensuite été de nouveau découvert par Aalen (1978) qui a adapté l'estimateur en utilisant les processus de comptage. Par définition du risque cumulé, on a :

$$\Lambda(t) = \int_0^t \lambda(u) du = \int_0^t \frac{f(u)}{S(u)} du.$$

Dans le cas où T n'est pas défini en tout point de $\mathbb{R}^+$, on peut définir le risque cumulé en utilisant la défintion de la densité :

$$\Lambda(t) = -\int_0^t \frac{S(du)}{S(u^-)}.$$

Rappelons que $\widetilde{T} = \min(T, C)$ et posons les fonctions $H(t) = \mathbb{P}(\widetilde{T} > t)$ et $H_1(t) = \mathbb{P}(\widetilde{T} > t, \delta = 1)$ et introduisons $G(t)$ la fonction de survie de la variable C. D'après les hypothèses d'indépendance, on obtient :

$$\begin{aligned} H(t) &= \mathbb{P}(\widetilde{T} > t) = \mathbb{P}(T > t, C > t) = S(t)G(t) \\ H_1(t) &= \mathbb{P}(\widetilde{T} > t, \delta = 1) = \mathbb{P}(T > t, C \geq T) = \mathbb{E}\left(\mathbb{1}_{\{T>t\}} G(T^-)\right) \\ &= \int_t^\infty G(u^-) f(u) du = -\int_t^\infty G(u^-) S(du). \end{aligned}$$

On obtient donc $H_1(dt) = -G(t^-)S(dt)$ et le risque cumulé peut s'écrire :

$$\text{Soit } \frac{S(du)}{S(u^-)} = \frac{-H_1(du)}{H(u^-)} \text{ et } \Lambda(t) = -\int_0^t \frac{H_1(du)}{H(u^-)}.$$

En remplaçant les quantités $H_1(t)$ et $H(t)$ par leurs quantités empiriques, on obtient :

$$\hat{H}(u) = \frac{1}{n}\sum_{i=1}^n \mathbb{1}_{\{t_i > u\}} \text{ et } \hat{H}_1(u) = \frac{1}{n}\sum_{i=1}^n \mathbb{1}_{\{t_i > u, \delta_i = 1\}},$$

et l'estimateur de Nelson-Aalen est donné par les expressions suivantes :

$$\hat{\Lambda}(t) = -\int_0^t \frac{\hat{H}_1(du)}{\hat{H}(u^-)} = \sum_{t_i \leq t} \frac{\sum_{j=1}^n \mathbb{1}_{\{t_j = t_i, \delta_j = 1\}}}{\sum_{j=1}^n \mathbb{1}_{\{t_j \geq t_i\}}} = \sum_{t_i \leq t} \frac{d_i}{Y_i},$$

où Y_i est le nombre d'individus à risque avant le temps t_i et d_i représente le nombre d'individus décédés en t_i. L'estimateur de Nelson-Aalen est une fonction en escalier. On peut retrouver un estimateur de la fonction de survie de la manière suivante :

$$\hat{S}(t) = \exp\left(-\hat{\Lambda}(t)\right). \tag{1.15}$$

1.2.3 Modèle de Cox

Le modèle de Cox (Cox, 1972) est le modèle de référence dans le domaine de l'analyse de survie. Il permet d'étudier le temps écoulé jusqu'à un événement d'intérêt. L'avantage du modèle de Cox est sa capacité à prendre en compte des données censurées. Plusieurs types de censure existent en analyse de survie. Le type de censure présent dans nos données est appelé censure à droite. On dit que des données sont censurées à droite quand le temps observé est plus petit que la durée de survie. Le modèle de Cox est défini pour un individu i à partir du risque instantané λ qui est une fonction du temps conditionnellement aux variables explicatives données de l'individu i $X_{i.} = (X_{i1}, \ldots, X_{ip})^T \in \mathbb{R}^p$:

$$\lambda(t|X_{i.}) = \alpha_0(t)\exp(\beta^T X_{i.}), \tag{1.16}$$

avec $\alpha_0(t)$ risque de base et $\beta = (\beta_1, \ldots, \beta_p)^T \in \mathbb{R}^p$ le vecteur des coefficients de régression. Le risque de base est le risque instantané de décès quand toutes les variables sont nulles. Cette fonction $\lambda(t|X_{i.})$ correspond au risque instantané de décès au temps t sachant que l'individu i est vivant avant le temps t. Le risque instantané peut être séparé en deux parties car le terme $\alpha_0(t)$ dépend seulement du temps et va être le même pour tous les individus à un instant donné et le second terme de (1.16) dépend seulement des variables propres à chaque individu. Le modèle de Cox (Cox, 1972) est un modèle semi-paramétrique car l'estimation implique d'estimer un vecteur de paramètres de $\mathbb{R}^p$ et une fonction $\alpha_0(t)$. Mais il est possible d'estimer β sans avoir besoin de connaître la fonction du risque de base $\alpha_0(t)$ grâce à la vraisemblance partielle de Cox. Cela est

utile quand nous nous intéressons au pronostic, c'est-à-dire lorsque nous voulons seulement connaître les facteurs influençant la survie. En revanche, quand on souhaite faire de la prédiction de la durée de survie on utilise la fonction de survie $S(t)$ dépendant du risque instantané. Il est nécessaire de connaître le risque instantané complet et on aura donc besoin de l'estimation du risque de base $\alpha_0(t)$ et du vecteur de régression β. Une des caractéristiques du modèle de Cox liée à sa forme multiplicative entre une fonction du temps et une fonction des covariables est que le rapport des risques instantanés de deux individus dépend seulement des facteurs auxquels ils sont soumis et est constant au cours du temps :

$$\frac{\lambda(t|X_{i.})}{\lambda(t|X_{j.})} = \frac{\alpha_0(t)\exp(\beta_1 X_{i1} + \ldots + \beta_p X_{ip})}{\alpha_0(t)\exp(\beta_1 X_{j1} + \ldots + \beta_p X_{jp})} = \exp(\beta_1(X_{i1} - X_{j1}) + \ldots + \beta_p(X_{ip} - \beta_p X_{jp})).$$

Le modèle de Cox est un modèle à risques proportionnels et afin de pouvoir l'appliquer sur des données il est nécessaire que les variables de celles-ci ne dépendent pas du temps. Pour effectuer cette vérification, il est nécessaire de faire un test dont les hypothèses sont :

$$H_0 : \beta_j(t) \equiv \beta_j \;\; vs \;\; H_1 : \beta_j(t) \equiv \beta_j + \theta_j g_j(t), \tag{1.17}$$

où $g_j(t)$ est un processus prédictif. Pour vérifier l'hypothèse des risques proportionnels, il existe plusieurs démarches basées soit sur des méthodes graphiques soit sur un test. Ces démarches sont développées de manière plus approfondie dans le Chapitre 3. La première démarche consiste à comparer graphiquement des courbes de survie de deux individus. Si celles-ci sont parallèles, alors cela signifie que l'hypothèse des risques proportionnels est vérifiée. La seconde approche concerne l'analyse des résidus de Schoenfeld (Schoenfeld, 1982), elle est à la fois basée sur la visualisation graphique et sur un test. Le principe du test des résidus de Schoenfeld (résumé en Section 3.4.1 et détaillé en Annexe E) consiste à calculer dans un premier temps les résidus de Schoenfeld et ensuite à faire un test de corrélation entre les résidus et le temps. Il est également possible de tracer les résidus en fonction du temps ou d'une transformation du temps. Si les résidus sont répartis de façon homogène autour de la droite $y = 0$, cela signifie que l'hypothèse des risques proportionnels est vérifiée. Les résultats obtenus à partir de ces méthodes doivent être utilisés avec précaution car le test permet de détecter un certain nombre d'erreurs de spécification du modèle en plus de la non-proportionnalité (Keele, 2010). Nous renvoyons à l'Annexe E pour plus de détails sur les erreurs possibles d'interprétation du test. Si l'hypothèse des risques proportionnels est vérifiée, la prochaine étape concerne l'estimation des coefficients de régression. L'estimation des coefficients de régression est faite à partir de la maximisation de la log-vraisemblance partielle de Cox.

La vraisemblance partielle de Cox (Cox, 1975) est basée sur la probabilité qu'un individu i décède à un temps observé sachant qu'un décès a lieu. Soient $t_1, \ldots, t_n$ l'ensemble des temps observés ordonnés pour n individus et $R(t_i)$ est l'ensemble des individus à risque au temps t_i, la probabilité que l'individu i décède sachant qu'un évènement a lieu au temps t_i est :

$$\frac{\exp(\beta^T X_{i.})}{\sum_{l \in R(t_i)} \exp(\beta^T X_{l.})}. \tag{1.18}$$

La vraisemblance partielle de Cox (Cox, 1972) est une partie de la vraisemblance totale ne dépendant pas du risque de base $\alpha_0(t)$ et elle s'écrit :

$$L(\beta) = \prod_{i=1}^{n} \left[\frac{\exp(\beta^T X_{i.})}{\sum_{l \in R(t_i)} \exp(\beta^T X_{l.})} \right], \tag{1.19}$$

et nous obtenons dans le cas de la censure à droite la vraisemblance partielle de Cox :

$$L(\beta) = \prod_{i=1}^{n} \left[\frac{\exp(\beta^T X_{i.})}{\sum_{l \in R(t_i)} \exp(\beta^T X_{l.})} \right]^{\delta_i},$$

avec $R(t_i)$ les individus à risque au temps t_i et δ_i l'indicateur de censure. Maximiser $\mathcal{L}(\beta)$ permet en dimension raisonnable d'estimer correctement le paramètre β_0 du modèle de Cox. En grande dimension (*i.e.* quand le nombre de variables est supérieur à la taille de l'échantillon), la procédure classique d'estimation consistant à maximiser la log-vraisemblance partielle de Cox ne fonctionne plus. La procédure Lasso (Tibshirani, 1996) est une des méthodes de régularisation utilisée en grande dimension afin de réduire le nombre de variables. Cette procédure est détaillée en Section 2.2.1 et celle-ci mettra à zéro les variables non pertinentes. Elle a été adaptée dans le cadre de la survie (Tibshirani, 1997) en utilisant la vraisemblance partielle de Cox. Un terme de régularisation est ajouté à la log-vraisemblance partielle de Cox : $\mathcal{L}(\beta) + \lambda ||\beta||_1$ qui va permettre de la pénaliser et mettre les variables non pertinentes à zéro. Le vecteur de régression β est estimé à partir de la log-vraisemblance partielle de Cox :

$$\hat{\beta} = \arg\min_{\beta} \{-\mathcal{L}(\beta)\}. \tag{1.20}$$

La fonction de survie est définie à partir du risque instantané :

$$S(t|X_{i.}) = \mathbb{P}(T_i > t|X_{i.}) = \exp\left(- \int_0^t \underbrace{\alpha_0(s) \exp(\beta^T X_{i.})}_{\lambda(s|X_{i.})} ds \right).$$

Pour prédire la survie d'un individu, nous avons donc besoin de l'estimation du risque de base $\alpha_0(t)$. Il est difficile de l'estimer directement, mais des procédures en deux étapes existent comme par exemple celle de Lemler (2016). En effet, après l'estimation des coefficients de régression une seconde étape est effectuée afin d'estimer $\alpha_0(t)$. Pour estimer ce risque $\alpha_0(t)$, plusieurs estimateurs existent. Le risque cumulé de base peut être estimé à l'aide de l'estimateur de Breslow :

$$\Lambda_0(t) = \sum_{t \geq t_i} \frac{\delta_i}{\sum_{l \in R(t_i)} \exp(\widehat{\beta}^T X_{l.})} \tag{1.21}$$

En lissant les incréments de (1.21), Ramlau-Hansen (1983) a proposé un estimateur à noyau du risque de base $\alpha_0(t)$ dans lequel apparaît l'estimateur du paramètre de régression obtenu à la première étape en maximisant la vraisemblance partielle de Cox défini par :

$$\widehat{\alpha}_h(t) = \frac{1}{nh} \sum_{i=1}^{n} \int_0^{\tau} K\left(\frac{t-u}{h}\right) \frac{\delta_i}{\sum_{l \in R(t_i)} \exp(\widehat{\beta}^T X_{l.})} du, \tag{1.22}$$

avec $K : \mathbb{R} \to \mathbb{R}$ une fonction d'intégrale 1, appelée noyau et h est un paramètre réel strictement positif, appelé fenêtre. Ce paramètre h peut être obtenu par validation croisée ou par la méthode Goldenshluger & Lepski (Goldenshluger et al., 2011) par exemple. Pour plus de détails sur cette procédure, nous renvoyons à la thèse de Lemler (2014). Nous pouvons finalement en déduire un estimateur de la fonction de survie :

$$\widehat{S}(t|X_{i.}) = \exp\left(- \int_0^t \widehat{\alpha}_{\widehat{h}}(s) \exp(\widehat{\beta}^T X_{i.}) ds \right),$$

permettant de faire la prédiction de survie d'un individu i.

1.2.4 Modèle de vie accélérée (AFT)

Le modèle de vie accélérée (*Accelerated Failure Time model* AFT) (Kalbfleisch et al., 2002) est un modèle décrivant la relation entre les variables explicatives et les temps de survie. Ce modèle peut être défini à partir d'une régression linéaire de la variable d'intérêt. La variable T est positive, on peut donc modéliser $\log T$ par une régression linéaire :

$$\log T_i = \beta^T X_{i.} + \epsilon_i, \tag{1.23}$$

avec $X_{i.}$ les variables, β le vecteur de régression et ϵ_i le terme d'erreur. Le logarithme du temps d'événement est relié linéairement aux variables explicatives avec un terme d'erreur. Si on fait l'hypothèse que le terme d'erreur suit une loi normale, le modèle AFT sera un modèle de vie accéléré log-normal. En faisant d'autres hypothèses sur la distribution du terme d'erreur ϵ_i, on obtient un modèle avec une certaine loi de probabilité connue. Les lois les plus couramment utilisées pour caractériser le modèle de vie accélérée sont les lois exponentielle, Gompertz, Weibull, log-logistique et log-normale. En revanche, si aucune hypothèse n'est faite sur la distribution de ϵ_i alors le modèle de vie accéléré sera semi-paramétrique.

On appelle $S_0(t)$ la fonction de survie de base obtenue pour $X = 0$. Elle correspond à la queue de la distribution de $\exp(\epsilon_i)$. La fonction de survie dans un modèle AFT s'écrit

$$S(t|X_{i.}) = S_0(t \exp(\beta^T X_{i.})). \tag{1.24}$$

Nous en déduisons l'expression de la fonction de risque :

$$\lambda(t|X_{i.}) = \exp(\beta^T X_{i.}) \alpha_0(t \exp(\beta^T X_{i.})). \tag{1.25}$$

Nous voyons à partir de l'équation (1.25) que les variables ont un effet multiplicatif sur t plutôt que sur la fonction du risque, comme c'est le cas pour le modèle de Cox. Si $\exp(\beta^T X_{i.})$ est supérieur à 1, les variables vont avoir un effet accélérant sur la survenue de l'événement, c'est-à-dire qu'elles vont diminuer le temps de survie de l'individu i. En revanche, si $\exp(\beta^T X_{i.})$ est inférieur à 1, les variables vont avoir un effet décélérant sur le risque, c'est-à-dire qu'elles vont augmenter la durée de survie de l'individu i. Par exemple, si $\exp(\beta^T X_{i.}) = 2$ alors l'individu vieillit deux fois plus rapidement que le groupe contrôle et si $\exp(\beta^T X_{i.}) = \frac{1}{2}$, alors l'évolution de vieillissement de l'individu est deux fois moins rapide que le groupe contrôle. Pour utiliser le modèle AFT, il n'est donc pas nécessaire de vérifier l'hypothèse des risques proportionnels. Mais certaines configurations de ce modèle peuvent s'adapter soit à l'hypothèse AFT (*i.e.* l'hypothèse de l'effet multiplicatif des variables par rapport aux temps de survie), soit à l'hypothèse des risques proportionnels. En effet, si on a un modèle de vie accélérée exponentielle ou de Weibull le modèle peut vérifier l'hypothèse des risques proportionnels et l'hypothèse des effets multiplicatifs. Mais en choisissant un modèle de vie accélérée log-logistique le modèle ne vérifie pas l'hypothèse des risques proportionnels et si la loi Gompertz est utilisée, alors l'hypothèse des risques proportionnels est vérifiée mais celle des effets multiplicatifs ne l'est pas. L'estimation des paramètres d'un modèle AFT peut être réalisée par maximum de vraisemblance quand la fonction du risque de base α_0 est connue, c'est-à-dire qu'une fonction paramétrique est utilisée pour α_0. Si aucune hypothèse paramétrique est faite sur α_0, alors le modèle AFT sera un modèle semi-paramétrique comparable au modèle de Cox. Mais il n'existe pas d'équivalent de la vraisemblance partielle de Cox pour le modèle AFT permettant d'éliminer le paramètre α_0. L'estimation des paramètres β et α_0 est donc plus complexe et nous renvoyons à Kalbfleisch et al. (2002) pour plus de détails sur l'inférence des paramètres. Nous nous intéresserons dans cette thèse uniquement au cas paramétrique. Ce modèle est une alternative au modèle de Cox pour les données de survie censurées quand celles-ci ne respectent pas l'hypothèse des risques proportionnels. De

plus, en utilisant un modèle AFT, l'effet des variables explicatives sur la durée de survie est directement mesuré. Cette caractéristique permet une facilité d'interprétation des résultats car les paramètres mesurent l'effet d'une variable sur la moyenne des temps de survie.

1.2.5 Modèle des risques accélérés (AH)

Le modèle des risques accélérés AH (pour *Accelerated Hazards model*) a été proposé par Chen et al. (2000). Ce modèle permet une grande flexibilité pour modéliser les données de survie. Le risque instantané du modèle AH est défini pour un individu i par :

$$\lambda_{AH}(t|X_{i.}) = \alpha_0(t\exp(\beta^T X_{i.})), \tag{1.26}$$

avec α_0 le risque de base et β le vecteur des paramètres de régression. Dans le cadre d'un modèle avec seulement une variable binaire considérée qui correspond au traitement, le risque instantané s'écrit de la façon suivante :

$$\lambda_1(t) = \alpha_0(\beta t). \tag{1.27}$$

Le vecteur de régression β caractérise l'influence des variables sur le temps de survie des individus et $\exp(\beta^T X_{i.})$ est un facteur altérant l'échelle de temps sur le risque instantané. La valeur positive ou négative de $\beta^T X_{i.}$ impliquera respectivement une accélération ou décélération du risque. Si $\exp(\beta^T X_{i.}) = \frac{1}{2}$, alors le risque de l'individu i progresse deux fois moins rapidement et si $\exp(\beta^T X_{i.}) = 2$, le risque de l'individu i s'accroît deux fois plus rapidement.

La fonction de survie pour un modèle AH s'écrit :

$$S_{AH}(t|X_{i.}) = \left[S_0(t\exp(-\beta^T X_{i.}))\right]^{\exp(\beta^T X_{i.})} \tag{1.28}$$

où $S_0(u)$ est la fonction de survie du groupe contrôle. Les courbes de survie d'un modèle AFT ne peuvent pas se croiser car l'effet des variables agit de manière multiplicative sur l'échelle du temps pour la fonction de survie. Autrement dit, les courbes de survie de deux individus auront la même forme, mais une des courbes aura du retard ou de l'avance sur le temps de survie. Les courbes des risques instantanés du modèle AFT peuvent se croiser, mais pas celles du modèle de Cox. L'effet des variables dans un modèle de Cox vont agir de manière multiplicative sur le risque instantané, ce qui implique les courbes des risques instantanés ont la même forme mais dont certaines courbes vont avoir de l'avance ou du retard sur l'échelle des temps. Contrairement à ces deux modèles, les courbes de survie et des risques instantanés pour des patients différents du modèle AH peuvent se croiser.

L'estimation des paramètres dans le modèle AH peut être exécutée par maximum de vraisemblance quand la fonction du risque de base est connue. Si le risque de base est non-spécifié, l'estimation semi-paramétrique usuelle du modèle de Cox peut être adaptée afin d'estimer les paramètres de régression du modèle AH. Chen et al. (2000) a proposé une procédure d'estimation motivée par le fait que la différence entre les risques instantanés $\lambda_{AH}(t)$ et $\alpha_0(t)$ provient d'un changement d'échelle de temps. Chen et al. (2000) remarque que si $\beta_a \in \mathbb{R}^+$ et T une variable aléatoire positive avec un risque instantané $\lambda(t)$ alors le risque instantané de $\beta_a T$ est $\frac{\lambda(\frac{t}{\beta_a})}{\beta_a}$. De plus, si l'équation (1.27) est vérifiée et β_0 est le vrai paramètre, le risque instantané de la transformation de T : $T_a = \exp(\beta_a^T X_{i.})T$ est :

$$\lambda_a(t|X) = \alpha_0\left(t\exp\left[\left(\frac{\beta_0}{\beta_a}\right)^T X_{i.}\right]\right)\exp(-\beta_a^T X_{i.}). \tag{1.29}$$

Nous pouvons remarquer que quand $\beta_a = \beta_0$ dans (1.29), l'équation devient :

$$\lambda_a(t|X) = \alpha_0(t)\exp(-\beta_0^T X_{i.}). \tag{1.30}$$

Dans ce cas, on retrouve donc la proportionnalité entre les risques instantanés quand les valeurs correctes des paramètres β_0 sont utilisées pour la transformation mais avec un rapport $\exp(-\beta_0^T X_{i.})$. Basé sur cette observation, Chen et al. (2000) a proposé un algorithme utilisant la procédure d'estimation de la log-vraisemblance partielle de Cox en considérant (1.30) pour estimer les paramètres du modèle AH. Nous renvoyons à Chen et al. (2000) pour les détails de l'algorithme. L'algorithme proposé par Chen et al. (2000) revient à résoudre l'ensemble des équations $U(\beta) = 0$ où $U(\beta) = (U_1(\beta), \ldots, U_p(\beta))^T$ et

$$U_r(\beta) = \sum_{i=1}^{n} \delta_i \left[X_{i.} - \frac{\sum_{j=1}^{n} \mathbb{1}_{\{t_j \exp(\beta^T X_j) > t_i \exp(\beta^T X_{i.})\}} \exp(-\beta^T X_j) X_j}{\sum_{j=1}^{n} \mathbb{1}_{\{t_j \exp(\beta^T X_j) > t_i \exp(\beta^T X_{i.})\}} \exp(-\beta^T X_j)} \right], r = 1, \ldots, p. \tag{1.31}$$

Pour prédire la durée de survie, nous avons besoin d'estimer le risque instantané. Si une solution de (1.31) est obtenue, le risque cumulé de base peut être estimé à l'aide de l'estimateur de Breslow (Breslow et al., 1984). Pour estimer le risque de base, Chen et al. (2000) se placent dans le cadre des processus de comptages car en supposant (1.27) la variable X peut prendre soit la valeur 0 soit la valeur 1. L'estimation du risque cumulé de base $\Lambda_0(t) = \int_0^T \alpha_0(s)ds$ est obtenue en calculant :

$$\widehat{\Lambda}_0(t) = \int_0^t \frac{dN_0(s) + dN_1(\frac{s}{\widehat{\beta}})}{Y_0(s) + Y_1(\frac{s}{\widehat{\beta}})/\widehat{\beta}}, \tag{1.32}$$

où

$$N_i(t) = \sum_j N_{ij}(t) = \sum_j \mathbb{1}_{\{T_j \leq t, \delta_j = 1, X_j = i\}}$$

et

$$Y_i(t) = \sum_j Y_{ij}(t) = \sum_j \mathbb{1}_{\{T_j \geq t, X_j = i\}},$$

pour $i = 0, 1$. Nous pouvons finalement en déduire un estimateur de la fonction de survie :

$$\widehat{S}(t|X_{i.}) = \exp\left(-\widehat{\Lambda}_0(t)\exp(\widehat{\beta}^T X_{i.})\right),$$

permettant de faire la prédiction de survie d'un individu i. Contrairement aux modèles de Cox et AFT, le modèle AH est approprié quand on veut que les courbes des risques instantanés et de survie se croisent. Cela confère à ce modèle plus de flexibilité comparé aux deux autres modèles. De plus, une autre particularité de ce modèle est que les variables vont avoir un effet d'accélération ou décélération sur le risque des individus. Cependant, l'inférence du modèle est plus complexe que celle du modèle de Cox.

1.3 Métriques

Pour évaluer les performances d'un modèle prédictif, il est recommandé d'analyser à la fois son pouvoir de discrimination et de calibration. La discrimination consiste à évaluer la séparation des individus avec un risque élevé de décéder rapidement de ceux dont le risque est faible. La calibration consiste à calculer l'écart entre la probabilité que l'individu décède et la "vraie" probabilité qu'il décède pour chaque intervalle de temps donné. Dans le but de mesurer la discrimination et la calibration du modèle, plusieurs métriques existent en

analyse de survie. Les métriques les plus utilisées sont l'indice de concordance (appelé C-index) (Harrell, 1982) et le score de Brier intégré (appelé *Integrated Brier Score (IBS)*) (Gerds et al., 2006). La métrique C-index permet d'évaluer la discrimination des données, c'est-à-dire la capacité de la prédiction à séparer de façon cohérente les individus dont le risque de décès est important de ceux dont le risque est moindre. La seconde métrique *Integrated Brier Score (IBS)* permet d'évaluer la calibration et la discrimination de la prédiction. Elle calcule l'erreur moyenne entre l'indicatrice de la survie du patient et la probabilité de survie prédite à chaque point de temps.

1.3.1 Indices de concordance

L'indice de concordance (aussi appelé C-index) a tout d'abord été développé par Harrell (1982). Cette métrique est une extension de l'aire sous la courbe ROC (AUC pour *Area Under Curve*) dans le cas des données de survie censurées à droite. C'est pour cette raison que l'aire sous la courbe ROC est présenté dans un premier temps et le C-index est ensuite introduit dans deux contextes différents (continu et discret).

1 AUC et courbe ROC

Le AUC est un indice de performance pour les tests diagnostiques, c'est un indicateur naturel de discrimination pour les modèles binaires. En effet, il peut être utile d'évaluer une prédiction, notée P, à séparer une population malade d'une population non-malade. L'indicateur de la maladie est noté D (*Disease*), il vaut 1 si l'individu est malade et zéro sinon. Il est possible de construire un test diagnostique par dichotomisation de P à partir d'une valeur seuil choisie c. Le test est défini positif si $P > c$ et négatif si $P \leq c$. La probabilité de classifier malade sachant que le statut est malade (*i.e.* vrai positif) est appelée sensibilité et est définie par :

$$Se(c) = \mathbb{P}(P > c|D = 1).$$

La probabilité de classifier non-malade sachant que le statut est non-malade (*i.e.* vrai négatif) est appelée spécificité et est définie par :

$$Sp(c) = \mathbb{P}(P \leq c|D = 0).$$

Pour évaluer la capacité de discrimination de la prédiction sur l'ensemble des valeurs seuil c, on s'intéresse à la courbe ROC correspondant au graphe de la sensibilité *versus* le complément de la spécificité $\{(Se(c), 1 - Sp(c)), c \in \mathbb{R}\}$. Plus l'aire sous la courbe (AUC) est grande, plus la prédiction discrimine bien la population malade de la population non-malade. L'aire sous la courbe représente la probabilité que la prédiction d'un individu malade soit plus élevée que celle d'un individu non-malade :

$$AUC = \mathbb{P}(P_i > P_j|D_i = 1; D_j = 0),$$

avec P_i la prédiction de l'individu malade et P_j la prédiction de l'individu non-malade. Autrement dit, le AUC correspond à la probabilité, pour toute paire d'individus comparables tels que leurs statuts puissent être ordonnés (*i.e.* un individu est malade et un autre individu est non-malade), que le rang entre la valeur de la prédiction soit concordant avec le rang du statut. Si AUC vaut 1, cela signifie que la discrimination est maximale et si AUC vaut 0.5, cela signifie qu'il y a absence de discrimination. On suppose maintenant $n_1 + n_2$ individus, où n_1 sont les individus malades (indicés de $i = 1, \ldots, n_1$) et n_2 sont les individus non-malades (indicés de $j = n_1, \ldots, n_1 + n_2$). Soient $P_1, \ldots, P_{n+m}$ les valeurs observées des prédictions. Pour $i = 1, \ldots, n$, $j = n + 1, \ldots, n + m$, l'indicateur binaire est défini par $conc_{ij} = \mathbb{1}_{\{P_i > P_j\}}$; il vaut 1 si les individus i, j sont concordants et zéro sinon. L'estimation du AUC est le rapport des paires concordantes

sur le nombre total de paires comparables :

$$\widehat{AUC} = \frac{\sum_{i=1}^{n} \sum_{j=n+1}^{n+m} conc_{ij}}{nm}.$$

2 C-index

L'indice de concordance, appelé C-index, est similaire à l'AUC. Il considère également la concordance entre les temps d'événement prédits et les temps d'événement observés pour des paires d'individus. L'indice de concordance mesure donc si la prédiction du modèle étudié correspond au rang des données de survie. Si le temps d'événement d'un individu i est plus petit que celui d'un individu j, un bon modèle prédira une plus grande probabilité de survie pour l'individu j. Cette métrique tient compte des données censurées, et elle prend une valeur entre 0 et 1. Si le C-index est égale à 0.5, cela signifie que le modèle est équivalent à un processus aléatoire.

Le C-index est très utile pour les modèles à risques proportionnels comme le modèle de Cox car le rang entre les temps prédits obtenu en considérant la prédiction linéaire, c'est-à-dire $P_i = (\beta^T X_{i.}) > P_j = (\beta^T X_j)$ est équivalent à $S(t_{(k)}|X_{i.}) < S(t_{(k)}|X_j)$ pour tout $t_{(k)},\ k = 1, \ldots, K$ ($t_{(k)}$ est le temps écoulé entre $t_{(0)}$ et la k^e observation). Cette équivalence est valable pour les modèles à risques proportionnels et certains modèles de transformation linéaire (Fine et al., 1998). Cependant, cette équivalence n'est plus vérifiée pour les modèles à risques non-proportionnels et à temps discret. Dans ce manuscrit, nous nous sommes intéressés à la fois à des modèles à risques proportionnels, le modèle de Cox (Cox, 1972) et à des modèles ne vérifiant pas l'hypothèse des risques proportionnels, comme le modèle AFT (Kalbfleisch et al., 2002), mais qui est un modèle de transformation linéaire (Fine et al., 1998) ou les modèles à temps discret (Biganzoli et al., 1998) et le modèle AH (Chen et al., 2000). C'est pour cela que nous développons dans les paragraphes suivants le C-index dans le cadre où l'équivalence suivante :

$$P_i = (\beta^T X_{i.}) > P_j = (\beta^T X_j) \Leftrightarrow S(t_{(k)}|X_{i.}) < S(t_{(k)}|X_j) \ \forall t_{(k)} \tag{1.33}$$

est vérifiée (modèles des risques proportionnels et modèles de transformation linéaire) et dans le cadre où l'équivalence (1.33) n'est pas vérifiée (modèles à temps discret, modèle AH).

C-index pour les modèles vérifiant (1.33)

Dans le cas des modèles où l'équivalence (1.33) est vraie, le C-index peut être défini sans prédire les temps de survie des individus. Les prédictions linéaires obtenues pour deux individus i et j $P_i = \beta^T X_i > P_j = \beta^T X_j$ sont équivalents à $S(t_{(k)}|X_{i.}) < S(t_{(k)}|X_j)$ pour tout $t_{(k)}$. En absence de censure, pour toute paire d'individus, le C-index est :

$$Ci = \mathbb{P}((\beta^T X_{i.}) > (\beta^T X_{j.})|T_i < T_j).$$

Il correspond à la probabilité que le risque de décès de l'individu i soit plus élevé que celui de l'individu j sachant que le temps observé de l'individu i est plus petit que celui de l'individu j. Le C-index considère également la probabilité de concordance en présence de données censurées. Afin de définir si deux individus i, j sont concordants, il faut d'abord vérifier qu'ils sont comparables, ce qui signifie que leurs temps de survie T_i et T_j peuvent être ordonnés. Leurs rangs peuvent être déterminés à partir des données (T_i, δ_i) et (T_j, δ_j) si le minimum entre T_i et T_j est un temps d'événement. Supposons que le temps observé de l'individu i est plus petit que celui de l'individu j, et que l'événement est observé pour l'individu i, alors les individus sont comparables et la probabilité est égale à :

$$pcomp_{ij} = P(\widetilde{T}_i < \widetilde{T}_j; \delta_i = 1)$$

En présence de censure, le C-index est donc défini de la manière suivante :

$$\begin{aligned} Ci &= \mathbb{P}(P_i > P_j | \widetilde{T}_i < \widetilde{T}_j; \delta_i = 1) \\ &= \frac{pconc_{ij}}{pcomp_{ij}}, \end{aligned} \tag{1.34}$$

où $pconc_{ij}$ est la probabilité de concordance :

$$pconc_{ij} = \mathbb{P}(P_i > P_j \cap T_i < T_j, \delta_i = 1).$$

On suppose qu'on a un échantillon de n individus et on se place dans le cas d'un modèle à risques proportionnels. $\{(t_1, \delta_1, P_1), \ldots, (t_n, \delta_n, P_n)\}$ sont respectivement les temps observés, l'indicateur de censure et la prédiction linéaire. Pour $i, j = 1, \ldots, n\, i \neq j$, on définit les indicateurs :

$$\begin{aligned} comp_{ij} &= \mathbb{1}_{\{(t_i < t_j; \delta_i = 1) \cup (t_i = t_j; \delta_i = 1, \delta_j = 0)\}} \text{ et} \\ conc_{ij} &= \mathbb{1}_{\{P_i > P_j \cap (t_i < t_j; \delta_i = 1 \cup t_i = t_j; \delta_i = 1, \delta_j = 0)\}} \\ &= \mathbb{1}_{\{P_i > P_j\}} comp_{ij}. \end{aligned}$$

L'estimation du C-index pour les modèles à risques proportionnels est égale à :

$$\widehat{Ci} = \frac{\sum_{i=1}^{n} \sum_{j \neq i} conc_{ij}}{\sum_{i=1}^{n} \sum_{j \neq i} comp_{ij}} \tag{1.35}$$

Le C-index permet d'estimer la proportion d'individus dont les rangs des prédictions sont concordants avec ceux des temps d'événements. Si la valeur estimée du C-index est égale à 1, cela signifie que les prédictions obtenues par le modèle possède le même ordre que les temps d'événements. Si la valeur du C-index vaut 0.5, cela signifie que les prédictions du modèle correspond à un processus aléatoire. Plus la valeur estimée du C-index est proche de 1, meilleure est la prédiction du modèle et plus la valeur est proche de 0.5, plus la prédiction du modèle est mauvaise.

C-index pour les modèles ne vérifiant pas (1.33)

Dans le cas des modèles de survie à temps discret par exemple, l'équivalence (1.33) entre les prédictions linéaires obtenues et la prédiction de la survie de deux individus ($P_i > P_j \Leftrightarrow S(t_{(k)}|X_{i.}) < S(t_{(k)}|X_j)$ pour tout $t_{(k)}$) n'est plus vraie. Antolini et al. (2005) a donc proposé une métrique basée sur le C-index dépendant du temps (Antolini et al., 2005). Pour cela, ils se placent dans le cadre de temps discrets. Pour tout temps $t_{(k)}$, ils notent $D(t_{(k)})$ le statut à $t_{(k)}$. $D_i(t_{(k)})$ est égal à 1 si l'individu i est décédé à $t_{(k)}$ et $D_i(t_{(k)})$ vaut 0 si l'individu i est vivant jusqu'à $t_{(k)}$. Afin d'évaluer la capacité de discriminer des individus décédés au temps $t_{(k)}$ des individus vivants jusqu'au temps $t_{(k)}$, la probabilité $S(t_{(k)}|X)$ de survivre au temps $t_{(k)}$ est la quantité "naturelle" à considérer. Pour toute valeur de $c \in [0, 1]$ définissant une règle de prédiction pour classer les individus à risque au temps $t_{(k)}$, les individus sont décédés au temps $t_{(k)}$ si $S(t_{(k)}|X) \leq c$ et vivants jusqu'au temps $t_{(k)}$ si $S(t_{(k)}|X) > c$. La probabilité de classifier décédé sachant que le statut est décédé au temps $t_{(k)}$ (*i.e.* vrai positif) est appelée sensibilité et est définie par :

$$Se(c, t_{(k)}) = \mathbb{P}(S(t_{(k)}|X) \leq c | D(t_{(k)}) = 1).$$

La probabilité de classifier vivant sachant que le statut est vivant jusqu'au temps $t_{(k)}$ (*i.e.* vrais négatifs) est appelée spécificité et est définie par :

$$Sp(c, t_{(k)}) = \mathbb{P}(S(t_{(k)}|X) > c | D(t_{(k)}) = 0).$$

Pour évaluer la capacité de discrimination de la prédiction $S(t_{(k)}|X)$, l'aire sous la courbe ROC est utilisé et correspond pour deux individus i et j :

$$AUC(t_{(k)}) = \mathbb{P}(S(t_{(k)}, X_i) < S(t_{(k)}, X_{j.})|D_i(t_{(k)}) = 1; D_j(t_{(k)}) = 0).$$

Enfin, pour résumer la capacité de $S(t_{(k)}, X_{i.})$ à discriminer les individus décédés au temps $t_{(k)}$ des individus vivants jusqu'au temps $t_{(k)}$ sur l'ensemble de l'étude, Antolini et al. (2005), a proposé le C_{td} Index (pour *time-dependent discrimination index*). Cette métrique est une moyenne pondérée de l'AUC au cours du temps :

$$\begin{aligned} C_{td} &= \frac{\sum_{k=0}^{K} AUC(t_{(k)})w(t_{(k)})}{\sum_{k=0}^{K} w(t_{(k)})}, \quad \text{avec} \\ w(t_{(k)}) &= \mathbb{P}(D_i(t_{(k)}) = 1; D_j(t_{(k)}) = 0). \end{aligned} \tag{1.36}$$

$w(t_{(k)})$ est la probabilité que les individus i et j soient comparables au temps $t_{(k)}$, c'est-à-dire que l'individu i soit décédé au temps $t_{(k)}$ et l'individu j soit vivant jusqu'au temps $t_{(k)}$. On suppose qu'on a un échantillon de n individus et on se place dans le cas d'un modèle de survie.
$\{(t_1, \delta_1, S(t_{(k)}, X_1); k = 1, \ldots, K), \ldots, (t_n, \delta_n, S(t_{(k)}, X_n); k = 1, \ldots, K))\}$ sont respectivement les temps observés, l'indicateur de censure et la fonction de survie prédite. Pour $i, j = 1, \ldots, n\, i \neq j$, on définit les indicateurs :

$$\begin{aligned} comp_{ij} &= \mathbb{1}_{\{(t_i<t_j;\delta_i=1)\cup(t_i=t_j;\delta_i=1,\delta_j=0)\}} \quad \text{et} \\ conc_{ij}^{td} &= \mathbb{1}_{\{S(t_i|X_{i.})<S(t_j|X_j)\}} comp_{ij}. \end{aligned}$$

On peut remarquer que dans le cas où la relation (1.33) n'est pas vérifiée, $comp_{ij}$ est défini de la même manière que dans le cas où (1.33) est vérifiée. L'estimation du C-index pour les modèles de survie ne vérifiant pas (1.33) est égale à :

$$\widehat{C}_{td} = \frac{\sum_{i=1}^{n} \sum_{j \neq i} conc_{ij}^{td}}{\sum_{i=1}^{n} \sum_{j \neq i} comp_{ij}} \tag{1.37}$$

Si la relation (1.33) est vérifiée, la métrique $\widehat{C}_{td}$ de l'équation (1.37) est équivalente à la métrique $\widehat{C}_i$ de l'équation (1.35).

1.3.2 Score de Brier Intégré

Afin de présenter le score de Brier Intégré, je vais d'abord introduire la définition de la métrique score de Brier (Brier, 1950).

1 Score de Brier

Le score de Brier, développé initialement pour évaluer les prévisions météorologiques par Brier (1950), mesure l'erreur quadratique entre l'indicatrice $\mathbb{1}_{\{T \geq t_i\}}$ de survivre jusqu'au temps t_i et sa prédiction par le modèle $\widehat{S}(t|X_{i.})$. Graf et al. (1999) ont adapté le score de Brier (Brier, 1950) pour les données de survie censurées en utilisant la probabilité inverse des poids censurés (*Inverse Probability of Censoring Weights (IPCW)*) et Gerds et al. (2006) ont proposé par la suite un estimateur consistant du score de Brier en présence de

données censurées. Le score de Brier est défini par :

$$BS(t, \widehat{S}) = \mathbb{E}\left[\left(Y_i(t) - \widehat{S}(t|X_{i.})\right)^2\right], \tag{1.38}$$

où $Y_i(t) = \mathbb{1}_{\{T_i \geq t\}}$ est le statut de l'individu i au temps t et $\widehat{S}(t|X_{i.})$ est la probabilité de survie prédite au temps t pour l'individu i. Contrairement à l'AUC ou au C-index, une valeur faible de ce score montre une bonne capacité de prédiction du modèle. Plus la valeur du score de Brier est proche de 0, meilleure est la prédiction. Comme ce score correspond à une erreur quadratique, il est possible de le décomposer sous la forme d'un terme de biais et d'un terme de variance :

$$BS(t, \widehat{S}) = \mathbb{E}\left[\left(\mathbb{E}[Y_i(t)|X_{i.}] - \widehat{S}(t|X_{i.})\right)^2\right] + \mathbb{E}\left[\left(Y_i(t) - \mathbb{E}[Y_i(t)|X_{i.}]\right)^2\right]. \tag{1.39}$$

Le premier terme est appelé terme de calibration, il mesure la différence entre la prédiction de survivre et la "vraie" probabilité de survivre sachant les variables X. Le second terme est appelé inséparabilité et capture la capacité de discrimination du modèle. Pour évaluer un modèle prédictif, comme évoqué dans le premier paragraphe, il est important d'analyser son pouvoir de discrimination et de calibration. Le score de Brier est donc intéressant car il permet par sa décomposition de mesurer les performances globales d'un modèle prédictif.

2 Score de Brier dans le cadre de la censure

Comme évoqué ci-dessus, Gerds et al. (2006) donnent une estimation du score de Brier en présence de données de survie censurées. L'estimation du score de Brier en absence de censure est :

$$\widehat{BS}(t, \widehat{S}) = \frac{1}{n}\sum_{i=1}^{n}\left[Y_i(t) - \widehat{S}(t|X_{i.})\right]^2, \tag{1.40}$$

avec $Y_i(t) = \mathbb{1}_{\{T_i \geq t\}}$ le statut de l'individu i au temps t, $\widehat{S}(t|X_{i.})$ la probabilité de survie prédite au temps t pour l'individu i et n le nombre d'individus dans l'ensemble de test. L'ensemble de test est une partie du jeu de données qui n'a pas été utilisée pour estimer les paramètres du modèle et permet de calculer la performance du modèle. Si les données sont censurées à droite, nous ne pouvons plus utiliser le statut de l'individu i. Le statut de l'individu i est donc remplacé par le statut observé de l'individu i : $\widetilde{Y}_i(t) = \mathbb{1}_{\{\widetilde{T}_i > t\}}$. De plus, en présence de données censurées il est nécessaire d'ajuster le score en le pondérant par la probabilité inverse des poids censurés (*Inverse Probability of Censoring Weights (IPCW)*). Cette pondération est définie par :

$$\widehat{W}_i(t) = \frac{(1 - \widetilde{Y}_i(t))\delta_i}{\widehat{G}(\widetilde{Y}_{i-}|X_{i.})} + \frac{\widetilde{Y}_i(t)}{\widehat{G}(t|X_{i.})}, \tag{1.41}$$

où $\widehat{G}(t|x)$ est l'estimateur de Kaplan-Meier (Kaplan et al., 1958) de la fonction de survie des temps censurés au temps t. L'estimation du score de Brier dans le cadre de la censure à droite est :

$$\widehat{BS}(t, \widehat{S}) = \frac{1}{n}\sum_{i=1}^{n}\widehat{W}_i(t)(\widetilde{Y}_i(t) - \widehat{S}(t|X_{i.}))^2, \tag{1.42}$$

avec $\widetilde{Y}_i(t)$ le statut observé de l'individu i, $\widehat{S}(t|X_{i.})$ la probabilité de survie prédite au temps t pour l'individu i et n le nombre d'individus dans l'ensemble de test.

3 Score de Brier intégré

Le score de Brier intégré (Mogensen et al., 2012) permet de résumer la performance prédictive estimée par le score de Brier (Brier, 1950) :

$$\widehat{IBS} = \frac{1}{\tau}\int_0^\tau \widehat{BS}(t, \widehat{S})dt, \tag{1.43}$$

où $\widehat{BS}(t, \widehat{S})$ est le score de Brier estimé et $\tau > 0$. On fixe $\tau > 0$ qui peut être par exemple le maximum des temps observés et le score de Brier est moyenné sur l'intervalle $[0, \tau[$. En pratique, nous utilisons la méthode des trapèzes pour calculer le Score de Brier intégré. Cette méthode calcule une approximation de l'intégral d'une fonction f, en couvrant l'aire sous le graphe par une collection de trapèzes dont les bases sont déterminées par les valeurs de la fonction. L'intervalle $[0, \tau]$ est subdivisé en n sous-intervalles $[t_i, t_{i+1}]$ où $0 = t_0 < t_1 < \ldots < t_n = \tau$ de longueurs τ/n. L'approximation de l'intégrale est alors calculée à partir de la formule :

$$\int_0^\tau f(x)dx \approx h\left(\frac{f(0) + f(\tau)}{2} + \sum_{i=1}^{n-1} f\left(t_i\right)\right)$$

avec $h = \tau/n$ et $t_i = ih$. Comme pour le score de Brier, une valeur du score de Brier intégré proche de zéro indique que les performances prédictives du modèle sont bonnes.

Chapitre 2

Sélection de variables

2.1 Sélection de variables pour la détection de marqueurs

Dans cette section, nous nous intéressons à la sélection de variables pour la détection de marqueurs. Nous allons présenter deux méthodes afin de trouver les gènes avec une influence sur le cancer du rein à cellules claires. Cette section correspond à un travail en collaboration avec l'équipe de Diana Tronik Le Roux (Hôpital Saint Louis/CEA), qui a été à l'origine d'un article publié dans Cancer Immunotherapy, Immunology (Tronik-Le Roux et al., 2020). La partie statistique de cette collaboration a été réalisée conjointement avec Mahmoud Bentriou.

2.1.1 Contexte de l'étude

Le thème de recherche de l'équipe du Dr Diana Tronik Le Roux concerne l'immuno-oncologie. L'équipe s'intéresse à un cancer en particulier, le cancer du rein à cellules claires (ccRCC). Celui-ci est le type de cancer du rein le plus répandu (70% des cancers du rein). L'origine du ccRCC serait due à la perte de régions sur le chromosome 3p, siège de gènes suppresseurs de tumeurs comme *von Hippel-Lindau (VHL)*. Les patients atteints par ce cancer ne peuvent pas être traités par chimiothérapie ou radiothérapie. Dans les années 1980 et 1990, la chimiothérapie a été prescrite avec 10% de réponses complètes (Vano et al., 2015). Mais celle-ci provoquait une sévère toxicité, amenant à l'arrêt de la chimiothérapie. La radiothérapie peut être suggérée quand le cancer est à un stade précoce. Cependant, le diagnostic est rarement fait à ce stade. Il a été ensuite montré que le ccRCC est un cancer immunogène, c'est-à-dire qu'il est capable d'arrêter l'action du système immunitaire. Une attention a été portée vers l'optimisation de thérapies basées sur les *Immune Checkpoints (IC)*. Les *Immune-Checkpoints (IC)* sont des molécules de surface cellulaire qui génèrent des signaux positifs ou négatifs dans les cellules effectrices. Dans le cas des cellules tumorales, ces molécules bloquent l'action de la cellule immunitaire T et l'empêchent de jouer son rôle de destruction de la cellule tumorale. Des traitements avec des anticorps anti-CTLA-4 ou PD-L1 :PD-1 ont prouvé leur efficacité sur la restauration de la réponse anti-tumorale. Ces traitements ont permis d'aider certains patients, mais ils restaient inefficaces sur d'autres. Plusieurs hypothèses ont été émises. L'une d'entre elles concerne la complexité des voies de signalisation des IC. Beaucoup d'IC ne sont pas encore décrits, à part CTLA-4 et PD-1 :PD-L1, et pourraient être exprimés simultanément et agir de façon concomitante. Une autre hypothèse de l'inefficacité des traitements chez certains patients concerne l'hétérogénéité de l'expression à l'intérieur de la cellule tumorale. Une dernière hypothèse évoque la toxicité due à l'action des anticorps sur les cellules saines. Dans ce contexte, la recherche de nouvelles cibles IC s'est intensifiée (Sharma et al., 2015 ; Marin-Acevedo et al., 2018). Le travail réalisé lors de cette collaboration a été de fournir une étude poussée de 44 IC dans le cancer du rein (ccRCC) et de mettre en évidence les IC caractérisants le ccRCC ainsi que ceux discriminant le plus les cellules tumorales

des cellules saines.
Nous avons réalisé une analyse différentielle, qui consiste à trouver les gènes différentiellement exprimés entre deux conditions (par exemple : cellule malade/cellule saine). Cette approche est couramment effectuée dans ce contexte. D'autre part, nous nous sommes intéressés à une approche plus originale. Nous avons utilisé une méthode d'apprentissage statistique afin de détecter les gènes ayant le plus d'impact pour discriminer la cellule tumorale de la cellule saine. Ces deux méthodes ont été appliquées sur les données RNA-seq du ccRCC provenant de la base de données publique *The Cancer Genome Atlas (TCGA)* accessible à `https://www.cancer.gov/tcga`. Dans cette base de données, nous avons téléchargé les données RNA-seq de 539 patients atteints du ccRCC. Pour chacun de ces patients, nous avons eu accès à 25 283 gènes. Ces données ont été normalisées par la méthode FPKM-UQ *(Fragments Per Kilobase of transcript per Million mapped reads Upper Quartile)* qui est le ratio entre le nombre de *reads* (un *read* est une lecture d'ADN issue d'un séquenceur) alignés au gène et le 75e percentile des expressions de l'échantillon. Dans un premier temps, une analyse différentielle a été exécutée sur l'ensemble des données RNA-seq. Ensuite, la méthode *Recursive Feature Elimination* (détaillée à la section 2.1.3) a été appliquée sur les données RNA-seq contenant seulement les ICs considérés différentiellement exprimés lors de la première étape.

2.1.2 Analyse différentielle

De nombreux outils statistiques ont été développés pour analyser les données RNA-seq (Anders et al., 2010; Love et al., 2014; Smyth, 2005). La comparaison de ces différents outils a montré que DESeq2 et edgeR étaient les plus conservatifs (Soneson et al., 2013; Rapaport et al., 2013; Seyednasrollah et al., 2015). L'analyse différentielle a été exécutée sur les données RNA-seq en utilisant DESeq2 (Love et al., 2014). C'est une méthode implémentée dans un package R, qui permet d'identifier les gènes différentiellement exprimés entre deux conditions. Dans notre étude, le but est d'identifier les IC différentiellement exprimés entre la cellule tumorale et la cellule saine chez des patients atteints du ccRCC. Une grande variabilité existe entre les patients dans les données RNA-seq. Pour prendre en compte cette caractéristique, il est nécessaire d'utiliser une distribution pour modéliser ces données, capable de gérer cette sur-dispersion. La loi binomiale négative est une distribution avec deux paramètres. Le premier paramètre correspond à celui de la moyenne et le second paramètre permet de gérer la sur-dispersion. DESeq2 utilise un test de Wald basé sur un modèle binomial négatif pour tester si l'expression de chaque gène est similaire entre les cellules tumorales et les cellules saines chez les patients ccRCC. Love et al. (2014) note K_{ij} les comptages du gène i dans un échantillon j et modélise K_{ij} par une loi binomiale négative de moyenne μ_{ij} et de dispersion α_i. La dispersion α_i est calculée sur l'ensemble des échantillons.

$$\begin{aligned} K_{ij} &\sim NB(mean = \mu_{ij}, dispersion = \alpha_i) \\ \mu_{ij} &= s_{ij} q_{ij} \end{aligned} \tag{2.1}$$

L'estimation de la dispersion est réalisée en trois étapes.

1. La première étape consiste à utiliser les données de comptage pour chaque gène séparément afin d'obtenir l'estimateur préliminaire de la dispersion au niveau du gène α_i^{GW} par estimation du maximum de la vraisemblance ajustée de Cox-Reid :

$$\alpha_i^{GW} = \arg\max_{\alpha} \left\{ \sum_j \log f_{NB}(K_{ij}; \mu_{ij}, \alpha) \right\}, \tag{2.2}$$

   où f_{NB} est la fonction de densité de la loi Binomiale Négative de moyenne μ et de dispersion α et le second terme permet de prendre en compte l'estimation initiale $\hat{\mu}_{ij}^0$.

2. La seconde étape consiste à calculer la tendance de la dispersion α_{tr} en traçant l'estimation de la dispersion du gène α_i^{GW} en fonction des moyennes des comptages normalisés $\bar{\mu}$. α_{tr} est une fonction de la moyenne des comptages normalisés d'un gène :

$$\alpha_{tr_i}(\bar{\mu}_i) = \frac{a_1}{\bar{\mu}_i} + \alpha_0, \tag{2.3}$$

où $\bar{\mu}_i = \frac{1}{m}\sum_j \frac{K_{ij}}{s_{ij}}$ et a, α_0 sont des hyperparamètres obtenus par un modèle linéaire généralisé gamma.

3. La troisième consiste à estimer le paramètre de dispersion par maximum *a posteriori* (la log-vraisemblance ajustée à laquelle on ajoute le logarithme du prior) :

$$\alpha_i^{MAP} = \arg\max_{\alpha} \left\{ \sum_j \log f_{NB}(K_{ij}; \mu_{ij}, \alpha) + \Lambda_i(\alpha) \right\},$$

où $f_{NB}(k; \mu, \alpha)$ est la fonction de densité de la loi Binomiale Négative avec de moyenne μ et dispersion α et $\Lambda_i(\alpha)$ est le logarithme de la densité du prior $\log \alpha_i \sim \mathcal{N}(\log \alpha_{tr_i}, \sigma_{id}^2)$:

$$\Lambda_i(\alpha) = \frac{-\left(\log \alpha - \log \alpha_{tr_i}(\mu)\right)^2}{2\sigma_{id}^2},$$

et σ_{id} est l'écart-type du prior décrivant le niveau de dispersion des véritables dispersions des gènes individuels autour de la tendance. Son maximum (la valeur du Maximum *A Posteriori*) est utilisé comme estimation finale de la dispersion.

Love et al. (2014) postule un prior normal centré en zéro pour les coefficients β_{ir} du modèle :

$$\log_2(q_{ij}) = \sum_r X_{jr}\beta_{ir},$$

qui représentent le logarithme en base 2 du *fold change* (appelé *log fold-change*) et la matrice X_{jr} indique la condition r (sain ou malade) d'un échantillon j. L'estimation des β_{ir}, *i.e.* le log fold-change, est obtenue en calculant la somme de la log-vraisemblance de (2.1.2) et le logarithme de la densité du prior $\beta_{ir} \sim \mathcal{N}(0, \sigma_i r^2)$:

$$\begin{aligned} \beta_i &= \arg\max_{\beta} \left\{ \sum_j \log(f_{NB}(K_{ij}, \mu_j(\beta), \alpha_i) + \Lambda(\beta) \right\} \\ \text{où } \mu_j(\beta) &= s_{ij} \exp\left(\sum_r X_{jr}\beta_r \right) \text{ et } \Lambda(\beta) = \sum_r \frac{-\beta_r^2}{2\sigma_r^2}, \end{aligned} \tag{2.4}$$

α_i est l'estimation finale de la dispersion pour le gène i, c'est-à-dire $\alpha_i = \alpha_i^{MAP}$ et σ_{ir} sont les écart-types du prior pour les coefficients du modèle. Les coefficients estimés β_i indiquent la force d'expression des gènes. Un test de Wald est ensuite utilisé pour tester si les coefficients estimés diffèrent significativement de zéro. Le test de Wald consiste à utiliser l'estimation des coefficients β_{ir} pour construire la statistique de test :

$$T_{stat_{ir}} = \frac{\beta_{ir} - \beta_{ir}^{\star}}{SE(\beta_{ir})},$$

où $\beta^{\star}_{ir}$ est la valeur testée du coefficient qu'on aimerait rejeter (*i.e.* zéro). Le test de Wald compare donc l'estimation des coefficients divisée par l'erreur standard estimée $(SE(\beta_{ir}))$ à la distribution normale. L'erreur standard estimé est la racine carré de la matrice des covariances pour les coefficients. Les p-valeurs du test de Wald du sous-ensemble de gènes passant l'étape de filtrage indépendant (c'est-à-dire que leur p-valeur est supérieure au seuil de première espèce α) sont ajustées pour les tests multiples utilisant la procédure de Benjamini-Hochberg (Benjamini et al., 1995). La procédure de Benjamini-Hochberg (Benjamini et al., 1995) est utilisée pour contrôler le taux des faux positifs. L'algorithme général pour l'estimation des p-valeurs ajustées à partir d'une liste de p-valeurs est :

1. Soient $p_{(1)} \leq p_{(2)} \leq \ldots \leq p_{(m)}$ les p-valeurs ordonnées et notons $H_0^{(i)}$ l'hypothèse nulle correspondant à $p_{(i)}$.
2. Calculer le nombre de tests i^* tel que :

$$i^* = \arg\max_i \left(p_{(i)} \leq \frac{i}{m}\alpha \right)$$

   et α est le seuil de première espèce.
3. Rejeter tous les $H_0^{(j)}$ tels que $j \leq i^*$ et la p-valeur ajustée est calculée de la manière suivante :

$$\widetilde{p_{(i)}} = \frac{m}{rank(p_{(i)})} p_{(i)}.$$

Les gènes dont la p-valeur ajustée est inférieure à 0.05 sont considérés comme différentiellement exprimés.

2.1.3 Recursive Feature Elimination

L'apprentissage supervisé est un outil puissant pour l'analyse statistique de données biologiques. Dans cette section, nous voulons détecter parmi les gènes différentiellement exprimés ceux qui sont capables de discriminer un tissu sain d'un tissu tumoral. Notre objectif est de classer les IC en fonction de leur puissance de prédiction : nous voulons trouver le meilleur ensemble de gènes réalisant la tâche de classification (tissu sain ou tissu tumoral). Pour cela, nous nous intéressons à la méthode *Recursive Feature Elimination (RFE)* (Guyon et al., 2002) avec un modèle machine à vecteurs de support (*Linear-Support Vector Machine (linear-SVM)*). La méthode RFE sélectionne récursivement en considérant des ensembles de variables de plus en plus petits et assigne des poids aux variables directement liés aux coefficients du modèle linéaire. Un ensemble des gènes ordonnés est sélectionné en fonction du critère SVM-IC (Claeskens et al., 2008). Les détails de la méthode sont expliqués ci-dessous.

1 Linear Support Vector Machine training

Nous nous concentrons sur un modèle de classification binaire. Nous voulons construire un classifieur f :

$$\begin{aligned} f : \mathbb{R}^p &\to \{-1, 1\} \\ (X_1, \ldots, X_p) &\to y \end{aligned}$$

où $(X_1, \ldots, X_p)$ est l'expression des gènes d'un échantillon de tissu et la valeur cible y est le type du tissu : sain ou tumoral. Une machine à vecteur support avec un noyau linéaire est choisi pour l'algorithme de classification. Nous définissons le dataset $\{(X_{i.}, y_i)\}_{1 \leq i \leq n}$ où $X_{i.} = (X_{i1}, \ldots, X_{ip})$ est l'expression des gènes d'un individu i and y_i son label (tissu sain ou tissu tumoral). Pour chaque gène $j \in \{1, \ldots, p\}$, un poids

w_j est calculé pour la tâche de classification. Ce poids est utilisé pour ordonner les gènes. La méthode d'apprentissage SVM-linéaire peut être définie comme un problème d'optimisation :

$$\begin{aligned} &\arg\min_{w,b,\epsilon} \frac{1}{2}||w||^2 + C\sum_{i=1}^{n}\epsilon_i \\ \text{sous la contrainte } &\forall i \in \{1,\dots,n\}, y_i(w^t X_{i.} + b) \geq 1 - \epsilon_i \\ &\forall i \in \{1,\dots,n\}, \epsilon_i \geq 0. \end{aligned}$$

2 Recursive Feature Elimination (RFE)

La méthode RFE supprime récursivement les gènes qui sont les moins importants dans la tâche de classifcation. À une itération $k \in \{1,\dots,p\}$, il reste $p-k$ gènes à ordonner. Le modèle SVM-linéaire est entraîné sur tous les gènes et le gène avec la plus petite importance $c_j = |w_j|^2$ dans la tâche de classification est supprimé. Soit X la matrice du jeu de données $\{(X_{i.}, y_i)\}_{1\leq i\leq n}$ (où i est le patient) et soit $y = (y_i)_i$ le vecteur des labels. L'algorithme RFE est détaillé dans Algorithm 1.

Algorithm 1 RFE algorithm

Require: X, y
Ensure: A set of ranked genes $G = [gene_1, \dots, gene_p]$
  Initialization : $S \leftarrow [1,\dots,p], G \leftarrow 0$
  while $S \neq \emptyset$
    $X_{current} \leftarrow X[:, S]$
    Get w from linear-SVM training on the dataset $X_{current}, y$
    $j_{\min} \leftarrow \arg\min_{j\in S} w_j^2$
    $S \leftarrow S \backslash \{j_{\min}\}$
    $G \leftarrow concat(j_{\min}, G)$
  End while

L'ensemble des gènes ordonnés G est ainsi obtenu. Ensuite, les classifieurs entraînés sur un sous-ensemble de variables G_k sont comparés dans le but d'obtenir le meilleur sous-ensemble de gènes :

$$G_k = [gene_1, \dots, gene_k], \forall k \in \{1,\dots,p\},$$

où $gene_1$ est le gène le plus important et $gene_p$ le moins important. Pour comparer ces classifieurs, un critère d'information est utilisé (Claeskens et al., 2008). Celui-ci peut être vu comme un critère équivalent du critère d'information d'Akaike pour les modèles SVM-linéaire. Dans le cadre de l'apprentissage du modèle SVM-linéaire, le critère d'information SVM (SVM-IC) est défini par :

$$SVMIC(G_k) = \sum_{i=1}^{n}\epsilon_i + 2Card(G_k),$$

où ϵ_i est obtenu par l'apprentissage du SVM-linéaire et $Card(G_k)$ est la longueur du sous-ensemble. Dans ce cadre, $Card(G_k) = k$. Plus le critère est petit, meilleur est le classifieur. Le meilleur sous-ensemble de variables est :

$$G_{k_{best}} = \arg\min_{G_k} SVMIC(G_k).$$

Ce sous-ensemble est le meilleur compromis entre la performance de classification et la réduction du sous-ensemble dans le but d'obtenir les gènes les plus significatifs pour discriminer la cellule tumorale de la cellule saine.

2.1.4 Résultats de l'étude

1 Contexte des IC exprimés dans le ccRCC

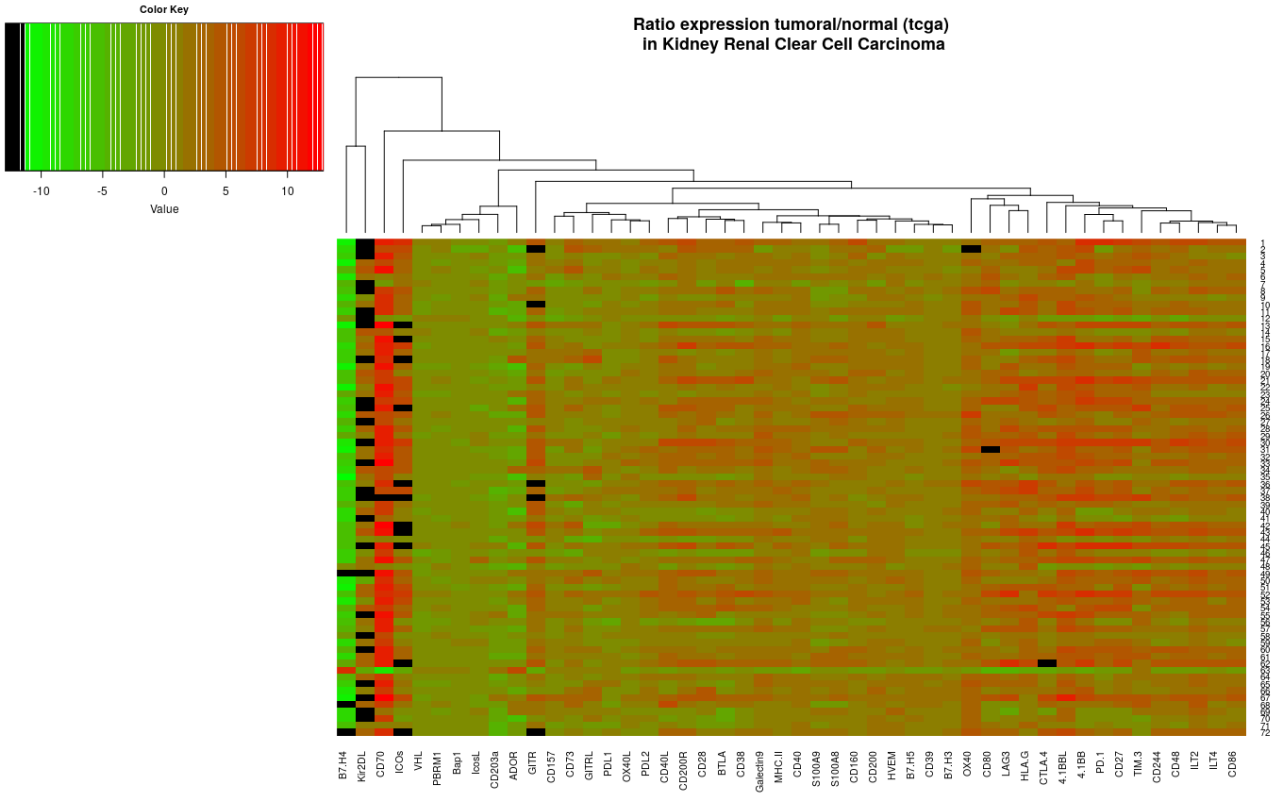


Figure 2.1 – Représentation sur la heatmap du logarithme du rapport des expressions des gènes entre le tissu tumoral et le tissu sain. Chaque colonne représente un gène et chaque ligne un patient. Les valeurs en noires représentent un rapport égal à 0 (dont le logarithme vaut $-\infty$).

L'étude s'est concentrée sur quarante quatre IC en se basant sur la revue (Pardoll, 2012). Trois gènes suppresseurs de tumeurs (von Hippel-Lindau (VHL), Polybromo 1 (PBRM1) et BRCA1-associated protein (BAP1)) ont été ajoutés comme contrôles. Ces derniers sont connus pour être exprimés à un faible niveau dans la plupart des ccRCC. Tout d'abord, le rapport des expressions de gènes entre le tissu tumoral et le tissu sain a été représenté sur une *heatmap* (cf. Figure 2.1). Dans cette visualisation, chaque colonne représente un gène et chaque ligne un patient. La Figure 2.1 montre une faible différence entre les cellules saines et tumorales des patients. Nous pouvons également observer que les gènes contrôles PBRM1, VHL et BAP1, sous-exprimés dans ccRCC, sont groupés ensemble et semblent faiblement exprimés sur la Figure 2.1.

2 Analyse différentielle des IC

Pour établir des résultats statistiquement interprétables, nous avons conduit une analyse différentielle afin de comparer les niveaux d'expression des IC dans les tissus tumoraux *versus* les tissus sains en utilisant la méthode DESeq2 (Love et al., 2014). Cette méthode a été appliquée sur l'ensemble des gènes. Un ensemble de 1 264 gènes ont été trouvés sur-exprimés et 1 194 gènes ont été trouvés sous-exprimés. À partir de cela, nous avons extrait l'information des expressions pour les 44 IC. Les résultats obtenus sont représentés sur la Figure 2.2. L'axe des abscisses représente la valeur du *log fold change* (défini dans la section 2.1.2) et l'axe des ordonnées représente les gènes. De plus, les gènes sur l'axe des ordonnées sont rangés selon leur valeur de p-valeur ajustée. Plus la p-valeur ajustée est faible, plus la position du gène associé sur l'axe est basse et plus il est significativement exprimé. Un gradient de couleurs est utilisé afin d'informer si le niveau d'expression du gène dans l'étude est significatif. Enfin, la Table A.1 en Annexe A résume de façon plus précise les résultats des IC obtenus à partir de la méthode DESeq2 (Love et al., 2014). Parmi les gènes les plus différentiellement sur-exprimés dans la cellule tumorale, nous trouvons ceux encodants les paires ligand-récepteur tels que : *CD70-CD27; HLA.G-ILTs; 4.1BB-4.1BBL, CD40-CD40L; CD86-CTLA4; MHC.II-Lag3; CD200-CD200R; CD244-CD48*. En revanche, des niveaux d'expression significativement plus hauts de $B7.H4$ sont présents dans la cellule saine comparé à la cellule tumorale. Comme il est attendu, les niveaux d'expression des trois gènes contrôles : $PBRM1$, VHL et $BAP1$ sont plus faibles dans le tissu tumoral que dans le tissu sain.

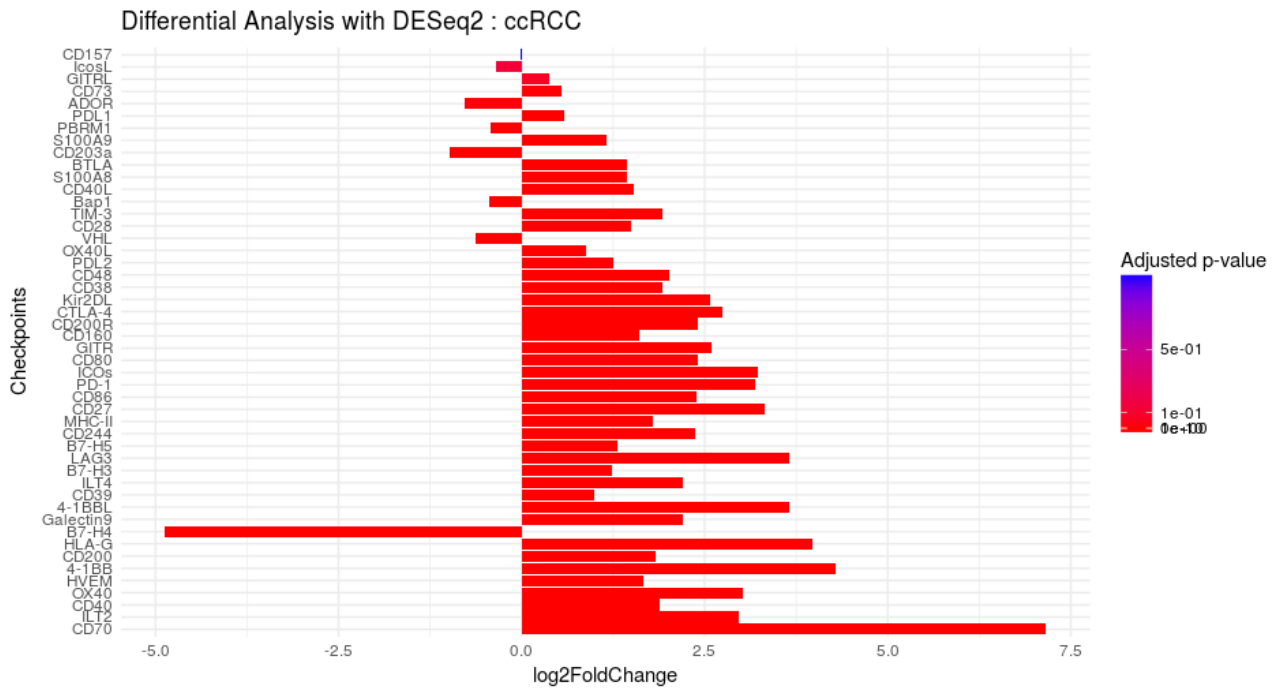
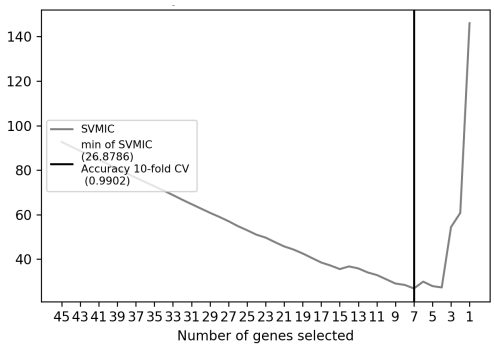


Figure 2.2 – Catégorisation des 44 IC différentiellement exprimés. L'axe des abscisses représente la valeur du *logarithme fold change*. Si cette valeur est négative alors le gène est sous-exprimés et si elle est positive alors le gène est sur-exprimé. Plus la valeur s'éloigne de 0, plus la force d'expression est élevée dans la cellule tumorale comparé à la cellule saine. Pour représenter la p-valeur ajustée pour chaque gène, un gradient de couleurs est utilisé. Plus la barre est rouge, plus l'IC est considéré comme différentiellement exprimé.

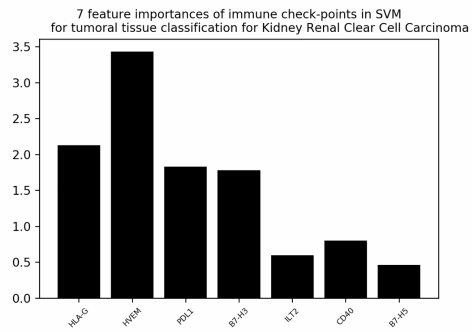
3 Détection des IC exprimés les plus importants dans ccRCC

Notre analyse précédente des IC exprimés dans le ccRCC révèle la sur-expression de 38 IC. Afin de détecter lesquels sont les plus représentatifs de ce type de cancer, nous avons réalisé une *Recursive Feature Elimination*

(RFE) avec un SVM-linéaire. La méthode estime les gènes les moins importants pour la classification, en les supprimant de manière itérative un par un jusqu'à ce que l'ensemble des gènes soit vide. Les gènes sont alors ordonnés en fonction de leur moment de suppression de l'ensemble par la méthode. Celle-ci permet de trouver le meilleur sous-ensemble de gènes et a l'avantage de pouvoir être appliquée sur l'ensemble du jeu de données sans fixer un seuil en amont. La méthode RFE a donc été appliquée sur les 539 patients présents dans la base de données TCGA, suivi par le modèle SVM-linéaire. Les résultats, nommés SVMIC, ont révélé que le sous-ensemble de gènes optimal était composé de 7 IC. La précision est visible sur la Figure 2.3a. Elle a été estimée par une validation croisée sur l'ensemble du jeu de données. La validation croisée consiste à couper le jeu de données en partitions. Ici, le jeu de données est séparé en dix parties. La méthode est entraînée sur neuf parties et la précision est calculée pour la partie restante. Cela est effectué autant de fois que de partitions. La moyenne des dix valeurs de précision obtenues est calculée. Le sous-ensemble de gènes obtenu par la méthode est : *HLA-G, HVEM, PD-L1, B7-H3, ILT2, CD40, B7-H5*. L'importance de ces gènes dans la discrimination des cellules tumorales et saines est représentée sur la Figure 2.3b. L'axe des abscisses représente les gènes sélectionnés par la méthode, l'axe des ordonnées l'importance des gènes dans le modèle SVM-linéaire. Nous pouvons observer sur la Figure 2.3b que *HLA-G* est le IC le plus important pour discriminer la cellule tumorale et la cellule saine en considérant l'ordre des gènes sélectionnés. Nous pouvons également voir sur la Figure 2.3b que sa valeur d'importance n'est pas la plus haute, ce qui pourrait signifier que *HLA-G* partage l'information avec un ou plusieurs des gènes sélectionnés.



(a) Graphe des critères d'information SVM en fonction des sous-ensembles générés par l'algorithme RFE. L'axe des abscisses représente le nombre de gènes sélectionnés et l'axe des ordonnées représente le score de la validation croisée.



(b) Barplot montrant l'importance de chacun des sept variables sélectionnées. L'axe des abscisses représente les gènes sélectionnés et l'axe des ordonnées l'importance des gènes dans le modèle SVM-linéaire.

Figure 2.3 – Résultats de la méthode RFE avec un modèle SVM-linéaire sur les IC différentiellement exprimés

4 Discussion

À partir de cette étude, *i.e.* l'analyse de transcriptome ainsi que les analyses biologiques réalisées par l'équipe du Dr Diana Tronik Le Roux, des IC clés ont pu être mis en évidence. Certains de ces IC n'avaient jamais été reportés pour être sur-exprimés dans le cancer du rein à cellules claires. De plus, l'analyse originale de l'importance des IC a montré une existence possible de plusieurs IC s'exprimant simultanément dans les cellules tumorales. Ce résultat donne un nouvel éclairage sur les cibles potentielles pour traiter le ccRCC. Enfin, l'équipe du Dr Diana Tronik Le Roux évoque que cibler HLA-G/ILT serait une stratégie intéressante dans le cas de non-réponse à anti-PD1/PDL-1 au vu des résultats obtenus.

2.2 Sélection de variables pour la survie

Dans cette section, notre objectif est de sélectionner les variables les plus pertinentes pour expliquer la survie des patients. Comme évoqué en section 1.1, le nombre de variables considérées dans le cadre de la survie est de plus en plus important mais la taille de l'échantillon n'a pas augmenté. Nous sommes donc dans un cadre de grande dimension et la procédure classique d'estimation des paramètres n'est plus valide. L'estimateur des paramètres $\hat{\beta}$ de l'équation (1.20) n'est plus consistant. Les paramètres β traduisent le poids des variables explicatives $(X_1, X_2, \ldots, X_p)$ sur la durée de survie T. Connaître la contribution de chaque variable devient difficile en grande dimension. La sélection de variables consiste à déterminer les coefficients non-nuls dans β. Pour estimer β, la méthode usuelle quand peu de variables sont disponibles est la maximisation de la log-vraisemblance partielle de Cox. Mais cette maximisation amène toujours à choisir le plus grand modèle et cela pose problème quand le nombre de variables devient important. La vraisemblance croît avec le nombre de variables. De plus, beaucoup de variables seront sélectionnées quand le nombre de variables est grand et le modèle ne sera pas facilement interprétable. Enfin, le problème d'optimisation ne peut également pas être résolu en grande dimension. La solution consiste donc à minimiser l'opposé de la log-vraisemblance partielle de Cox (Cox, 1975) en y ajoutant un terme de pénalisation. L'ajout de ce terme de pénalisation permet de résoudre le problème d'optimisation en incitant à choisir un modèle plus petit et également plus facilement interprétable pour la grande dimension. De nombreuses pénalités existent avec différentes propriétés d'interprétabilité. Nous renvoyons à Bickel et al. (2006) pour plus de détails sur le concept de pénalisation. Ces fonctions pénalisées sont également appelées méthodes de régularisation, nous utiliserons ce terme dans l'ensemble du manuscrit et nous présentons en section 2.2.1 plusieurs d'entre-elles : Lasso (Tibshirani, 1996; Tibshirani, 1997), Ridge (Hoerl et al., 1970; Verweij et al., 1994), Elastic-Net (Zou et al., 2005) et Adaptive-Lasso (Zou, 2006; Zhang et al., 2007). Cependant, il a été montré que ces méthodes de régularisation pouvaient être instables en sélection de variables (Michiels et al., 2005). D'autres méthodes sont donc apparues, elles sont appelées *Screening*. L'idée générale de ces méthodes est de faire une pré-sélection avant d'appliquer une procédure de régularisation comme le Lasso (Tibshirani, 1997) par exemple. Ces méthodes de *Screening* diffèrent par la pré-sélection utilisée, nous présentons en section 2.2.2 les méthodes SIS (Fan et al., 2010b) et ISIS (Fan et al., 2010b), CoxCS (Hong et al., 2018) et PSIS (Zhao et al., 2012).

2.2.1 Méthodes de régularisation

Les méthodes de régularisation ont d'abord été implémentées dans un cadre linéaire et ont été adaptées à l'analyse de survie en utilisant la vraisemblance partielle de Cox (Zou et al., 2005; Tibshirani, 1997). La plus connue et la plus étudiée des méthodes de régularisation est le Lasso (Tibshirani, 1997) (pour *Least absolute shrinkage and selection operator*) car elle permet une optimisation convexe et une interprétabilité en sélection de variables. Mais elle n'est pas consistante en sélection. D'autres pénalités sont donc dérivées de cette dernière. Dans les sections suivantes, nous présentons le Lasso et d'autres alternatives comme la procédure Ridge (Hoerl et al., 1970; Verweij et al., 1994), Elastic-Net (Zou et al., 2005) et Adaptive-Lasso (Zou, 2006; Zhang et al., 2007).

1 Lasso

La procédure Lasso a d'abord été introduite dans le cadre d'un modèle de régression linéaire par Tibshirani (1996) et ensuite dans le cadre de l'analyse de survie par Tibshirani (1997). Cette procédure est classique en

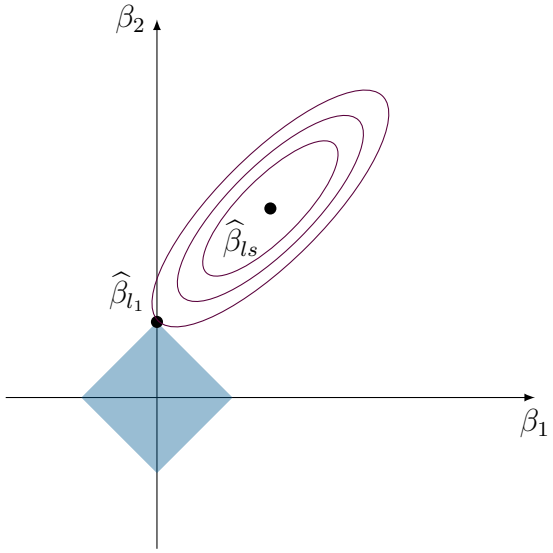


Figure 2.4 – Illustration de la pénalisation l_1

grande dimension et est donc la plus connue et la plus utilisée. Sa pénalisation se présente sous la forme :

$$\begin{aligned} pen(\beta) &= \lambda ||\beta||_1 \\ &= \lambda \left(\sum_{j=1}^{p} |\beta_j| \right). \end{aligned} \tag{2.5}$$

L'estimateur Lasso du paramètre β est obtenu en considérant le problème suivant :

$$\widehat{\beta}_{l_1} = \arg\min_{\beta} \left\{ -\mathcal{L}(\beta) + \lambda \sum_{j=1}^{p} |\beta_j| \right\}, \tag{2.6}$$

où $\mathcal{L}(\beta)$ est la log-vraisemblance partielle de Cox. Ce problème d'optimisation est convexe en β et permet donc d'utiliser les algorithmes d'optimisation convexe pour l'estimation des β. Le problème d'optimisation est donc équivalent à un problème de minimisation de la log-vraisemblance partielle de Cox (Cox, 1972) en y ajoutant une contrainte du type :

$$\sum_{j=1}^{p} |\beta_j| \leq s,$$

avec $s \in \mathbb{R}^+$. Cela revient à contraindre β à être dans une boule de norme l_1 de rayon s dans $\mathbb{R}^p$. L'estimateur $\widehat{\beta}$ obtenu est alors sparse, c'est-à-dire qu'un certain nombre de coefficients de $\widehat{\beta}$ sont nuls. Cela lui confère une interprétabilité en sélection de variables qui est notre objectif dans ce chapitre. La Figure 2.4 illustre la capacité de la pénalité l_1 à obtenir un estimateur sparse. Sur la Figure 2.4, $\widehat{\beta}_{ls}$ représente l'estimeur des β non contraint obtenu par les moindres carrés. Les ellipses violettes représentent les contours de la fonction de perte quadratique autour de l'estimateur $\widehat{\beta}_{ls}$ et le carré bleuté correspond à la partie admissible pour l'estimateur Lasso (soit sur la Figure 2.4, $\{\beta \in \mathbb{R}^2, ||\beta||_1 \leq s\}$). Cette figure illustre l'estimateur obtenu sous la contrainte l_1, quand l'estimateur des moindres carrés $\widehat{\beta}_{ls}$ n'appartient pas à la zone admissible. Cet estimateur est noté $\widehat{\beta}_{l_1}$ sur la Figure 2.4. Nous pouvons observer que la première composante de $\widehat{\beta}_{l_1}$ est annulée car l'ellipse atteint la région admissible sur l'angle situé sur l'axe $\beta_1 = 0$. $\widehat{\beta}_{l_1}$ représente alors l'estimateur lasso obtenu et montre la sparsité induite par la pénalité puisque la première composante de l'estimateur est nulle.

La procédure Lasso (Tibshirani, 1997) se résume à la minimisation de l'opposé de la log-vraisemblance partielle de Cox en y ajoutant un terme de pénalité l_1 :

$$\widehat{\beta}_{l_1} = \arg\min_{\beta} \left\{ -\sum_{i=1}^{n} \left(\beta^T X_{i.}\right) - \sum_{i=1}^{n} \delta_i \log \left(\sum_{l \in R_{i.}} \exp\left(\beta^T X_{l.}\right) \right) + \lambda \sum_{j=1}^{p} |\beta_j| \right\},$$

avec $R_{i.}$ les individus à risque au temps $t_{i.}$, δ_i l'indicateur de censure et $\lambda \in \mathbb{R}^+$ un hyperparamètre. Enfin, nous pouvons remarquer que la valeur de l'hyperparamètre λ influence directement le nombre de composantes non-nulles. Une valeur nulle de λ implique que l'estimateur obtenu sera celui des moindres carrés, alors qu'une grande valeur de λ donnera un vecteur nul pour l'estimateur de $\widehat{\beta}_{l_1}$. Le choix de λ est donc important, plusieurs méthodes pour le choix de celui-ci existent mais la plus utilisée est celle de la validation croisée (Tibshirani, 1996).

2 Ridge

La pénalité Ridge a été introduite initialement par Hoerl et al. (1970) et adaptée pour le modèle de Cox par Verweij et al. (1994). Cette pénalité est en norme l_2 :

$$\begin{aligned} pen(\beta) &= \lambda ||\beta||_2^2 \\ &= \lambda \sum_{j=1}^{p} |\beta_j|^2. \end{aligned} \tag{2.7}$$

La méthode de régularisation Ridge, contrairement au Lasso, n'annule pas les coordonnées de l'estimateur $\widehat{\beta}_{l_2}$ de β mais permet de les réduire. Plus la valeur de l'hyperparamètre de λ est élevée, plus la contraction de l'estimateur $\widehat{\beta}_{l_2}$ sera proche de 0. Si la valeur de l'hyperparamètre λ est égale à 0, alors l'estimateur obtenu sera celui des moindres carrés. C'est pourquoi, il est important de bien choisir cet hyperparamètre. Comme pour le Lasso (Tibshirani, 1997), il est généralement choisi par validation croisée. Ainsi, la pénalisation Ridge n'introduit pas de sparsité mais elle permet de gérer la corrélation potentielle entre deux variables (Hastie et al., 2015). Si deux prédicteurs sont très corrélés, la régularisation Ridge tendra à leur donner des poids égaux. La procédure Ridge (Hoerl et al., 1970) correspond à la minimisation de la log-vraisemblance partielle de Cox en y ajoutant un terme de pénalité l_2 :

$$\widehat{\beta}_{l_2} = \arg\min_{\beta} \left\{ -\sum_{i=1}^{n} \left(\beta^T X_{i.}\right) - \sum_{i=1}^{n} \delta_i \log \left(\sum_{l \in R_{i.}} \exp\left(\beta^T X_{l.}\right) \right) + \lambda \sum_{j=1}^{p} |\beta_j|^2 \right\},$$

avec $R_{i.}$ les individus à risque au temps $t_{i.}$, δ_i l'indicateur de censure et $\lambda \in \mathbb{R}^+$ un hyperparamètre choisi par validation croisée. La régularisation Ridge peut également être utilisée comme méthode de sélection en classant les covariables par importance à partir des $||\widehat{\beta}_{l_2}||^2$ en sélectionnant un certain nombre fixé en avance, puis en réalisant à nouveau une régression pénalisée.

3 Elastic-net

La pénalité Elastic-Net a été introduite par Zou et al. (2005) et Wu (2012), elle combine une pénalité en norme l_1 avec celle en norme l_2 :

$$\begin{aligned} pen(\beta) &= \lambda_1 ||\beta||_1 + \lambda_2 ||\beta||_2^2 \\ &= \lambda_1 \sum_{j=1}^{p} |\beta_j| + \lambda_2 \sum_{j=1}^{p} |\beta_j|^2. \end{aligned} \tag{2.8}$$

La pénalité Elastic-Net (Wu, 2012) combine les forces de la pénalité Lasso (Tibshirani, 1996) avec celles de la pénalité Ridge (Hoerl et al., 1970). Elle dispose de la propriété de la sélection de variables et pallie le défaut de l'estimation Lasso quand des variables sont fortement corrélées grâce à la partie Ridge. La procédure Elastic-Net (Zou et al., 2005) correspond à la minimisation de la log-vraisemblance partielle de Cox en y combinant deux termes de pénalité l_1 et l_2 :

$$\widehat{\beta}_{l_{12}} = \arg\min_{\beta} \left\{ -\sum_{i=1}^{n} \left(\beta^T X_{i.}\right) - \sum_{i=1}^{n} \delta_i \log \left(\sum_{l \in R_{i.}} \exp\left(\beta^T X_{l.}\right) \right) + \lambda_1 \sum_{j=1}^{p} |\beta_j| + \lambda_2 \sum_{j=1}^{p} |\beta_j|^2 \right\},$$

avec $R_{i.}$ les individus à risque au temps $t_{i.}$, δ_i l'indicateur de censure et $(\lambda_1, \lambda_2) \in (\mathbb{R}^+)^2$ deux hyper-paramètres choisis par validation croisée. Cette régularisation est très utile dans notre problème de grande dimension où le nombre important de variables provoque souvent des problèmes de colinéarité entre variables.

4 Adaptive-Lasso

Lors de la présentation de la méthode de régularisation Lasso (Tibshirani, 1997), nous avons rappelé que celle-ci est instable en sélection. En effet, si deux variables sont fortement corrélées entre elles, le Lasso les sélectionnera de façon aléatoire si elles ont toutes deux un effet sur la variable à expliquer. Pour résoudre cela, Zou (2006) a proposé une version adaptative du Lasso, appelée Adaptive-Lasso. Cette procédure a été par la suite étendue au modèle de Cox par Zhang et al. (2007). Cette régularisation pénalise moins les grands coefficients que ceux qui sont plus petits. Cela se réalise grâce à la pénalisation utilisée, qui est une pénalité Lasso (de norme l_1) pondérée :

$$pen(\beta) = \lambda \sum_{j=1}^{p} w_j |\beta_j|, \tag{2.9}$$

avec $w_j = \frac{1}{|\widehat{\beta}_j|^\gamma}$ où $\gamma > 0$ et les $\widehat{\beta}_j$ sont les coordonnées obtenues par un estimateur lors d'une étape préliminaire. Dans le travail présenté dans ce manuscrit, nous avons utilisé comme estimateur préliminaire $\widehat{\beta}$ l'estimateur Lasso $\widehat{\beta}_{l_1}$. La pénalité utilisée est donc de la forme :

$$pen(\beta) = \lambda \sum_{j=1}^{p} \frac{|\beta_j|}{|\widehat{\beta}_j^{l_1}| + \epsilon},$$

où ϵ est le minimum des $\widehat{\beta}_j^{l_1}$ non-nuls. Cette constante ϵ est ajoutée au dénominateur pour éviter de diviser par zéro. La procédure adaptive-Lasso (Zhang et al., 2007) correspond donc à la minimisation de la log-vraisemblance partielle de cox avec l'ajout d'une pénalisation Lasso pondérée :

$$\widehat{\beta}_{l_{ada}} = \arg\min_{\beta} \left\{ -\sum_{i=1}^{n} \left(\beta^T X_{i.}\right) - \sum_{i=1}^{n} \delta_i \log \left(\sum_{l \in R_{i.}} \exp\left(\beta^T X_{l.}\right) \right) + \lambda \sum_{j=1}^{p} \frac{|\beta_j|}{|\widehat{\beta}_j^{l_1}| + \epsilon} \right\},$$

avec $R_{i.}$ les individus à risque au temps $t_{i.}$, δ_i l'indicateur de censure et $\lambda \in \mathbb{R}^+$ choisi par validation croisée. L'estimateur adaptive-Lasso $\widehat{\beta}_{l_{ada}}$ est consistant en sélection de variables, mais est plus biaisé que le Lasso. Une solution pour débiaiser l'estimation consiste à exécuter de nouveau la procédure d'estimation classique sans pénalité avec seulement les variables sélectionnées par l'adaptive-Lasso.

2.2.2 Méthodes de Screening

Les méthodes de régularisation sont réputées instables en sélection (Michiels et al., 2005). De plus, ce phénomène d'instabilité est renforcé quand le nombre de variables augmente de manière importante, ce qui est le cas avec les données moléculaires. Il existe plusieurs méthodes de *Screening* dont le principe général peut se résumer en deux étapes. La première consiste à réduire le nombre de variables en gardant seulement celles qui ont un score dans le modèle de Cox plus grand qu'un certain seuil. La seconde étape consiste à exécuter une procédure Lasso pour sélectionner les variables les plus importantes parmi les variables sélectionnées lors de la première étape.

Nous décrivons dans les sections suivantes les méthodes de *Sure Independence Screening (SIS)* et *Iterative Sure Independence Screening (ISIS)* proposées initialement pour la régression linéaire par Fan et al. (2008) et étendues par la suite au modèle de Cox par Fan et al. (2010a). Ensuite, nous introduirons le *Principled Sure Independence Screening (PSIS)* proposé par Zhao et al. (2012) dont l'avantage est de contrôler les faux-positifs. Enfin, nous présenterons *Cox Conditional Screening* (CoxCS) introduit par Hong et al. (2018) qui utilise la connaissance biologique pour effectuer une pré-sélection de variables. Nous avons ensuite appliqué ces méthodes sur des données réelles concernant le cancer du rein à cellules claires de la base de données TCGA (*The Cancer Genome Atlas*).

1 (I)SIS

Les méthodes SIS et ISIS ont été introduites par Fan et al. (2010a) et un *package* R a été réalisé dans lequel nous pouvons retrouver les différentes variantes de SIS et ISIS.

SIS

Pour chaque variable, on calcule un score, appelé *marginal utility*, qui correspond à l'estimation des coefficients de régression par maximum de la log-vraisemblance partielle du modèle marginal contenant seulement la m^e variable :

$$u_m = \max_{\beta_m} \left(\sum_{i=1}^{n} (\delta_i \beta_m X_{im}) - \sum_{i=1}^{n} \delta_i \log \left(\sum_{j \in R_{i.}} \exp(\beta_m X_{jm}) \right) \right).$$

Les variables sont classées suivant la valeur du score par ordre décroissant. Ensuite, les d premières variables ayant le plus grand score sont sélectionnées et les indices de celles-ci correspondront à l'ensemble $\mathcal{I}$. L'ensemble de ces variables constitue le premier modèle sélectionné. Cependant, nous ne pouvons pas savoir

l'ordre de grandeur de cet ensemble et en choisissant une valeur trop grande pour d (le choix de la valeur de d est discuté à la fin de la section 1), le modèle pourrait contenir des variables non-importantes. La réduction du modèle à la taille d nous permet par la suite d'appliquer la pénalité Lasso sur le problème d'optimisation de la log-vraisemblance partielle du modèle de Cox :

$$\arg\max_{\beta_{\mathcal{I}}} \left(\sum_{i=1}^{n} \delta_i \beta_{\mathcal{I}}^T X_{\mathcal{I}} + \sum_{i=1}^{n} \delta_i \log \left(\sum \exp(\beta_{\mathcal{I}}^T X_{\mathcal{I}}) \right) - \sum_{m \in \mathcal{I}} p_\lambda(\beta_m) \right),$$

où $\beta_{\mathcal{I}}$ est le vecteur des coefficients des variables dont les indices appartiennent à $\mathcal{I}$. La procédure Lasso permet de mettre à zéro les variables non-informatives. Le modèle final va être composé des variables dont les paramètres sont non nuls, l'ensemble des variables sélectionnées du modèle est noté $\widehat{\mathcal{M}}$ et les coefficients estimés sont notés $\beta_{\widehat{\mathcal{M}}}$.

ISIS

La méthode SIS peut ne pas performer correctement quand certaines variables importantes sont non-corrélées avec la variable à expliquer. Cela peut se produire quand les variables sont corrélées entre elles, mais elles n'ont pas de grande influence individuellement sur la variable à expliquer ou quand des variables ne sont pas forcément corrélées entre elles mais vont avoir individuellement un impact plus fort sur la variable à expliquer que certaines variables importantes. Une version itérative de SIS peut résoudre ce problème en comparant à chaque étape les variables sorties avec le modèle sélectionné par SIS. ISIS essaye donc d'utiliser plus l'information conjointe des variables. La procédure est la suivante :

1. La méthode *SIS* est dans un premier temps appliquée sur l'ensemble des variables du modèle initial. Les k_1 variables avec le score le plus élevé sont sélectionnées. Le modèle obtenu sera $\widehat{\mathcal{M}}_1$ de dimension $|\widehat{\mathcal{M}}_1|$ et l'ensemble des indices des variables appartenant au modèle $\widehat{\mathcal{M}}_1$ est $\mathcal{I}_1$.
2. L'ensemble des indices des variables non-sélectionnées est noté $\mathcal{I}_1^C$. Pour chaque $m \in \mathcal{I}_1^C$, un nouveau score est calculé, appelé *conditional utility* :

$$u_{m|\widehat{\mathcal{M}}_1} = \max_{\beta_m, \beta_{\widehat{\mathcal{M}}_1}} \left[\sum_{i=1}^{n} \delta_i (\beta_m X_{im} + \beta_{\widehat{\mathcal{M}}_1}^T X_{i\widehat{\mathcal{M}}_1}) - \sum_{i=1}^{n} \delta_i \left\{ \log \sum_{j \in R_{i.}} \exp(\beta_m X_{jm} + \beta_{\widehat{\mathcal{M}}_1}^T X_{i\widehat{\mathcal{M}}_1}) \right\} \right],$$

   Ce score est calculé pour chaque variable non sélectionnée à la première étape en prenant en compte le modèle $\widehat{\mathcal{M}}_1$ (celles sélectionnées à la première étape) et chaque variable est classée en respectant ce score. Les k_2 variables dont le score est le plus élevé sont sélectionnées. L'ensemble $\mathcal{I}_2$ contient les indices des k_2 variables sélectionnées et il est appelé ensemble relatif.
3. Ensuite, la log-vraisemblance partielle du modèle de Cox avec un terme de pénalité Lasso va être maximisée pour les deux ensembles de variables sélectionnées à l'étape 1 et 2 ($\mathcal{I}_1 \cap \mathcal{I}_2$). Les variables dont le coefficient est non-nul seront sélectionnées et constitueront le modèle final. Il sera noté $\widehat{\mathcal{M}}_2$.
4. Enfin, l'étape 2 (avec k_i) et 3 sont répétées jusqu'à ce que la cardinalité du modèle final atteigne la valeur d définie en amont ou jusqu'à ce que $\widehat{\mathcal{M}}_i = \widehat{\mathcal{M}}_{i+1}$.

Pour les méthodes SIS et ISIS, une valeur seuil d correspondant au nombre maximal de variables sélectionnées dans le modèle doit être définie. Mais cette valeur d est difficile à choisir. Saldana et al. (2018) suggérait dans le papier accompagnant leur *package* R SIS de fixer $d = \lfloor \frac{n}{4\log(n)} \rfloor$ dans le cadre des données de survie censurées. Ce paramètre est basé sur des expérimentations et le choix de celui-ci n'est pas justifié. Pour éviter de fixer d, d'autres procédures de *Screening* ont vu le jour comme PSIS.

2 PSIS

La méthode PSIS est une méthode de *Screening* développée par Zhao et al. (2012). Cette méthode possède des similitudes avec la méthode SIS, mais le calcul du score est différent entre les deux méthodes. L'originalité de cette méthode est le choix du seuil pour sélectionner le nombre de variables dans le modèle intermédiaire. Le choix du seuil est justifié par Zhao et al. (2012) afin de contrôler les faux-positifs. Les étapes de la méthode PSIS sont :

1. Les coefficients de régression sont estimés individuellement pour chaque variable par maximum de la log-vraisemblance partielle de Cox. L'estimation des coefficients de régression β_j ainsi que l'estimation de la variance des coefficients β (calculée à partir de l'inverse de la matrice de l'information de Fisher) $I_j(\widehat{\beta}_j)^{-1}$ sont récupérés.
2. Le taux de faux-positifs est fixé, on introduit : $q_n = f/p_n$, où f est le nombre de faux positifs toléré et p_n est le nombre des variables et ainsi un seuil γ est calculé de la manière suivante : $\gamma = \phi^{-1}(1 - q_n/2)$, où ϕ est la fonction de répartition de la loi normale.
3. Les variables sont ensuite classées suivant leur valeur de score. Ce score est calculé à partir des coefficients de régression et de la variance de ces coefficients : il est égal à $I_j(\widehat{\beta}_j)^{1/2}|\widehat{\beta}_j|$. Les variables dont le score est supérieur au seuil γ sont sélectionnées. Ces variables appartiennent donc au modèle intermédiaire.
4. Enfin, une estimation par maximum de la log-vraisemblance partielle associée à un terme de pénalité Lasso est réalisée sur les variables appartenant au modèle intermédiaire. Les variables dont les coefficients sont non-nuls sont sélectionnées et correspondent au modèle final.

L'avantage de cette procédure est la propriété de contrôler les faux-positifs, c'est-à-dire qu'on assure que le taux de faux-positifs sera plus faible que le taux autorisé (f/p_n). Bien que cette méthode soit basée sur le contrôle de faux positifs, elle ne prend pas en compte le taux de faux-négatifs. Le fait de contrôler cette dernière permettrait d'éviter d'omettre certaines variables importantes et ainsi éviter d'obtenir un modèle peu informatif. Enfin, la nature non-itérative de PSIS amène aux mêmes problèmes que ceux engendrés par SIS dans le cas de variables non-corrélées à la variable à expliquer, mais influençant malgré tout la variable à expliquer à travers la corrélation conjointe avec d'autres variables.

3 CoxCS

La méthode CoxCS est développée par Hong et al. (2018). L'idée de cette méthode est l'utilisation de la connaissance biologique pour supprimer les variables non pertinentes.

1. Certaines variables sont connues pour avoir une influence sur la survie dans le contexte d'une maladie particulière. Ces variables vont être sélectionnées et vont appartenir au modèle $\mathcal{M}_C$. La connaissance biologique aide à sélectionner un ensemble de variables en amont qui devrait avoir un impact sur la survie des patients.
2. Pour chaque variable qui n'appartient pas au modèle $\mathcal{M}_C$, leur coefficient de régression va être estimé par maximum de la log-vraisemblance partielle du modèle de Cox dans laquelle sont incluses les variables du modèle $\mathcal{M}_C$:

$$\max_{\beta_j, \beta_{\mathcal{M}_C}} \left(\sum_{i=1}^{n} (\delta_i(\beta_j X_{ij} + \beta_{\mathcal{M}C} X_{i\mathcal{M}_C}) - \sum_{i=1}^{n} \delta_i \log \left(\sum_{k \in R_{i.}} \exp(\beta_j X_{kj} + \beta_{\mathcal{M}C} X_{k\mathcal{M}_C}) \right) \right).$$

3. Pour un seuil $\gamma > 0$ donné, l'ensemble des indices sélectionnés en plus de ceux dans $\mathcal{M}_C$ sont :

$$\widehat{\mathcal{M}}_{-C} = \left\{ j \notin \mathcal{M}_C : |\widehat{\beta}_j| \geq \gamma \right\}$$

4. Enfin, la log-vraisemblance partielle associée au terme de pénalité Lasso est maximisée sur les modèles $\widehat{\mathcal{M}}_{-\mathcal{C}} \cap \mathcal{M}_{\mathcal{C}}$. Les variables dont l'estimation des coefficients est non-nulle constitueront le modèle final $\widehat{\mathcal{M}}$.

L'atout de cette méthode est de pouvoir détecter les variables avec un faible score individuel, mais un signal important conjointement à d'autres variables. Ces autres variables sont connues pour avoir une importance dans l'application biologique étudiée. De plus, Hong et al. (2018) utilisent le même seuil que celui proposé dans la méthode PSIS par (Zhao et al., 2012) pour la sélection des variables. Ce seuil a l'avantage de pouvoir contrôler les faux positifs. Cette méthode permet donc de pallier le problème de la co-expression de gènes par sa pré-sélection, mais obtenir de la connaissance biologique en amont n'est pas une chose simple dans certaines applications.

2.2.3 Applications

Nous avons appliqué les méthodes présentées ci-dessus à un jeu de données réelles. Ce jeu de données concerne le cancer du rein à cellules claires (ccRCC) provenant de la base de données TCGA (*The Cancer Genome Atlas*). Les données sont disponibles sur le site `https://www.cancer.gov/tcga`. Nous avons appliqué deux types de méthodes, les méthodes de régularisation et les méthodes de *Screening*. Nous présentons tout d'abord les résultats obtenus pour les méthodes de régularisation : Lasso (Tibshirani, 1997), Ridge (Verweij et al., 1994), Adaptive-Lasso (Zhang et al., 2007) et Elastic-Net (Zou et al., 2005). Nous précisions que les résultats de Ridge et Elastic-Net sont calculés à partir du top 20 des gènes. Ensuite les résultats obtenus pour les méthodes de *Screening* : SIS (Fan et al., 2010a), ISIS (Fan et al., 2010a), PSIS (Zhao et al., 2012) et CoxCS (Hong et al., 2018) sont présentés.

Trois approches ont été suivies pour l'étude de ces méthodes. La première approche concerne seulement les *Immune-Checkpoints* (IC), gènes impliqués dans le processus de la réponse immunitaire. Ces gènes sont au nombre de quarante-huit. La deuxième approche concerne les gènes considérés différentiellement exprimés lors de notre étude présentée en section 2.1. Le nombre de gènes trouvés différentiellement exprimés par la méthode DESeq2 (Love et al., 2014) est de 11 289 gènes. Enfin, la troisième approche concerne l'ensemble des gènes codants présents chez l'humain. Après un filtrage afin de supprimer les gènes nuls, nous avons obtenus 17 789 gènes.

Les méthodes de régularisation sont connues pour être instables quand la dimension augmente et les méthodes de *Screening* ont donc été développées pour essayer de répondre à cette problématique. Dans cette section, nous essayons également de quantifier la stabilité de ces méthodes. Nous proposons d'utiliser un indice de similarité, appelé l'indice de Sørensen (Arita, 2017; Baselga, 2013; Baselga, 2010) généralisé à une famille de n ensembles avec $n > 2$, classiquement utilisé en écologie pour mesurer la variabilité de la composition des espèces dans différents sites. Nous renvoyons à l'Annexe B pour les détails de son utilisation en écologie et nous présentons dans le paragraphe suivant notre utilisation de cet indice dans le cadre de la sélection de variables. Dans le contexte de notre étude, nous voulons mesurer la variation de la sélection par les différentes méthodes lorsque celles-ci sont exécutées plusieurs fois. En effet, la sélection des gènes par les différentes méthodes diffère quand elles sont lancées pour différentes graines. Une graine est un entier utilisé pour initialiser un générateur de nombres aléatoires. Or, les méthodes de régularisation exécutent une validation croisée (concept détaillé dans la section 3) pour le choix de l'hyperparamètre λ du critère de pénalisation. Les jeux de données sont donc séparés en k sous-parties et le remplissage de ces sous-parties se fait de manière aléatoire, en fonction de la graine choisie.

La démarche que nous avons suivie est l'exécution des méthodes sur 100 graines différentes et nous avons créé une matrice avec en ligne les différentes graines et en colonne les différents gènes. Si le gène j est sélectionné pour la graine i, alors le coefficient (i, j) de la matrice vaudra 1 sinon il vaut 0. Nous utilisons alors l'indice

de Sørensen qui permet de mesurer la similarité des gènes sélectionnés entre les différentes graines. L'indice de Sørensen permet de comparer les graines en considérant la présence ou l'absence de gènes. Il correspond au rapport entre les recouvrements de gènes sélectionnés par les différentes graines et le nombre moyen de gènes sélectionnés.

Soient N le nombre de gènes sélectionnés par au moins une graine et S le nombre de graines. On appelle E_i l'ensemble des gènes sélectionnés par la graine i, on note $|E_i| = n_i$ son cardinal. Réciproquement, on note s_j le nombre de graines dans lesquelles le gène j est sélectionné.

S'il n'y a que 2 graines, l'indice de Sørensen est facilement interprété en terme ensembliste comme le rapport entre la taille de l'intersection des ensembles de gènes sélectionnés et la taille moyenne des ensembles. L'indice de Sørensen est ainsi donné par :

$$S_2 = \frac{|E_1 \cap E_2|}{\frac{1}{2}\left(|E_1| + |E_2|\right)}$$

Si un gène appartient à $E_1 \cap E_2$, on dit qu'il appartient à un recouvrement, et s'il n'appartient pas à $E_1 \cap E_2$, il n'appartient à aucun recouvrement. Cela se généralise à un nombre de graines plus grand. Le nombre de recouvrements d'un Le nombre de recouvrements avec un des ensembles auquel le gène j appartient est donc tout simplement $s_j - 1$. Dans le cas optimal, la taille de ce recouvrement serait $S - 1$, et on on appelle taux de recouvrement pour le gène j : $(s_j - 1)/(S - 1)$.

Finalement, la mesure de recouvrement est la somme sur tous les gènes de ce taux de recouvrement. Dans le cas ou $S = 2$, ce recouvrement est directement la taille de l'intersection

$|E_1 \cap E_2| = \sum_{j=1}^{N}(s_j - 1)/(S - 1)$

S_2 se récrit alors :

$$S_2 = \frac{\sum_{j=1}^{N}(s_j - 1)}{\frac{1}{2}\left(n_1 + n_2\right)}.$$

Pour la généralisation à un nombre de graines plus grands, on divise la mesure de recouvrement par la taille moyenne sur tous les ensembles de sélection :

$$S_S = \frac{\frac{1}{S-1}\sum_{j=1}^{N}(s_j - 1)}{\frac{1}{S}\sum_{i=1}^{S} n_i}.$$

Dans le cas où toutes les graines sélectionnent les mêmes N gènes, le dénominateur (c'est à dire le recouvrement) vaut N, et c'est également la taille moyenne des ensembles, l'indice vaut donc 1. A l'inverse si chaque gène n'est sélectionné qu'une fois, $s_j = 1$ pour tout j, et l'indice vaut 0.

Les résultats de cet indice et du nombre de gènes sélectionnés pour les deux types de méthodes sont présentés dans les Tables 2.1 et 2.4.

Nous avons également calculé le critère d'information d'Akaike (appelé AIC pour *Akaike Information Criterion*) du modèle obtenu pour chaque graine. Le critère AIC a été développé par Akaike (1998) et permet d'évaluer la qualité d'un modèle. Il permet de gérer à la fois la qualité de l'ajustement et de la complexité du modèle en pénalisant les modèles ayant un grand nombre de paramètres. Le meilleur modèle sera celui avec la valeur la plus faible du critère AIC. Nous calculons la moyenne et l'écart-type du critère AIC pour chacune des méthodes sur les 100 graines. Ce calcul nous permet également de juger la qualité de la sélection et les résultats de l'AIC moyen des méthodes de régularisation et de *Screening* ont été ajoutés respectivement dans les Table 2.1 et 2.4.

Enfin, nous avons donc calculé un pourcentage de sélection pour chaque gène. Les méthodes ont été appliquées sur 100 graines différentes pour étudier leur stabilité, mais également pour une meilleure interprétabilité de la sélection. Cela nous permet d'éviter de conclure à l'importance d'un gène qui aurait été sélectionné de rares fois, par chance. L'analyse biologique des gènes sélectionnés a été réalisée à partir du site `GeneCards` (Stelzer et al., 2016) accessible à l'adresse `https://www.genecards.org/` et du site `COSMIC` (Sondka et al., 2018) (pour *Catalogue Of Somatic Mutations In Cancer*) accessible à l'adresse `https://cancer.sanger.ac.uk/cosmic`. `GeneCards` correspond à une base de données qui fournit une information complète de l'ensemble des gènes humains prédits et annotés. `COSMIC` est également une base de données, mais celle-ci est exclusivement dédiée au cancer. Pour `COSMIC`, nous nous sommes particulièrement basés sur le catalogue *Cancer Gene Census* qui répertorie les gènes avec mutations impliquées dans le cancer.

1 Méthodes de régularisation

Nous précisons ici que les résultats des méthodes Ridge et Elastic-Net sont calculés à partir du top 20 des gènes, classés à partir des $|\widehat{\beta}_j|$ obtenus par régression pénalisée.

		Lasso	Ridge	Adaptive-Lasso	Elastic-Net
Immune-Checkpoints	Indice de Sørensen	0.9960	0.9975	0.9933	0.9940
	Nombre de gènes sélectionnés	15.36 (2.83)	20 (fixé)	10.84 (3.53)	20 (fixé)
	AIC	1915.50 (4.33)	1919.42 (2.01)	1917.71 (11.06)	1932.69 (3.34)
Sur les variables différentiellement exprimées	Indice de Sørensen	0.9946	0.9500	0.9436	0.9501
	Nombre de gènes sélectionnés	11.72 (2.34)	20 (fixé)	7.84 (3.01)	20 (fixé)
	AIC	1867.35 (1.95)	1869.61 (2.91)	1862.47 (23.04)	1878.10 (4.45)
Sur l'ensemble des gènes	Indice de Sørensen	0.9332	0.9940	0.8284	0.9755
	Nombre de gènes sélectionnés	17.70 (3.57)	20 (fixé)	8.65 (3.64)	20 (fixé)
	AIC	1873.43 (24.95)	1870.44 (8.33)	1870.42 (40.97)	1874.36 (5.60)

Table 2.1 – Résultats de l'indice de similarité (indice de Sørensen), de l'AIC moyen et du nombre moyen de gènes sélectionnés sur l'ensemble des graines pour l'ensemble des méthodes de régularisation étudiées suivant l'ensemble de gènes donné en entrée. Leurs écart-types sont précisés dans les parenthèses.

Immune-Checkpoints

Tout d'abord, les méthodes de régularisation ont été appliquées sur un sous-ensemble de gènes. L'équipe de biologistes de l'hôpital Saint-Louis nous avait transmis pour l'étude de la section 2.1 une liste de ckeckpoints dont une implication dans le ccRCC était possible. Nous avons voulu voir si ces checkpoints pouvaient également influencer la durée de survie.

Nous présentons une partie des résultats dans cette partie, nous renvoyons à l'Annexe C pour les tableaux de résultats supplémentaires. Nous présentons les résultats obtenus à partir des ensembles de gènes : *Immune-Checkpoints* et sur les variables différentiellement exprimées ici et les résultats obtenus à partir de l'ensemble des variables sont présentés en Annexe C.

	gènes sélectionnés			
Lasso (AIC = 1915.50 ±4.33)	**ADORA1**(100%) $CD274$(87%) **CTLA4**(100%) $KIR2DL1$(1%) $PDCD1$(27%) **TNFSF4**(100%)	$BAP1$(15%) **CD276**(100%) **ENTPD1**(98%) $LAG3$(15%) **S100A8**(100%)	**BST1**(100%) $CD40LG$(27%) $HAVCR2$(78%) **LILRB1**(98%) **TNFRSF18**(100%)	$CD200R1$(87%) $CD80$(27%) **HLA.G**(100%) $PDCD1LG2$(78%) **TNFRSF4**(98%)
Ridge (top 20) Gènes classés par importance de l'estimateur (AIC = 1919.42 ±2.01)	**ADORA1**(100%) $CD200$(4%) $CD48$(64%) **HAVCR2**(100%) $LILRB2$(36%) $S100A9$(2%) VHL(4%)	**BST1**(100%) **CD274** (94%) **CD80**(100%) **HLA.G**(100%) **PBRM1**(100%) **TNFRSF18**(100%)	$CD160$(96%) **CD276**(100%) **CTLA4**(100%) **LAG3**(100%) **PDCD1LG2**(100%) **TNFRSF4**(100%)	**CD200R1**(100%) $CD40LG$(49%) **ENTPD1**(100%) **LILRB1** (100%) $S100A8$(51%) **TNFSF4** (100%)
Adaptive-Lasso (AIC = 1917.71 ±11.06)	**ADORA1**(98%) $CD274$(66%) $ENTPD1$(84%) $PDCD1LG2$(60%) $TNFRSF4$(65%)	$BAP1$(4%) **CD276** (98%) **HLA.G**(97%) $PDCD1$(19%) $TNFSF4$(65%)	**BST1**(98%) $CD40LG$(6%) $LAG3$(4%) $S100A8$(56%)	$CD200R1$(60%) **CTLA4**(98%) $LILRB1$(65%) $TNFRSF18$(41%)
Elastic-Net Top 20 Gènes classés par importance de l'estimateur (AIC = 1932.69 ±3.34)	**ADORA1**(100%) $C10orf54$(29%) $CD244$(12%) $CD28$(7%) **CD80** (94%) **HLA.G**(89%) **LILRB2**(94%) $S100A9$(84%)	$BAP1$(47%) **CD160**(100%) $CD274$(12%) $CD38$(7%) **CTLA4**(100%) **LAG3** (94%) $PBRM1$(71%) **TNFRSF18**(100%)	**BST1**(100%) $CD200R1$(24%) **CD276**(100%) $CD40LG$(7%) $ENTPD1$(89%) $LGALS9$(67%) $PDCD1LG2$(47%) $TNFRSF4$(89%)	$BTLA$(34%) $CD200$(12%) $CD27$(12%) $CD40$(1%) $HAVCR2$(89%) **LILRB1**(100%) **S100A8**(99%) **TNFSF4**(100%)

Table 2.2 – Résultats des méthodes de régularisation sur les checkpoints. Les gènes sélectionnés avec un pourcentage supérieur à 90% sont en violet, les gènes soulignés sont ceux en commun entre les quatre méthodes avec un pourcentage de sélection supérieur à 98% et le gène HLA.G est mis en évidence en bleu (car il est d'intérêt pour les biologistes).

La Table 2.2 représente les gènes sélectionnés par les différentes méthodes de régularisation : Lasso (Tibshirani, 1997), Ridge (Verweij et al., 1994), Adaptive-Lasso (Zhang et al., 2007) et Elastic-Net (Zou et al., 2005) en considérant 100 graines différentes.

L'indice de Sørensen est grand pour les méthodes de régularisation mais les méthodes Lasso et Ridge sont celles avec l'indice le plus élevé, ce qui signifie que leurs sélections sont les plus stables. Cependant, le top 20 des variables obtenues avec la méthode de ridge ne sont pas celles qui expliquent le mieux la durée de survie. Le Lasso possède un meilleur AIC moyen que la méthode Ridge (1915.50). Lasso a de très bons résultats sur ce jeu de données. Les valeurs de l'indice de Sørensen obtenues par les 4 méthodes sur ce jeu de données sont les plus hautes de la Table 2.1. Cette plus grande stabilité dans ce cas de jeu de données n'est pas surprenante, le nombre de variables est réduit ($p < n$).

De plus, nous pouvons également observer que l'intersection des checkpoints sélectionnés avec un pourcentage supérieur à 80% par l'ensemble des méthodes de régularisation correspond à l'ensemble $\{ADORA1, BST1, B7H3(CD276), CTLA4, HLA.G\}$. $CD276$ est un nom alternatif de $B7H3$ et celui-ci serait un

biomarqueur pour discriminer les cellules tumorales des cellules saines (selon l'étude menée en section 2.1) et impacterait également la survie de patients. Nous pouvons également observer que le gène $CTLA4$ est sélectionné. Celui-ci est déjà connu et utilisé dans certains traitements d'immunothérapie. Nous pouvons également observer dans la Table 2.2 que $HLA.G$ est sélectionné par toutes les méthodes et son pourcentage minimal de sélection est de 89%. Un autre gène qui est sélectionné par l'ensemble des méthodes mais dont le pourcentage pour une méthode (Elastic-Net) est inférieur à 80% est le gène $LILBR1$ (appelé $ILT2$ dans l'étude de la section 2.1). Dr Diana Tronik Le Roux évoquait l'intérêt du ciblage des deux gènes ($HLA.G/ILT2$) comme traitement intermédiaire en cas de non réponse au traitement $PD1/PDL1$. Ces deux gènes semblent donc également intéressants pour expliquer la survie des patients.

Sur les variables différentiellement exprimées

Dans ce paragraphe, nous donnons les résultats de l'application des méthodes de régularisation sur les gènes différentiellement exprimés. Ces gènes différentiellement exprimés ont été obtenus à l'aide de la méthode DESeq2 (Love et al., 2014) et une partie des résultats a été montrée en section 2.1. Comme pour l'application sur les checkpoints, nous pouvons observer que les méthodes semblent être stables. Les valeurs d'indice de Sørensen sont toutes supérieures à 0.9. Sur ce jeu de données, la méthode adaptive-Lasso est la moins stable des quatre méthodes, sa valeur est égale à 0.9436. Ce résultat est confirmé par la Table 2.3, où nous pouvons voir que le nombre de gènes sélectionnés avec un pourcentage inférieur à 5% est important. Le Lasso est la méthode de régularisation la plus stable avec une valeur d'indice de Sørensen égale à 0.9946. Les variables sélectionnées à partir de cette méthode sont également celles qui expliquent le mieux en moyenne la survie. La valeur de son AIC moyen est de 1867.30, celle de l'adaptive-lasso est plus faible mais son écart-type est très élevé. L'adaptive-lasso est donc moins stable que le Lasso.

	gènes sélectionnés			
Lasso (AIC = 1867.35 ±1.95)	**B3GNTL1**(100%) $EHHADH$(41%) **MAST4**(96%) $RGS17$(67%) **STRADA**(84%)	$C10orf90$(29%) **FBXL5**(100%) $NUMBL$(16%) $RGS20$(67%) $TPRG1L$(15%)	**CHEK2**(99%) $GTPBP2$(5%) **OTOF**(100%) **SEC61A2**(98%)	**CKAP4**(99%) $KIF18B$(15%) $RANGAP1$(41%) **SORBS2**(100%)
Ridge (top 20) Gènes classés par importance de l'estimateur (AIC = 1869.61 ±2.91)	$APCDD1L$(60%) $CDCA3$(3%) $EHHADH$(24%) $KIF18B$(3%) $NEK2$(40%) **RGS20**(100%) **SLC12A8**(100%) $TMEM132A$(40%) $TROAP$(3%)	AR(68%) $CDCA7$(59%) **FBXL5**(100%) $KIF23$(3%) **OTOF**(100%) $RORA$(60%) $SLC16A12$(37%) $TPRG1L$(29%) $VAMP3$(37%)	**B3GNTL1**(60%) $CENPBD1$(1%) $FOXF2$(63%) $KLHL36$(13%) **RANGAP1**(100%) **SEC61A2**(100%) **SORBS2** (100%) $TRAIP$(37%)	**C10orf90**(100%) **CKAP4**(63%) $HJURP$(3%) **MAST4**(100%) **RGS17**(100%) **SGCB**(97%) **STRADA**(100%) **TRNP1**(97%)
Adaptive-Lasso (AIC = 1862.47 ±23.04)	$AGTRAP$(1%) **C10orf90**(25%) $CENPBD1$(4%) $CLCN4$(2%) $DQX1$(14%) $GATA4$(1%) $MAMSTR$(1%) OMD(1%)	$ANKRD17$(1%) $C15orf48$(4%) $CGNL1$(1%) $CTNNA1$(1%) $EFNA4$(1%) $KIAA2026$(4%) **MAST4**(76%) **OTOF**(94%)	$ATP8B4$(1%) $CACNG6$(2%) **CHEK2**(3%) $DCAF16$(2%) **FBXL5**(96%) $LEPR$(2%) $MYEOV$(3%) $PDHX$(1%)	**B3GNTL1**(96%) $CDK5R2$(9%) **CKAP4** (92%) $DEDD2$(4%) $FLT4$(2%) $LSM7$(2%) $MYL5$(2%) $PRKAA2$(4%)

	$PTGDS(1\%)$	$PTPRN2(2\%)$	$RAB40AL(4\%)$	$RANGAP1(2\%)$
	$RASAL1(2\%)$	$RGS17(6\%)$	$RGS20(21\%)$	**SEC61A2**(79%)
	$SMYD2(1\%)$	**SORBS2**(96%)	$SPIC(3\%)$	$TEX19(3\%)$
	$TNNT1(4\%)$	$TRAM1(1\%)$	$UNC13A(2\%)$	$WDFY4(1\%)$
	$WFIKKN1(3\%)$	$WNT1(3\%)$	$ZNF432(3\%)$	$ZSWIM3(2\%)$
Elastic-Net (top 20) Gènes classés par importance de l'estimateur (AIC = 1878.10 ±4.45)	$ANKRD13D(61\%)$	$AR(66\%)$	**B3GNTL1**(66%)	**C10orf90**(36%)
	CDCA3(100%)	$CDCA7(1\%)$	**CHEK2**(65%)	**CKAP4**(100%)
	$EHHADH(65\%)$	$EME1(1\%)$	**FBXL5**(100%)	$FOXF2(35\%)$
	HJURP(100%)	**KIF18B**(100%)	$KIF23(35\%)$	**MAST4**(100%)
	$NEK2(40\%)$	$NUMBL(62\%)$	**OTOF**(100%)	**RANGAP1**(100%)
	$RGS17(37\%)$	$RGS20(37\%)$	**SEC61A2**(100%)	$SGCB(65\%)$
	$SLC12A8(39\%)$	**SORBS2**(100%)	**STRADA**(100%)	**TMEM132A**(100%)
	$TRNP1(1\%)$	$TROAP(99\%)$	$VAMP3(1\%)$	

Table 2.3 – Résultats des méthodes de régularisation sur les gènes différentiellement exprimés. Les gènes sélectionnés avec un pourcentage supérieur à 70% sont en violet, les gènes soulignés sont ceux en commun entre les quatre méthodes avec un pourcentage de sélection supérieur à 70% et deux gènes intéressants (CHEK2 et C10orf90), qui soit ne sont pas sélectionnés dans un des graines soit leur pourcentage de sélection dans certaines méthodes sont faibles, sont représentés en bleu.

De plus, la Table 2.3 montre que les gènes $\{B3GNTL1, FBXL5, CKAP4, MAST4, OTOF, SORBS2$ et $SEC61A2\}$ correspondent à l'ensemble minimal sélectionné par les méthodes de sélection (Lasso, Adaptive-Lasso et Elastic-Net) et le top 20 (les 20 variables dont l'estimation de leur coefficient est la plus élevée) de la méthode Ridge avec un pourcentage de sélection supérieur à 90%. Parmi les gènes sélectionnés, deux semblent particulièrement intéressants. Tout d'abord, nous avons le gène $FBXL5$ qui est impliqué dans le système immunitaire. Plusieurs phénotypes sont reportés pour ce gène, dont un correspond à des maladies chroniques du rein. Ce gène se situe également sur le même segment que les *Immune-Checkpoints* $BST1$ et $CD38$ sur le chromosome 4. Le gène $FBXL5$ semble donc être un gène très important pour expliquer la survie pour le cancer du rein à cellules claires. Ensuite, le gène $CKAP4$ apparaît dans la sélection des différentes méthodes et est impliqué dans le système immunitaire. Une maladie associée au gène $CKAP4$ est la cystite, c'est une maladie qui touchant la vessie qui est une partie du système urinaire comme les reins. Cela peut renforcer l'idée que le gène $CKAP4$ serait un bon marqueur pour la survie de patients atteint du cancer du rein. Deux autres gènes nous semblent intéressants bien qu'ils ne soient pas sélectionnés par les méthodes et/ou sélectionnés avec un faible pourcentage, ce sont les gènes $CHEK2$ et $C10orf90$ qui sont des suppresseurs de tumeur. Un gène suppresseur de tumeur a pour fonction d'éviter l'emballement de la division cellulaire. S'il est donc présent dans une cellule cancéreuse, il a donc tendance à ralentir la prolifération cellulaire. Ce gène est donc un bon indicateur de la survie d'un patient à la maladie considérée.

Sur l'ensemble des variables

Dans ce paragraphe, nous donnons les résultats de l'application des méthodes de régularisation sur l'ensemble des gènes codants.

Nous pouvons observer sur la Table 2.1 que la stabilité des méthode est plus faible sur l'ensemble des gènes comparé aux ensembles contenant les checkpoints ou les variables différentiellement exprimés. C'est notamment le cas pour les méthodes de régularisation utilisant la norme l_1, mais moins pour les méthodes utilisant la norme l_2. En effet, Ridge, méthode de régularisation utilisant une norme l_2, a la valeur de l'indice de Sørensen la plus élevée sur l'ensemble des gènes comparée à celle sur les gènes différentiellement exprimés. Elle

est donc plus stable sur ce jeu de données que celui des gènes différentiellement exprimés et nous voyons le même comportement pour l'Elastic-Net. Cette méthode utilise à la fois la norme l_1 et l_2, mais nous évoquons que la norme l_2 a plus d'importance sur ce jeu de données que la norme l_1. Ridge, contrairement à la méthode Lasso, ne met pas les variables non informatives à zéro mais les fait tendre vers zéro. Pour faire de la sélection, nous avons donc décidé de prendre le top 20 des variables. Sa stabilité est la meilleure et les variables "sélectionnées" sont celles qui expliquent le mieux en moyenne la survie. De plus, la stabilité des méthode Lasso et adaptive-Lasso diminue avec l'augmentation de la dimension des données. Enfin, leurs valeurs moyennes d'AIC sont similaire à Ridge, mais leurs écart-types est très élevés. Cela s'explique par le manque de stabilité en sélection. Nous pouvons également voir sur la Table C.1 que beaucoup de gènes ne sont sélectionnés que dans une seule graine, surtout pour les méthodes Lasso et Adaptive-Lasso, c'est moins le cas pour les méthodes Ridge et Elastic-Net. Cela pourrait s'expliquer par la fait qu'on a seulement gardé le top 20 des gènes sélectionnés pour ces deux dernières méthodes.

Nous pouvons également observer que les gènes sélectionnés sont la plupart du temps les mêmes en considérant seulement les gènes différentiellement exprimés (cf. Table 2.3) et l'ensemble des gènes (cf. Table C.1) pour le Lasso et l'Adaptive-Lasso. C'est également le cas pour la méthode de régularisation Ridge. Pour l'ensemble des méthodes, nous avons l'ensemble minimal $\{GDF5, CKAP4, CUBN, OTOF, SORBS2\}$ pour lequel leur pourcentage de sélection est supérieur à 0.25. Dans cet ensemble, nous avons les gènes $CKAP4$ qui semblait déjà être de bons biomarqueurs de la survie des patients dans la section 2.1 précédente. De plus, le gène $FBXL5$ évoqué ci-dessus comme bon candidat de biomarqueur est sélectionné par toutes les méthodes sauf pour la méthode Ridge. Enfin, le gène $CUBN$ semble également être un bon biomarqueur pour expliquer la survie des patients. Celui-ci est un récepteur situé sur le tissu épithélial de l'intestin et du rein. Il a été montré par Gremel et al. (2017) que $CUBN$ pouvait être un marqueur pronostique du cancer du rein à cellules claires.

2 Méthodes de Screening

		SIS	ISIS	PSIS	CoxCS
Immune-Checkpoints	Indice de Sørensen	0.9708	0.9936	0.9983	0.9974
	Nombre de gènes sélectionnés	2.57 (2.23)	4.29 (0.69)	8.58 (0.57)	7.98 (2.01)
	AIC	1953.37 (21.36)	1935.35 (3.66)	1961.22 (4.33)	1946.50 (9.92)
Sur les variables différentiellement exprimées	Indice de Sørensen	0.9905	0.9841	0.9662	0.8885
	Nombre de gènes sélectionnés	4.12 (1.57)	4.27 (1.02)	18.51 (5.11)	8.96 (1.47)
	AIC	1903.63 (7.78)	1895.25 (5.92)	1944.20 (5.27)	1960.39 (3.50)
Sur l'ensemble des gènes	Indice de Sørensen	0.9962	0.9988	0.9610	0.9341
	Nombre de gènes sélectionnés	5.80 (1.04)	5.39 (1.50)	27.21 (9.18)	25.85 (14.44)
	AIC	1873.80 (0.71)	1880.01 (24.69)	1931.38 (12.55)	1937.71 (13.69)

Table 2.4 – Résultats de l'indice de similarité (indice de Sørensen), de l'AIC moyen et du nombre moyen de gènes sélectionnés sur l'ensemble des graines pour l'ensemble des méthodes de *Screening* étudiées suivant l'ensemble de gènes donné en entrée. Leurs écart-types sont précisés dans les parenthèses.

Immune-Checkpoints

Comme pour les méthodes de régularisation, nous avons tout d'abord appliqué les méthodes de *Screening* sur la liste des checkpoints transmise par l'équipe de l'hôpital Saint-Louis/CEA pouvant avoir un impact sur le cancer du rein à cellules claires (ccRCC). Les résultats des méthodes SIS (Fan et al., 2010a), ISIS (Fan et al., 2010a), PSIS (Zhao et al., 2012) et CoxCS (Hong et al., 2018) sont présentés ici et les résultats de ces méthodes sur l'ensemble des variables est présentés en Annexe C. De plus, comme nous l'avons présenté en section 3 la méthode CoxCS utilise la connaissance biologique pour faire une pré-sélection. Nous avons décidé d'utiliser le gène $HLA.G$ comme connaissance biologique. Ce choix est justifié par l'évocation dans l'étude précédente que $HLA.G$ pouvait être une cible alternative intéressante lors d'une non-réponse au traitement $PD1/PDL.1$ pour le ccRCC.

	gènes sélectionnés			
SIS (AIC = 1953.37 ±21.36)	**ADORA1**(61%) $LILRB1$(1%) $TNFSF4$(3%)	**BST1**(54%) $S100A9$(1%) **HLA.G**(4%)	**CD276**(10%) $TNFRSF18$(61%)	**CTLA4**(61%) $TNFRSF4$(1%)
ISIS (AIC = 1935.35 ± 3.66)	**ADORA1**(100%) **TNFRSF18**(100%)	**BST1**(96%) $TNFSF4$(1%)	**CD276**(25%) **HLA.G**(7%)	**CTLA4**(100%)
PSIS (AIC = 1961.22 ±4.33)	**ADORA1**(100%) **CD276**(54%) **CD40LG**(100%)	**BAP1**(100%) **CD27**(100%)	**BST1**(4%) **CD28**(100%)	**BTLA**(100%) **CD39**(100%)
coxCS (AIC = 1946.50 ±9.92)	**ADORA1**(100%) $CD80$(59%) **CD28**(100%)	**BAP1**(100%) $CD86$(59%) $ENTPD1$(92%)	**BST1**(59%) **CD274**(100%)	$CD70$(34%) **CD276**(95%)

Table 2.5 – Résultats des méthodes de *Screening* sur les checkpoints. Les gènes sélectionnés avec un pourcentage de sélection de 100% sont en violet, les gènes en violet et soulignés sont ceux en commun entre les quatre méthodes et le gène HLA.G est mis en évidence en bleu (car il est d'intérêt pour les biologistes).

Nous pouvons voir sur la Table 2.4 que les méthodes de *Screening* ont de meilleurs résultats de stabilité en sélection que les méthodes de régularisation. C'est notamment le cas pour les méthodes SIS et ISIS. En revanche, ce n'est pas le cas sur ce jeu de données (*Immune-Checkpoints*). Les méthodes de *Screening* sélectionnent très peu de variables, notamment SIS et ISIS. La moyenne des variables sélectionnées pour SIS est de 2.57. Le Lasso sélectionne, au contraire, en moyenne 15.36 gènes. Le nombre de variables présentes dans ce jeu de données est faible et ces variables ont été choisies en amont car elles avaient un impact potentiel sur le cancer du rein à cellules claires. De nombreuses variables pourraient donc être corrélées entre elles et cela pourrait expliquer ce résultat de stabilité en sélection moins bon. De plus, la valeur moyenne de l'AIC est également élevée et un grand écart-type existe. La méthode ISIS a été développée pour résoudre ces problèmes de corrélation et les résultats obtenus vont dans cette direction. La valeur de l'indice de Sørensen est plus élevée et sa valeur moyenne de l'AIC est plus faible. Enfin, PSIS et CoxCS ont de meilleurs résultats de stabilité en sélection que les méthodes de régularisation. Mais les variables sélectionnées par PSIS et CoxCS semblent expliquer moins bien la durée de survie.

Aucun gène n'a un pourcentage de 100% pour la méthode SIS et le pourcentage le plus élevé est pour $ADORA1$ et $TNFRSF18$ qui sont sélectionnés 61 fois lorsque la méthode SIS est exécutée 100 fois. Nous pouvons également observer que les gènes sélectionnés par ISIS sont les mêmes. Seuls 3 gènes diffèrent $LILRB1$, $S100A9$ et $TNFSF4$, ils sont présents en plus dans les résultats de sélection de SIS (cf. Table 2.5). Nous pouvons également voir sur la Table 2.5 que $CTLA4$ est sélectionné par SIS, ISIS. Ce gène est déjà utilisé

dans le traitement du ccRCC. Nous pouvons également observer que $HLA.G$ est sélectionné par les méthodes SIS et ISIS. Comme évoqué ci-dessus, $CTLA4$ est déjà utilisé comme traitement pour le ccRCC. Cela pourrait confirmer le potentiel de $HLA.G$ comme nouvelle cible d'un traitement pour le ccRCC. De plus, quand $HLA.G$ et $CTLA4$ ne sont pas sélectionnés, les gènes de la superfamille récepteur *Tumor Factor Necrosis* ne le sont pas non plus. Enfin, ISIS a été développée pour résoudre le problème de performance de SIS quand certaines variables importantes ne sont pas corrélées avec la variable à expliquer. ISIS utilise donc l'information conjointe. À partir de la Table 2.5 nous pouvons penser que $HLA.G$ partage de l'information avec d'autres gènes et c'est pour cette raison que le pourcentage de sélection du gène $HLA.G$ par ISIS est plus forte que celui de SIS. Cela expliquerait également pourquoi il n'est pas sélectionné par PSIS et CoxCS. Pour CoxCS, $HLA.G$ est utilisé comme connaissance biologique. Cela provoquerait la sélection des gènes conjointement liés à $HLA.G$, mais celui-ci ne serait pas sélectionné car son impact sur la durée de survie est plus faible. L'augmentation du pourcentage de sélection de $CD276(B7\text{-}H3)$ dans CoxCS pourrait expliquer cela. Dans l'étude précédente en section 2.1, il avait déjà été évoqué que $HLA.G$ partage de l'information avec d'autres gènes sélectionnés. En effet, $HLA.G$ est un bon marqueur de la tolérance immunitaire, du pronostic et de la progression du cancer chez les patients (Lin et al., 2018). $HLA.G$ semble donc un bon marqueur pour expliquer la survie des patients, mais des études complémentaires devraient être réalisées afin de mettre en évidence les autres gènes interagissant avec lui.

Sur les variables différentiellement exprimés

Nous avons ensuite appliqué les méthodes de *Screening* sur les gènes considérés comme différentiellement exprimés entre les cellules tumorales et les cellules saines dans la cadre du ccRCC. Ce sont les résultats complets de l'analyse différentielle effectuée en section 2.1. Cette analyse a été exécutée à l'aide de la méthode DESeq2 (Love et al., 2014) et 11 289 gènes ont été trouvés différentiellement exprimés pour le ccRCC. Les résultats des méthodes de *Screening* sont présentés sur la Table 2.6 pour les gènes différentiellement exprimés. De plus, comme nous l'avons présenté en section 3 la méthode CoxCS utilise la connaissance biologique pour faire une pré-sélection. Pour cette pré-sélection, nous avons eu l'idée de nous servir des dernières publications parlant des marqueurs capables de prédire la survie des patients. Pour cela, nous avons effectué une recherche pubmed en utilisant les mots clés ainsi que les termes MeSH (pour *Medical Subject Headings*) suivants : `ccRCC[tiab] prognosis[tiab] survival[tiab] genes NOT RNA`. MeSH est un vocabulaire hiérarchiquement organisé et contrôlé, il est produit par la *National Library of Medicine*. Nous avons ainsi obtenu 32 résultats correspondant à notre recherche et nous avons décidé de garder les 10 premiers *abstracts*. Ces 10 *abstracts* ont été sauvegardés et chargés par la suite dans l'application web `beca annotate` (Nunes et al., 2013). Cette application web a pour but de permettre l'identification et l'annotation de concepts médicaux dans du texte. L'identification et l'annotation des gènes et de protéines sont réalisées à partir de méthodes de *machine learning* présentes dans Gimli (Campos et al., 2013). Gimli (Campos et al., 2013) est un outil *open-source* pour la reconnaissance automatique de termes biomédicaux. Cet outil est décrit en Annexe D. La liste des gènes obtenus est : $KDM2B$, $HMGCS2$, $HSD11B1$, $IL10$, $KDM5B$, $KDM5A$, $KDM5C$, $KDM5D$, $KDM1B$, $OGDHL$, $SSBP2$, $VSIG4$ et $XCR1$. Nous avons décidé d'utiliser cette liste comme connaissance biologique pour la pré-sélection dans CoxCS.

La première chose que nous pouvons observer sur la Table 2.4 est que la méthode SIS est la plus stable comparé aux autres méthodes (ISIS, PSIS et CoxCS). La stabilité de SIS et ISIS semble assez similaire entre elles, mais SIS semble avoir un léger avantage pour la stabilité en sélection avec une valeur d'indice de Sørensen égal à 0.9905. En revanche, la méthode ISIS semble sélectionner en moyenne un modèle expliquant mieux la survie. Nous pouvons également voir que CoxCS a le plus faible indice de Sørensen et est donc la méthode la moins stable. De plus, le modèle sélectionné en moyenne est celui qui explique le moins bien la survie. Sur ce jeu de données, l'ajout de la connaissance biologique ne semble pas améliorer les résultats. Nous pouvons

	gènes sélectionnés			
SIS (AIC = 1903.63 ± 7.78)	$AQP1$(1%)	$CENPF$(53%)	**CHEK2**(99%)	**EHHADH**(99%)
	GPR173(89%)	$SPC24$(1%)	$TPRG1L$(35%)	$ULBP3$(35%)
ISIS (AIC = 1895.25 ±5.92)	$CASP4$(1%)	$CD9$(1%)	$CENPF$(1%)	**CHEK2**(100%)
	$CNTNAP5$(2%)	**EHHADH** (100%)	$LCN1$(1%)	$RORA$(3%)
	$SDCBP2$(1%)	$TPRG1L$(1%)	$ULBP3$(1%)	
PSIS (AIC = 1944.20 ±5.27)	$AACS$(1%)	$ABHD15$(1%)	$ACCS$(1%)	$ACSF3$(1%)
	$ACSM3$(29%)	**ADAM17**(99%)	**ADAP1**(96%)	$AHNAK2$(4%)
	$ALDH1L1$(43%)	**ALG1**(99%)	$ANKRD28$(1%)	$ANKRD34A$(4%)
	$AOAH$(3%)	$AP2A2$(33%)	**APOBEC3D**(97%)	$APOL1$(1%)
	$ARHGAP12$(1%)	**ARHGAP39**(100%)	$ARMCX5$(1%)	$ASAH2$(10%)
	$ASB15$(1%)	ASL(91%)	$ASPHD1$(1%)	$ATP11A$(1%)
	$ATP6AP2$(1%)	$ATRX$(1%)	$AURKB$(1%)	$AVPR2$(1%)
	$B3GALT5$(57%)	$BAIAP2$(4%)	$BDKRB1$(1%)	**BIRC3**(99%)
	$BRI3$(1%)	$BRPF3$(1%)	$BRS3$(1%)	$C10orf82$(1%)
	C19orf54(100%)	$C1QA$(1%)	$C1QTNF4$(1%)	**CAMKK1**(98%)
	CASKIN1(98%)	**CASKIN2**(99%)	$CBR3$(1%)	**CCDC34**(100%)
	$CCDC73$(1%)	$CCKAR$(1%)	$CCL20$(57%)	$CCL21$(56%)
	$CCL22$(1%)	$CCNB1$(1%)	**CD209**(100%)	$CD33$(9%)
	CD59(100%)	$CDK16$(1%)	$CDKL1$(1%)	**CDNF**(93%)
	$CENPE$(1%)	$CEP152$(33%)	$CHCHD4$(4%)	$CHD1$(1%)
	$HDAC8$(1%)	$HESX1$(1%)	$HIPK1$(1%)	$HSPB6$(1%)
	$IL18BP$(1%)	$ILF3$(1%)	$IQSEC3$(1%)	$KANK4$(1%)
	$KIAA1257$(1%)	$KPNA2$(1%)	$LIN7C$(1%)	$LYPD6$(1%)
	MAG(1%)	MAK(1%)	$MARCO$(1%)	$MCEE$(1%)
	$MCF2L2$(1%)	$MEGF11$(1%)	$MFSD11$(1%)	$MPPE1$(1%)
	$MRPL9$(1%)			
coxCS (AIC = 1960.39 ±3.50)	**A4GNT**(100%)	$AAK1$(17%)	**AARS2**(91%)	**AASDH**(100%)
	$AASS$(71%)	**ABCA13**(91%)	**ABCA1**(100%)	**ABCA5**(100%)
	ABCA6(98%)	**ABCB6**(98%)	$ABCB7$(30%)	

Table 2.6 – Résultats des méthodes de *Screening* sur les gènes différentiellement exprimés Les gènes sélectionnés avec un pourcentage supérieur à 80% sont en violet, les gènes en violet et soulignés sont ceux en commun entre les méthodes SIS et ISIS.

observer que l'ensemble des gènes sélectionnés est plus petit pour les méthodes de *Screening* SIS et ISIS comparé à la taille des ensembles de gènes sélectionnés par les méthodes de régularisation (cf. Table 2.6). Mais il y a très peu gènes également qui ont des pourcentages élevés.

Ensuite, nous pouvons voir que le gène $CHEK2$ est sélectionné par SIS et ISIS. Dans la section 2.2.1, nous avons déjà évoqué qu'il pourrait être intéressant comme marqueur pour le pronostic de patients atteints de ccRCC. Enfin, nous pouvons remarquer que la connaissance biologique introduite dans la méthode de *Screening* a changé la sélection de gènes comparé aux autres méthodes SIS, ISIS et PSIS qui n'utilisent pas de connaissances. En effet, aucun des gènes sélectionnés dans CoxCS ne se retrouve dans SIS, ISIS ou PSIS (cf. Table 2.6). Il est plus difficile de conclure sur ces résultats car aucun des gènes n'est sélectionné à la fois par toutes les méthodes. De plus, la stabilité des méthodes est mauvaise pour l'ensemble des méthodes.

Sur l'ensemble des variables

Dans ce paragraphe, nous donnons les résultats de l'application des méthodes de *Screening* sur l'ensemble des gènes codants.

Nous pouvons observer sur la Table 2.4 que les méthodes SIS et ISIS ont de meilleures performances de stabilité quand la dimension augmente. Leurs indices de Sørensen valent respectivement 0.9962 et 0.9988. Leurs valeurs sont meilleures que celles obtenues sur l'ensemble des *Immmune-Checkpoints* et sur l'ensemble des gènes différentiellement exprimés. PSIS et CoxCS maintiennent de bons résultats de stabilité en sélection, leurs valeurs d'indice sont égales respectivement à 0.9610 et à 0.9341. Mais leurs valeurs moyennes de l'AIC sont plus élevées pour ces deux méthodes de *Screening* que celles des méthodes de régularisation. Les variables sélectionnées semblent moins pertinentes pour expliquer la survie. Les méthodes SIS et ISIS semblent plus stables en sélection que les méthodes de régularisation sur ce jeu de données. De plus, la qualité de la sélection de variables semble être équivalente entre ces deux méthodes et les méthodes de régularisation Lasso et Ridge. La méthode SIS a le meilleur AIC moyen avec une valeur de 1873.80 et un très faible écart-type (0.71).

Dans la sélection effectuée par ces méthodes, nous pouvons observer la présence des gènes $CHEK2$ et $CUBN$ pour SIS et ISIS. Ces gènes ont été évoqués en section 2.3 comme intéressants, ils étaient déjà sélectionnés par les méthodes de régularisation et leur rôle biologique semble être en relation avec la survie des individus. Le fait qu'ils soient aussi sélectionnés par ces méthodes renforce ce potentiel. Comme nous avons pu l'évoquer lors du paragraphe précédent, la pré-sélection (par la connaissance biologique) du modèle de CoxCS provoque une sélection de gènes très différente comparé aux méthodes SIS, ISIS (cf. Table C.2). PSIS a également une sélection de gènes très différente de SIS et ISIS comme nous avons pu déjà le remarquer pour l'étude sur les gènes différentiellement exprimés. Mais les sélections de CoxCS et PSIS diffèrent aussi entre elles. Cependant, les méthodes SIS et ISIS sont les plus stables des 4 méthodes et renforcent l'intérêt des gènes $CHEK2$ et $CUBN$.

3 Discussions

Dans cette étude, les méthodes de *Screening* permettent de donner une interprétation plus simple des résultats. Le nombre de variables sélectionnées par les méthodes de *Screening* est plus faible que celui des méthodes de régularisation. La méthode Elastic-Net sélectionne de nombreuses variables. Sur notre jeu de données, la pénalisation en norme l_2 de la procédure Elastic-Net a un poids plus important sur la pénalisation en norme l_1, ce qui provoque une sélection de variables plus grande et une interprétation plus complexe. De plus, les méthodes de régularisation sont réputées être instables en sélection et les méthodes de *Screening* ont été développées pour en partie répondre à ce problème. Les valeurs de l'indice de Sørensen pour les méthodes de *Screening* et de régularisation sont en accord avec cela, les valeurs de cet indice des méthodes de *Screening* sont plus élevées en grande dimension que celles des méthodes de régularisation. Un indice élevé signifie que les différentes sélections d'une méthode exécutée plusieurs fois se ressemblent et donc la méthode est stable. De plus, nous pouvons observer que la valeur des indices de Sørensen des méthodes de régularisation diminuent avec l'augmentation de la dimension, ce qui n'est pas le cas avec les méthodes de *Screening*. La méthode SIS performe mieux en sélection que la méthode Lasso sur notre jeu de données. En revanche, PSIS ne réalise pas de meilleures performances que SIS et a des résultats de stabilité en sélection similaires aux méthodes de régularisation. Nous pouvons observer le même comportement pour CoxCS et donc nous ne sommes pas parvenus à apporter de la connaissance biologique permettant d'améliorer les résultats de stabilité et de qualité de la sélection. Il serait pertinent d'étendre cette étude sur des données simulées afin de voir si les résultats issus du jeu de données ccRCC peuvent être généralisés. Enfin, la méthode de régularisation Ridge possède de bonnes performances de stabilité en sélection quand la dimension des données est grande et nous avons pu voir que les résultats, concernant la stabilité et la qualité de la sélection, étaient équivalents à la méthode SIS en grande dimension. Son indice de Sørensen était légèrement plus élevé et sa valeur moyenne de

l'AIC plus faible mais avec un écart-type plus fort. Ces résultats nous montrent le potentiel de la méthode Ridge pour faire de la sélection de variables en grande dimension en prenant un top de gènes ce qui n'est pas souvent évoqué. Il serait donc intéressant d'approfondir l'étude afin de voir si son comportement avec un nombre différent de gènes sélectionnés peuvent améliorer les résultats. Enfin, nous avons remarqué que l'indice de Sørensen avait tendance à indiquer comme plus stables les scénarios de sélections emboîtés, même si le nombre de variables varie, alors que ce type de scénario est pénalisé par l'indice de Jaccard (cf. Annexe B). Nous avons préféré l'indice de Sørensen à l'indice de Jaccard car il est beaucoup plus stable. Cela est dû au fait qu'on divise par le nombre total de gènes sélectionnés (qui peut grandement varier d'une expérience à l'autre) pour l'indice de Jaccard, alors qu'on divise par la moyenne du nombre de gènes sélectionnés pour l'indice de Sørensen. Les résultats de l'indice de Jaccard pour les méthodes de régularisation et de *Screening* sont présentés dans les Tables B.2 et B.3 en Annexe B. Un indice de sélection plus adapté serait sans doute à définir. D'ailleurs, cet indice de régularité plus grand ne se traduit pas forcément par un meilleur comportement du modèle en termes d'AIC.

L'étude de la sélection de variables pour expliquer la survie des patients à partir des *Immune-Checkpoints* montre que certains seraient intéressants à étudier de manière plus poussée. En effet, le gène $B7\text{-}H3(CD276)$ semblerait être un bon biomarqueur pour expliquer la survie des patients. De plus, $HLA.G$ semble également être un marqueur potentiel pour la survie des patients. Nous avons pu voir dans l'étude de cette section ainsi que dans la section 2.1.4 qu'il partageait de l'information avec d'autres gènes. Dr Diana Le Roux Tronik évoque la possibilité que ce couple $HLA.G/ILT2$ puisse être une alternative aux traitements $PD1/PDL1$ quand les patients ne répondent pas à ce dernier. La perspective de son équipe est d'étudier de manière plus approfondie ce couple.

En élargissant l'étude de sélection de variables aux gènes différentiellement exprimés et à l'ensemble des gènes, nous avons pu observer des gènes qui pourraient être potentiellement intéressants. Ils seraient pertinents d'approfondir leur étude. Les gènes $CHEK2$, $CKAP4$ semblent des marqueurs importants pour expliquer la survie. Le gène $CHEK2$ est déjà connu pour avoir un impact dans le cancer du sein et le gène $CKAP4$ est impliqué dans le système immunitaire. Nous rappelons que le cancer du rein à cellules claires est un cancer immunogène, ce qui signifie qu'il est capable d'arrêter le système immunitaire. Un autre gène qui apparaît dans la sélection pour expliquer la survie des patients est le gène $CUBN$, d'après une étude de Gremel et al. (2017), ce gène semble jouer un rôle dans le cancer du rein à cellules claires. Enfin, le gène $FBXL5$ semble être également un marqueur potentiel pour expliquer la survie des patients car il a un rôle dans le système immunitaire, est impliqué dans des maladies chroniques du rein et serait lié à deux *Immune-Checkpoints* listés dans l'étude de la section 2.1.

Chapitre 3

Étude des réseaux de neurones pour la prédiction de survie

3.1 Introduction

Le but de ce chapitre est d'étudier des nouvelles méthodes basées sur les réseaux de neurones pour prédire la durée de survie des patients en grande dimension et de les comparer à des méthodes classiques comme le modèle de Cox. Le modèle de Cox (Cox, 1972) est le modèle de référence dans le domaine de l'analyse de survie. Celui-ci permet de relier la durée de survie d'un individu pour une maladie donnée aux variables explicatives. Ce modèle permet également de prendre en compte les données censurées à droite. Avec l'arrivée des techniques de séquençage à haut-débit, les données transcriptomiques sont de plus en plus utilisées comme variables dans l'analyse de survie. L'ajout de ces variables explicatives amène de nouvelles problématiques car on se trouve dans le cadre de la grande dimension.

Le *deep learning* est un sujet de recherche très populaire depuis quelques années dans un certain nombre de champs d'applications comme la santé, la finance ou la physique. L'intérêt du domaine biomédical pour ce type de méthodes n'a cessé de croître. Une des applications médicales, où les réseaux de neurones ont montré leur capacité et sont donc devenus d'une grande importance, est l'imagerie. En effet, une recherche bibliographique sur le site pubmed (`https://pubmed.ncbi.nlm.nih.gov/`) avec les mots clés `deep learning` et `medical imaging` affiche plus de 3 938 résultats. Nous pouvons également observer qu'en 1975 un seul article avait été publié contre 1 627 articles en 2019. La recherche dans ce domaine a fait un réel bond depuis 2016. Les autres sphères médicales se passionnent à leur tour pour le *deep learning* et c'est notamment le cas de la génomique et de l'analyse de survie.

Ces dernières années, deux stratégies différentes ont été développées en analyse de survie. La première est un réseau de neurones basé sur la log-vraisemblance partielle de Cox comme celui développé par Faraggi et al. en 1995 et adapté au *deep learning* par Ching et al. en 2018 et par Katzman et al. en 2018. De plus, Kvamme et al. (2019b) a récemment proposé un réseau de neurones basé sur une extension du modèle de Cox permettant de s'affranchir de la restriction des risques proportionnels en introduisant la variable temps en entrée du réseau de neurones. Il a proposé une nouvelle fonction de perte qui peut gérer à la fois les risques proportionnels et non-proportionnels.

La seconde stratégie consiste à développer un réseau de neurones basé sur un modèle de survie à temps discret (Brown et al., 1997 ; Biganzoli et al., 1998 ; Rodrigo et al., 2017 ; Gensheimer et al., 2018). L'avantage de cette stratégie est l'estimation directe du risque instantané contrairement à une stratégie basée sur la log-vraisemblance partielle de Cox où apparaissent seulement les coefficients de régression et une seconde étape

doit être réalisée pour obtenir le risque instantané en calculant le risque de base. Nous avons choisi d'étudier plus en détails les réseaux de neurones élaborés à partir de modèles à temps discret qui n'ont pas ou peu été utilisés en grande dimension et de comparer leurs performances à ceux des réseaux basés sur le modèle de Cox. Plusieurs réseaux de neurones à partir d'un modèle à temps discret ont été proposés, ces réseaux de neurones diffèrent par l'architecture, la pénalisation utilisée dans la fonction de perte ou encore la définition de l'indicateur de censure. Biganzoli et al. (1998) a proposé une seule sortie, mais les individus sont répliqués pour chaque intervalle de temps. Cette structure de réseau de neurones nous a semblé pertinente dans notre contexte de grande dimension. Liestbl et al. (1994), Street (1998) et Mani et al. (1999) ont proposé un réseau de neurones avec la même structure, mais leurs réseaux de neurones possèdent L sorties. L'indicateur de censure n'est pas défini de la même manière pour chacun des réseaux de neurones de Liestbl et al. (1994), Mani et al. (1999) et Street (1998). Liestbl et al. (1994) utilise le même indicateur de censure que celui de Biganzoli et al. (1998) et Mani et al. (1999) et Street (1998) utilisent un indicateur de censure basé sur l'estimateur de Kaplan-Meier. La différence entre les réseaux de neurones de Mani et al. (1999) et Street (1998) est que le premier prédit le risque instantané alors que le second prédit la fonction de survie. Enfin, Mani et al. (1999) et Street (1998) ne proposent pas de pénalisation de la fonction de coût contrairement à Biganzoli et al. (1998) et Liestbl et al. (1994). Biganzoli et al. (1998) propose une pénalisation ridge et Liestbl et al. (1994) propose une pénalisation de type fused-lasso. Nous détaillons les différentes approches en section 3.2.3. Nous présentons par la suite le réseau de neurones que nous avons développé en se basant sur les différentes approches proposées par Biganzoli et al. (1998), Mani et al. (1999) et Liestbl et al. (1994). Nous comparons ces deux procédures, celle de Biganzoli et al. (1998) et la notre à d'autres méthodes : estimation classique des paramètres du modèle de Cox par maximisation de la log-vraisemblance pénalisée et procédure Cox-nnet (Ching et al., 2018) qui est un réseau de neurones basé sur le modèle de Cox et qui utilise donc la vraisemblance partielle de Cox.

De plus, l'étude des différentes approches de ces réseaux de neurones a été approfondie à partir d'un plan de simulations. Celui-ci se divise en trois grandes parties. Pour générer les données, nous nous sommes tout d'abord appuyés sur le modèle de Cox en se basant sur Bender et al. (2005). Simuler des données à partir d'un modèle de Cox dans cette étude avantage évidemment les deux méthodes comparées dans ce chapitre qui sont basées sur le modèle de Cox. Mais il était intéressant d'étudier comment se comportait notre méthode dans cette situation. Nous avons aussi choisi d'étendre notre étude à d'autres modélisations comme le modèle AFT (Kalbfleisch et al., 2002) présenté en section 1.2.4 et le modèle AH (Chen et al., 2000) présenté en section 1.2.5. Pour générer les données à partir d'un modèle AFT, nous nous sommes basés sur Leemis et al. (1990) et nous avons suivi la même démarche pour simuler les données à partir d'un modèle AH. Simuler des données de survie à partir de différents modèles est intéressant pour notre étude car les caractéristiques et hypothèses sous-jacentes des modèles diffèrent. Le modèle de Cox est un modèle à risques proportionnels, ce qui n'est pas le cas pour les modèles AFT et AH. De plus, les courbes des risques instantanées des individus se croisent pour le modèle AFT et AH, mais pas pour le modèle de Cox et les courbes de survie des individus se croisent dans le modèle AH, mais pas dans le modèle AFT et Cox. Nous commençons par présenter les réseaux de neurones en section 3.2 et nous introduisons par la suite les différentes approches des réseaux de neurones en analyse de survie . En section 3.3, nous présentons le plan de simulations utilisé dans l'étude des réseaux de neurones et nous terminons par montrer les résultats de l'étude sur des données simulées et réelles en sections 3.4 et 3.5 respectivement.

3.2 Réseaux de neurones

Plusieurs types de réseaux de neurones existent et celui considéré dans cette thèse est du type perceptron multi-couches (MLP). Ce réseau de neurones est constitué de plusieurs couches avec une couche d'entrée, au moins une couche cachée et une couche de sortie (cf.Figure 3.1). Chaque couche est constituée de plusieurs

neurones et chaque neurone joue le rôle d'une régression non-linéaire entre ses entrées et sa variable de sortie. Les coefficients pondérant les entrées dans ces régressions sont appelés poids, et la fonction non-linéaire de transformation de cette combinaison pour donner la sortie est appelée fonction d'activation. La sortie d'un neurone va être l'entrée d'un neurone en aval et c'est cette association qui va constituer le réseau. De plus, les neurones de la même couche n'ont pas de connexions entre eux. Ils possèdent seulement des connexions avec les neurones des couches précédentes et suivantes. Soient $X_1, \ldots, X_p$ les variables d'entrées. Un réseau de neurones MLP réalise donc une transformation des variables d'entrées :

$$Y = g(X_1, \ldots, X_p, W, b), \tag{3.1}$$

où g est la fonction d'activation, W, b sont les poids et les biais du réseau de neurones à estimer et Y est la variable à expliquer. Différentes fonctions d'activation existent, les plus utilisées sont les fonction linéaires (*i.e.* f est la fonction identité), les fonctions ReLU (pour *Rectified Linear Unit*), les fonctions sigmoïdes (ou logistiques) et les fonctions softmax. Cette dernière fonction est utilisée dans le cadre de la classification. Dans cette thèse, nous nous intéressons aux réseaux de neurones dans le cadre de la régression. De plus, les réseaux de neurones étudiés dans ce manuscrit utilisent soit une fonction ReLU soit une fonction sigmoïde soit les deux comme fonctions d'activation. La fonction ReLU est définie de la manière suivante :

$$f(x) = \max(0, x), \tag{3.2}$$

et la fonction sigmoïde est définie de la façon suivante :

$$f(x) = \frac{1}{1 + \exp(-x)}. \tag{3.3}$$

La sortie d'un réseau de neurones avec P variables d'entrées possédant une seule couche cachée avec H neurones sera donc :

$$\hat{h}_i = \hat{h}(X_i; W, b) = f_2\left(b + \sum_{k=1}^{H} W_k f_1\left(b_k + \sum_{j=1}^{P} W_{jk} X_{ij}\right)\right), \tag{3.4}$$

où f_1 et f_2 sont respectivement les fonctions d'activation de la couche cachée et de la couche de sortie et W et b sont respectivement les poids et les biais du réseau de neurones estimés. L'estimation des paramètres W et b du réseau de neurones est réalisée pendant l'étape appelée apprentissage. L'apprentissage consiste à minimiser une fonction de perte, souvent celle des moindres carrés en régression :

$$E = \frac{1}{n} \sum_{i=1}^{n} \left[Y_i - \hat{h}(X_i; W, b)\right]^2. \tag{3.5}$$

Différents algorithmes d'optimisation sont proposés et la plupart d'entre eux sont basés sur l'évaluation du gradient par rétro-propagation. La rétro-propagation de l'erreur consiste à calculer les dérivées de la fonction de perte en une observation et par rapport aux différents paramètres. Les gradients peuvent être écrits à partir de termes d'erreurs et ces termes sont évalués en deux étapes. La première étape, appelée passe avant, consiste à calculer la sortie $\hat{h}_i$ du réseau de neurones à partir des valeurs actuelles des paramètres et la seconde étape, appelée passe retour, consiste à déterminer les termes d'erreurs et ainsi calculer les gradients. L'algorithme le plus connu est l'algorithme du descente de gradient. Celui-ci est un algorithme itératif modifiant les poids

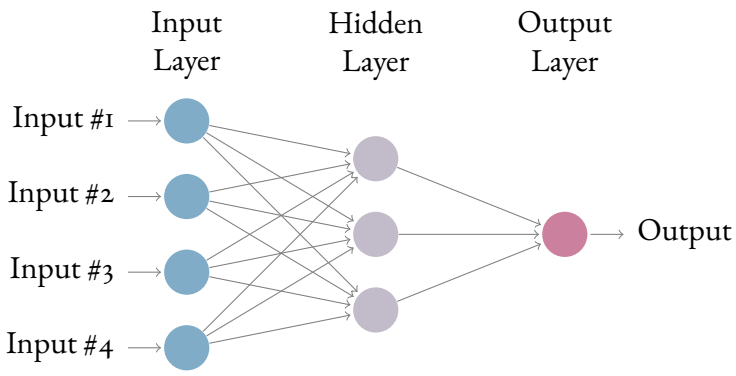


Figure 3.1 – Schéma d'un réseau de neurones de type perceptron multi-couches (MLP)

du réseau de neurones de la façon suivante :

$$b_k^{(r+1)} = b_k^{(r)} - \tau \frac{1}{n} \sum_{i=1}^{n} \frac{\partial E_i}{\partial b_k^{(r)}} \tag{3.6}$$

$$W_{jk}^{(r+1)} = W_{jk}^{(r)} - \tau \frac{1}{n} \sum_{i=1}^{n} \frac{\partial E_i}{\partial W_{jk}^{(r)}}, \tag{3.7}$$

avec τ est un hyperparamètre appelé taux d'apprentissage. Il peut être fixé par l'utilisateur ou varier lors de l'exécution du réseau de neurones. Souvent, le gradient de la fonction de perte est approché par le gradient de celle-ci prise seulement sur un mini-batch (Bottou, 1999). Comme nous l'avons présenté en section 3.1, deux modélisations différentes des réseaux de neurones existent en analyse de survie. La première est basée sur la modélisation de Cox comme par exemple Cox-nnet présenté en section 3.2.1. La seconde stratégie concerne une modélisation à partir de temps discrets présentée en section 3.2.2.

3.2.1 Cox-nnet

Cox-nnet est un réseau de neurones développé par Ching et al. en 2018, basé sur le modèle de Cox. Un réseau de neurones basé sur le modèle à risques proportionnels a été initialement développé par Faraggi et al. (1995). L'idée de Faraggi et al. (1995) consiste à remplacer la prédiction linéaire de la régression de Cox par la sortie de la couche cachée du réseau de neurones. Ching et al. (2018) ont repris ce principe en l'étendant dans le cadre du *deep learning*. En parallèle de Cox-nnet, un réseau de neurones similaire a été développé par Katzman et al. (2018). Ce réseau de neurones, appelé DeepSurv, adapte également l'idée de Faraggi et al. (1995) en utilisant le *deep learning*. Nous avons évoqué en Introduction que les performances de calculs des ordinateurs dans les années 90 ne permettaient pas de développer des réseaux de neurones avec des architectures complexes. Il a fallu attendre que les performances des ordinateurs augmentent avant de voir certains auteurs s'intéresser de nouveau à ces méthodes. En plus des faibles performances de calculs dans les années 90, le nombre de variables utilisées en analyse de survie était faible. Le séquençage à haut-débit commençait seulement et seules les variables cliniques étaient prises en compte pour étudier la durée de survie des individus. L'approche de Faraggi et al. (1995) n'avait donc pas été étudiée en grande dimension, contrairement à Ching et al. (2018) qui a appliqué la méthode Cox-nnet à des données de grande dimension. C'est pour cette raison que nous avons préféré nous concentrer sur le réseau Cox-nnet de Ching et al. (2018) plutôt que sur DeepSurv de Katzman et al. (2018) pour comparer ce réseau aux réseaux basés sur un modèle à temps discrets en grande dimension. Le principe de Cox-nnet est que sa couche de sortie correspond à une régression de Cox. Dans le modèle Cox-nnet, les variables présentes dans l'exponentielle de l'équation du

modèle de Cox sont remplacés par la sortie de la couche cachée :

$$\exp(\beta^T G(WX_{i.} + b)), \tag{3.8}$$

avec W la matrice des poids, b est le terme de biais pour chaque nœud caché et G est la fonction d'activation. Dans ce réseau, la fonction d'activation $tanh$ est utilisé et ils ont ajouté un terme de régularisation *ridge* dans la log-vraisemblance partielle de Cox. Ils emploient aussi un *dropout* pour réduire le sur-apprentissage.

La sortie de ce réseau de neurones est :

$$\hat{h}_i = \exp\left(\underbrace{\sum_{h=1}^{H} \beta_h G\left(b_h + \sum_{p=1}^{P} W_{ph} X_{ip}\right)}_{\theta_i = G(W^T X_{i.} + b)^T \beta}\right), \tag{3.9}$$

avec $\theta_i = G(WX_{i.} + b)^T \beta$ où G la fonction d'activation de la couche cachée ($tanh$), W les poids du réseau de neurones et β, b les biais du réseau de neurones à estimer.

Pour estimer les poids du réseau de neurones, Ching et al. (2018) utilise la log-vraisemblance partielle de Cox :

$$\mathcal{L}(\beta, W, b) = \sum_{i=1}^{n} \theta_i - \sum_{i=1}^{n} \delta_i \log\left(\sum_{l \in R_i} \exp(\theta_l)\right), \tag{3.10}$$

avec δ_i l'indicateur de censure, $\theta_i = G(WX_{i.} + b)^T \beta$ où G est la fonction d'activation de la couche cachée, β, W et b sont les paramètres du réseau de neurones. De plus, Ching et al. (2018) ajoute une pénalisation ridge. La fonction de coût pour ce réseau de neurones est :

$$Loss(\beta, W, b) = \mathcal{L}(\beta, W) + \lambda(||\beta||_2 + ||W||_2 + ||b||_2). \tag{3.11}$$

La sortie du réseau de neurones $\hat{h}_i$ correspond à la partie de la régression de Cox qui ne dépend pas du temps. Si l'on souhaite obtenir le risque $\hat{h}(x_i, t)$, une seconde étape doit être effectuée pour estimer le risque de base $\alpha_0(t)$. Ching et al. (2018) ne s'intéresse pas au risque instantané, ils utilisent juste la partie exponentielle de la régression de Cox. Notre étude s'intéresse à la prédiction de la durée de survie et pour cela, nous avons besoin du risque instantané. Pour estimer le risque de base $\alpha_0(t)$, nous avons donc utilisé un estimateur à noyau (Ramlau-Hansen, 1983). L'estimation de $\alpha_0(t)$ a été présentée en section 1.2.3 et cet estimateur est défini par :

$$\widehat{\alpha}_b(t) = \frac{1}{nb} \sum_{i=1}^{n} K\left(\frac{t-u}{h}\right) \frac{\delta_i}{\sum_{l \in R(t_i)} \widehat{h}_l}, \tag{3.12}$$

avec $\hat{h}_l$ est l'estimateur des poids obtenu en maximisant la fonction de coût (3.11), $K : \mathbb{R} \to \mathbb{R}$ une fonction d'intégrale 1, appelée noyau et h est un paramètre réel strictement positif, appelé fenêtre. Nous avons utilisé la méthode Goldenshluger & Lepski (Goldenshluger et al., 2011) pour sélectionner la fenêtre h. Nous pouvons finalement en déduire un estimateur de la fonction de survie :

$$\widehat{S}(t|X_i) = \exp\left(-\int_0^t \widehat{\alpha}_b(s) \exp(\widehat{\beta}^T X_{i.}) ds\right),$$

permettant de faire la prédiction de survie d'un individu i.

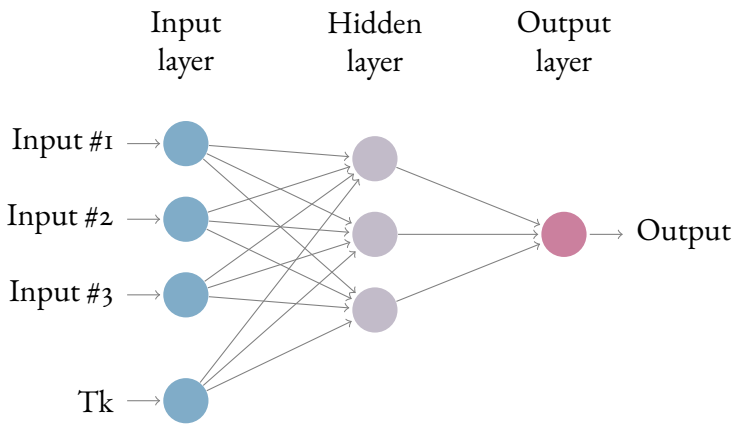


Figure 3.2 – Structure du réseau de neurones basé sur un modèle à temps discret proposée par Biganzoli et al. (1998)

3.2.2 Réseau de neurones à partir d'un modèle à temps discret (approche 1)

Comme nous l'avons présenté précédemment, deux stratégies existent pour modéliser les données de survie à partir d'un réseau de neurones MLP (perceptron multi-couches). Biganzoli et al. (1998) a proposé en 1998 un réseau de neurones basé sur un modèle à temps discret. Il introduit L intervalles de temps $A_l =]t_{l-1}, t_l]$ auxquels appartiennent les temps de survie. La fonction discrète du risque instantané s'écrit comme la probabilité conditionnelle de survie :

$$h_{il} = P(T_i \in A_l | T_i > t_{l-1}), \tag{3.13}$$

avec T_i le temps de survie de l'individu i.

Nous avons précisé précédemment que les individus sont dupliqués à l'entrée du réseau de neurones proposé par Biganzoli et al. (1998), cela lui confère une structure plus originale que celle d'un perceptron multi-couches classique. Le réseau de neurones de Biganzoli et al. (1998) prend en entrée l'ensemble des variables auquel l'individu est soumis et une variable supplémentaire correspondant au point moyen de chaque intervalle. L'ajout de cette variables amène à dupliquer les individus. Les p variables de chaque individu sont répétées pour chaque intervalle de temps. La sortie est donc le risque instantané estimé $\hat{h}_{il} = h_l(X_i, a_l)$ pour l'individu i au temps a_l. La structure de ce réseau de neurones est schématisée sur la Figure 3.2.

Biganzoli et al. (1998) utilisait initialement un réseau de neurones à 3 couches avec une fonction logistique à la fois comme fonction d'activation de la couche cachée et de la couche de sortie. La sortie du réseau de neurones avec H neurones dans la couche cachée et P variables d'entrée est donnée par :

$$\widehat{h}_{il} = \hat{h}(x_i, t_l) = f_2\left(b + \sum_{h=1}^{H} w_h f_1\left(a_h + \sum_{p=1}^{P} w_{ph} x_{ip}\right)\right),$$

où w_{ph} et w_h sont les poids du réseau de neurones et b et a_h sont les biais du réseau de neurones à estimer et f_1 et f_2 les fonctions d'activations logistiques. La cible de ce réseau de neurones est l'indicateur de décès d_{il}, il va indiquer si l'individu i est décédé dans l'intervalle A_l. On introduit $l_i \leq L$ qui est le nombre d'intervalles dans lequel l'individu i est observé, $d_{i0}, \ldots, d_{i(l_i-1)} = 0$ quel que soit le statut de l'individu i et d_{il_i} va être égal à 0 si l'individu i est censuré et 1 sinon. La fonction de coût utilisée par Biganzoli et al. (1998) est la fonction de cross-entropie et les poids du réseau de neurones peuvent être estimés en minimisant son

opposé :

$$\mathcal{L}(W) = -\sum_{i=1}^{n}\sum_{l=1}^{l_i} d_{il}\log(\widehat{h}_{il}) + (1-d_{il})\log(1-\widehat{h}_{il}). \tag{3.14}$$

Après l'estimation des paramètres du réseau de neurones, la sortie obtenue est l'estimation du risque discret $\widehat{h}_{il}$ pour chaque individu i. Une fois que le risque instantané $\widehat{h}_{il}$ est estimé, la fonction de survie de l'individu i est estimée en utilisant :

$$\widehat{S}(T_{l_i}) = \prod_{i=1}^{l_i}(1-\widehat{h}_{il}). \tag{3.15}$$

L'avantage de cette approche est que les individus sont dupliqués pour chaque intervalle de temps, ce qui augmente la taille d'échantillon dans le réseau de neurones. De plus, Biganzoli et al. (1998) ont ajouté une pénalisation ridge dans leur fonction de cross-entropie (3.14) :

$$Loss(W) = \mathcal{L}(W) + \lambda||W||_2, \tag{3.16}$$

λ est choisi par validation croisée, $\mathcal{L}$ est la fonction de cross-entropie et W est le vecteur des paramètres du réseau de neurones.

3.2.3 Réseau de neurones à partir d'un modèle à temps discret (approche 2)

Dans cette section, nous présentons le développement d'un réseau de neurones, également basé sur un modèle à temps discret comme pour la première approche, mais dont la structure du réseau de neurones et l'optimisation des paramètres diffèrent. Pour cette seconde approche, nous avons changé le nombre de sorties du réseau de neurones. Au lieu d'avoir une seule sortie, le réseau de neurones possède L sorties qui correspondent au nombre d'intervalles de temps. Avant de présenter notre approche, je vais détailler les réseaux de neurones proposés par Liestbl et al. (1994) et Mani et al. (1999).

Mani et al. (1999) et Liestbl et al. (1994) ont également proposé un réseau de neurones à temps discret avec L sorties, mais avec des différences en ce qui concerne la pénalisation et l'indicateur de censure. Tout d'abord, Mani et al. (1999) a proposé une modification de l'indicateur de censure. La modification proposée est elle-même basée sur celle utilisée initialement par Street (1998) pour un réseau de neurones à temps discret dont l'objectif est de prédire la fonction de survie. La variante de l'indicateur de décès proposé par Street (1998) et Mani et al. (1999) est inspirée de l'estimateur de Kaplan-Meier (présenté en section 1.2.2). Mani et al. (1999) utilise l'estimateur de la probabilité : $p_l = \mathbb{P}(T \leq t_l | T > t_{l-1})$. Cette probabilité p_l correspond à la probabilité de mourir dans l'intervalle A_l sachant que l'on était vivant au début de l'intervalle A_l. L'estimateur de la probabilité p_l est :

$$\hat{p}_l = \frac{r_l}{n_l}, \tag{3.17}$$

avec r_l le nombre de décès dans l'intervalle A_l et n_l le nombre d'individus à risque dans l'intervalle A_l. Street (1998) utilise l'estimateur de Kaplan-Meier de la fonction de survie, présenté en section 1.2.2. L'idée derrière cet estimateur provient du fait que survivre après un temps t_l signifie être en vie juste avant t_{l-1} et ne pas mourir (ou survivre) au temps t_l, autrement dit :

$$\mathbb{P}(T > t_l) = \underbrace{\mathbb{P}(T > t_{l-1})}_{S(t_{l-1})}\underbrace{\mathbb{P}(T > t_l | T > t_{l-1})}_{1-p_l}.$$

La probabilité de survivre après un temps t_l correspond à la probabilité de survivre pendant l'intervalle de

temps A_{l-1} multiplié par la probabilité de survivre pendant l'intervalle de temps A_l sachant qu'on était vivant au début de cet intervalle. Cependant, la sortie attendue de nos réseaux de neurones est le risque intantané et non la fonction de survie. Nous utilisons donc l'indicateur proposé par Mani et al. (1999). Ce nouvel indicateur de censure $\tilde{d}_{il}$ est toujours la cible qui sert à entraîner le réseau de neurones. On a toujours $l_i \leq L$ qui est le nombre d'intervalles dans lequel l'individu i est observé, $\tilde{d}_{i0}, \ldots, \tilde{d}_{i(l_i-1)} = 0$ quel que soit le statut de l'individu i et $\tilde{d}_{il_i}, \ldots, \tilde{d}_{iL}$ va être égal à $\frac{r_l}{n_l}$ si l'individu i est censuré et 1 sinon. $\frac{r_l}{n_l}$ est l'estimation du risque instantané pour l'intervalle de temps A_l, où r_l est le nombre d'individus décédés dans l'intervalle A_l et n_l est le nombre d'individus à risque dans l'intervalle A_l. Pour entraîner le réseau de neurones, Mani et al. (1999) utilise une fonction d'activation logistique à la fois pour la couche cachée et la couche de sortie. Les sorties du réseau de neurones avec H neurones dans la couche cachée et P variables d'entrée sont données par :

$$\widehat{h}_{il} = \hat{h}(x_i, t_l) = f\left(b_l + \sum_{h=1}^{H} w_{hl} f\left(a_h + \sum_{p=1}^{P} w_{ph} x_{ip}\right)\right),$$

où w_{ph} et w_{hl} sont les poids du réseau de neurones et b_l et a_h sont les biais du réseau de neurones à estimer et f la fonction d'activation logistique. Les paramètres du réseau de neurones sont estimés à partir de la fonction de coût (3.14) modifiée :

$$\mathcal{L}(W) = -\sum_{i=1}^{n} \sum_{l=1}^{l_i} \tilde{d}_{il} \log(\widehat{h}_{il}) + (1 - \tilde{d}_{il}) \log(1 - \widehat{h}_{il}). \tag{3.18}$$

À la fin de l'entraînement du réseau de neurones, le risque instantané est estimé pour les L intervalles de temps. On convertit les risques estimés en survie estimée de la même manière que celle réalisée dans le réseau de neurones de Biganzoli et al. (1998) en utilisant (3.15). La structure de ce réseau de neurones est schématisée sur la Figure 3.3.

Liestbl et al. (1994) a également proposé un réseau de neurones avec L sorties. Mais Liestbl et al. (1994) n'utilise pas le même indicateur de décès que celui proposé par Mani et al. (1999). Il utilise celui de Biganzoli et al. (1998). La structure du réseau de neurones est donc la même que celle de Mani et al. (1999) (cf.Figure 3.3), mais la fonction de coût diffère. La fonction de coût du réseau de neurones est la même que celle de l'équation (3.14) qui correspond au réseau de neurones proposé par Biganzoli et al. (1998). Cependant, Liestbl et al. (1994) n'a pas utilisé la pénalisation ridge dans la fonction de coût, il propose une pénalisation de type fused-lasso. En utilisant ce type de pénalisation, les poids w_{hl} $\forall l$ auront tendance à être égaux, ce qui signifie que l'importance du neurone h sera la même pour tous les intervalles de temps. Autrement dit, l'idée de cette pénalisation est d'obtenir peu de variations entre les poids sortants du neurone h. Cette contrainte peut être écrite :

$$w_{hl_1} \simeq w_{hl_2} \simeq \ldots \simeq w_{hl_L}. \tag{3.19}$$

Un réseau de neurones basé sur un modèle à temps discret sans la contrainte (3.19) équivaut à un modèle non linéaire avec des risques non proportionnels, c'est-à-dire que l'importance des variables pourrait changer au cours du temps. L'hypothèse de proportionnalité est restrictive mais elle peut sembler raisonnable en analyse de survie. Les paramètres du réseau de neurones sont donc estimés à partir de la fonction de coût :

$$Loss(W) = \mathcal{L}(W) + \lambda \sum_{l} \left|W_l - W_{(l-1)}\right|_1, \tag{3.20}$$

avec $\mathcal{L}(W)$ la fonction de cross-entropie (définie à l'équation (3.14)), W les paramètres du réseau de neurones à estimer et λ est un hyperparamètre choisi par validation croisée. À la fin de l'entraînement du réseau de

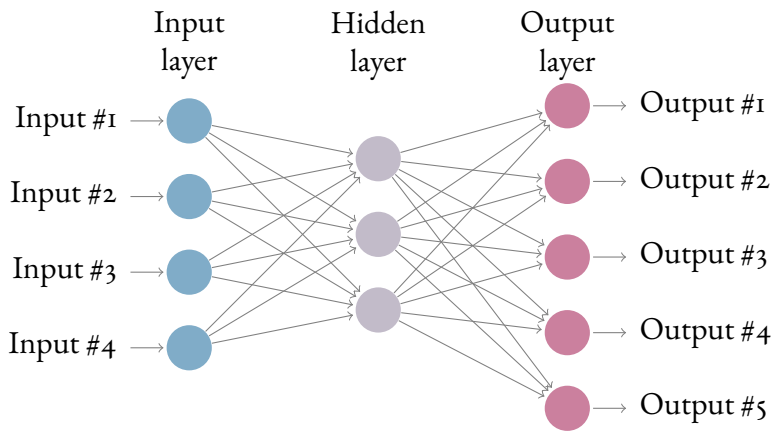


Figure 3.3 – Structure du réseau de neurones basé sur un modèle à temps discret avec une sortie multivariée

neurones, le risque instantané est estimé pour les L intervalles de temps. On convertit les risques estimés en survie estimée de la même manière que pour le réseau de neurones de Biganzoli et al. (1998) en utilisant (3.15).

Le réseau de neurones que nous développons possède L sorties et a donc la même structure que celle présentée en Figure 3.3. L'optimisation des paramètres du réseau de neurones est réalisée à partir de la fonction de coût (3.18) initialement proposée par Mani et al. (1999). Nous utilisons l'indicateur de décès basé sur l'estimateur de Kaplan-Meier. Nous ajoutons à la fonction de coût une pénalisation fused-lasso, elle s'écrit donc de la façon suivante :

$$Loss(W) = -\sum_{i=1}^{n}\sum_{l=1}^{l_i} \tilde{d}_{il}\log(\widehat{h}_{il}(W)) + (1-\tilde{d}_{il})\log(1-\widehat{h}_{il}(W)) + \lambda\sum_{h}\sum_{l}\left(W_{hl} - W_{h(l-1)}\right)^2, \tag{3.21}$$

avec λ un hyper-paramètre obtenu par validation croisée. À la fin de l'entraînement du réseau de neurones, le risque instantané est estimé pour les L intervalles de temps. On convertit les risques estimés en survie estimée de la même manière que celle réalisée dans le réseau de neurones de Biganzoli et al. (1998) en utilisant (3.15).

Nous étudions dans cette thèse plusieurs stratégies de réseaux de neurones pour modéliser les données de survie en grande dimension. Dans ce but, nous avons testé l'approche proposée par Biganzoli et al. (1998) que nous avons appelée NNsurv. Nous avons implémenté une procédure de validation croisée pour cette approche afin de choisir les hyperparamètres du réseau de neurones. Dans un second temps, nous avons développé un réseau de neurones avec plusieurs sorties dont chacune d'entre elles correspond aux intervalles de temps en y ajoutant un indicateur de censure basé sur l'estimateur non paramétrique du risque instantané et une pénalisation de type fused-lasso. Nous avons appelé cette approche NNsurvK. Ces deux réseaux de neurone basés sur un modèle à temps discret ont été développés par nos soins en utilisant la bibliothèque keras. Nous les avons comparés au réseau de neurones basé sur un modèle de Cox, Cox-nnet et également au modèle de Cox précédé par une étape de sélection effectuée par une méthode de régularisation Lasso (Tibshirani, 1997). Cette comparaison est effectuée à partir d'un plan de simulations présenté en section 3.3 et les résultats provenants des différents réseaux appliqués à ces données simulées sont présentées en section 3.4. Plusieurs comparaisons sur des données réelles sont également réalisées en section 3.5.

3.3 Simulations

Nous avons créé un plan de simulations afin de comparer les différentes approches de réseaux de neurones pour prédire la durée de survie en grande dimension. Le plan de simulation a été divisé en trois parties. La première partie concerne une étude de simulations basée sur Bender et al. (2005) qui propose de générer les données de survie à partir d'un modèle de Cox. Le choix de ce modèle favorise les deux méthodes comparées dans cette étude basées sur le modèle de Cox. C'est pourquoi nous avons choisi d'étendre notre étude à d'autres modélisations. Nous avons dans un premier temps simulé des données de survie à partir d'un modèle AFT en se basant sur Leemis et al. (1990). Considérer des modèles différents pour la simulation de données est intéressant dans le cas de notre étude car les hypothèses associées à ces modèles sont différentes. En effet, le modèle de Cox est un modèle à risque proportionnel ce qui n'est pas le cas pour le modèle AFT. Dans le modèle AFT, les variables vont avoir un effet accélérant ou décélérant sur la survie des individus. Malgré cela, les courbes des fonctions de survie d'un modèle AFT ne se croisent jamais comme pour le modèle de Cox. Cependant, le fait d'avoir des données avec des courbes de survie se croisent permet aux données d'être plus complexes et il est plus difficile pour les méthodes de prédire la survie. De plus, certaines modélisations ne prennent pas en compte le croisement des courbes de survie, ce qui permet de mettre en évidence leurs limites. Pour cet objectif, deux approches ont été considérées. La première approche consiste à modifier le modèle AFT afin d'avoir des courbes de survie qui se croisent. La seconde approche concerne l'utilisation d'un modèle AH (pour *Accelerated Hazards*) (Chen et al., 2000) pour générer les données de survie. Le modèle AH est plus flexible que les deux modèles précédemment cités. Dans le modèle AH, les variables vont accélérer ou décélérer le risque instantané de décès. Les courbes de survie du modèle AH peuvent donc se croiser. La génération des temps de survie est réalisée à partir des différents modèles cités précédemment où on suppose que la fonction du risque de base des modèles est connue et suit une certaine loi de probabilité. Pour ces différents modèles, la génération des temps de survie s'effectue à partir de lois paramétriques. La première loi considérée est la loi de Weibull. Celle-ci a seulement été considérée pour le modèle de Cox. En effet, considérer un modèle AFT avec une loi de Weibull implique que les risques sont proportionnels. Pour le modèle AFT, nous avons donc décidé d'utiliser une loi log-normale. Nous avons également utilisé cette loi pour le modèle AH. J'ai construit un jeu de données simulées dans chacune des modélisations, où j'ai fait varier la taille de l'échantillon, le nombre de variables explicatives total et le nombre de variables explicatives pertinentes considérées dans le modèle. À partir de ces simulations, nous avons pu observer le comportement de chacune des méthodes à partir des deux métriques présentées en section 1.3.

Il est à noter que nous sommes restés dans un cadre de dépendance linéaire et sans interaction. Notre plan de simulation initial prévoyait d'étudier ces différents aspects, mais n'a pu être réalisé par manque de temps. Notons toutefois que le plan proposé s'adaptant facilement au cadre non linéaire et avec interaction en remplaçant $\beta^T X$ par une dépendance $g(X)$ plus complexe.

3.3.1 Génération de données

1 Rappel des fonctions utilisées en analyse de survie

La Table 3.1 résume l'écriture des fonctions utilisées en analyse de survie (risque instantané $\lambda(t)$, fonction des risques cumulés $H_0(t)$, fonction de survie $S(t)$ et densité $f(t)$) pour chacun des modèles considérés (Cox, AFT et AH) pour la génération des données de survie.

2 Génération des temps de survie

Les modèles considérés se décomposent à partir d'une fonction, $\alpha_0(t)$, le risque de base et les paramètres β, traduisant l'effet des variables sur les temps de survie. Pour la génération des données, nous supposons

	Cox	AFT	AH
$H(t\|X_{i.})$	$H_0(t)\exp(\beta^T X_{i.})$	$H_0(t\exp(\beta^T X_{i.}))$	$H_0(t\exp(\beta^T X_{i.}))\exp(-\beta^T X_{i.})$
$\lambda(t\|X_{i.})$	$\alpha_0(t)\exp(\beta^T X_{i.})$	$\alpha_0(t\exp(\beta^T X_{i.}))\exp(\beta^T X_{i.})$	$\alpha_0(t\exp(\beta^T X_{i.}))$
$S(t\|X_{i.})$	$S_0(t)^{\exp(\beta^T X_{i.})}$	$S_0(t\exp(\beta^T X_{i.}))$	$S_0(t\exp(\beta^T X_{i.}))^{\exp(-\beta^T X_{i.})}$
$f(t\|X_{i.})$	$f_0(t)\exp(\beta^T X_{i.})$ $S_0(t)^{\exp(\beta^T X_{i.})}$	$f_0(t\exp(\beta^T X_{i.}))$ $\exp(\beta^T X_{i.})$	$f_0(t\exp(\beta^T X_{i.}))$ $S_0(t\exp(\beta^T X_{i.}))^{(\exp(-\beta^T X_{i.})-1)}$

Table 3.1 – Définition des fonctions utilisées en analyse de survie pour les modèles Cox, AFT et AH

que la fonction du risque de base $\alpha_0(t)$ est connue et suit donc une certaine loi de probabilité (Weibull ou log-normale). Les caractéristiques des deux lois sont résumées dans le Table 3.2, en montrant les correspondances en terme de risque de base et de risque cumulé.

	$Weibull$	$Log\text{-}normale$
Paramètres	$\lambda > 0$ (échelle) $a > 0$ (forme)	$\mu \in]-\infty, +\infty[$ $\sigma > 0$
Espace de définition	$[0, +\infty[$	$]0, +\infty[$
Risque de base	$\alpha_0(t) = \lambda a t^{(a-1)}$	$\alpha_0(t) = \frac{\frac{1}{\sigma\sqrt{2\pi t}}\exp\left[-\frac{(\log t-\mu)^2}{2\sigma^2}\right]}{1-\Phi\left[\frac{\log t-\mu}{\sigma}\right]}$
Risques cumulés	$H_0(t) = \lambda t^a$	$H_0(t) = -\log(1-\Phi\left[\frac{\log t-\mu}{\sigma}\right])$
Inverse des risques cumulés	$H_0^{-1}(u) = \left(\frac{u}{\lambda}\right)^{1/a}$	$H_0^{-1}(u) = \exp(\sigma\Phi^{-1}(1-\exp(-u))+\mu)$
Densité	$f(t) = \lambda a t^{(a-1)}\exp(-\lambda t^a)$	$f(t) = \exp\left[-\frac{(\log t-\mu)^2}{2\sigma^2}\right]\frac{1}{\sigma t\sqrt{2\pi}}$
Fonction de répartition	$F(t) = \exp(-\lambda t^a)$	$F(t) = 1-\Phi\left[\frac{\log t-\mu}{\sigma}\right]$
Espérance	$\mathbb{E}(T) = \Gamma(\frac{1}{a}+1)\frac{1}{\sqrt[a]{\lambda}}$	$\mathbb{E}(T) = \exp(\mu+\frac{\sigma^2}{2})$
Variance	$\mathbb{V}(T) = \left[\Gamma(\frac{2}{a}+1)-\Gamma^2(\frac{1}{a}+1)\right]\frac{1}{\sqrt[a]{\mu^2}}$	$\mathbb{V}(T) = (\exp(\sigma^2)-1)\exp(2\mu+\sigma^2)$

Table 3.2 – Caractéristiques des lois utilisées (Weibull et log-normale) pour la simulation des données de survie. Nous précisons que la paramétrisation utilisée pour la loi de Weibull est celle utilisée par Bender et al. (2005).

La fonction de survie s'écrit à partir du risque instantané :

$$S(t|X) = \exp\left(-\int_0^t \lambda(s|X)ds\right), \tag{3.22}$$

et le risque instantané de chacun des modèles est résumé dans la Table 3.3. Pour les modèles de Cox, AFT et AH, on peut écrire la fonction de survie $S(t|X)$ sous la forme :

$$S(t|X) = \exp(-H_0(\psi_1(X)t)\psi_2(X) \text{ avec} \tag{3.23}$$

$$(\psi_1(X), \psi_2(X)) = \begin{cases} (1, \exp(\beta^T X)) & \text{pour le modèle de Cox} \\ (\exp(\beta^T X), \exp(-\beta^T X)) & \text{pour le modèle AH} \\ (\exp(\beta^T X), 1) & \text{pour le modèle AFT.} \end{cases}$$

	$Cox/Weibull$	$AFT/log\text{-}normale$	$AH/log\text{-}normale$
Fonction de survie $S(t\|X)$	$\exp\left[-H_0(t)\exp(\beta^T X_{i.})\right]$	$\exp\left[-H_0(t\exp(\beta^T X_{i.}))\right]$	$\exp\left[-\frac{H_0(t\exp(\beta^T X_{i.}))}{\exp(\beta^T X_{i.})}\right]$
Durée de survie T	$\mathrm{H}_0^{-1}\left(\frac{-\log(1-U)}{\exp(\beta^T X_{i.})}\right)$	$\mathrm{H}_0^{-1}(-\log(1-U)) \times \exp(-\beta^T X_{i.})$	$\mathrm{H}_0^{-1}\left(\frac{-\log(1-U)}{\exp(-\beta^T X_{i.})}\right) \times \exp(-\beta^T X_{i.})$

Table 3.3 – Expressions de la fonction de survie et des temps de survie pour les modèles de COX, AFT et AH.
U est une variable aléatoire suivant une loi uniforme sur l'intervalle $[0, 1]$.

La fonction de répartition se déduit de la fonction de survie à partir de la formule suivante :

$$F(t|X) = 1 - S(t|X). \tag{3.24}$$

Pour la génération des données, si Y est une variable aléatoire qui suit une loi de probabilité F, alors $U = F(Y)$ suit une loi uniforme sur l'intervalle $[0, 1]$, et $(1 - U)$ suit également une loi uniforme $\mathcal{U}[0, 1]$. À partir de l'équation (3.24), nous obtenons finalement que :

$$1 - U = S(t|X) \tag{3.25}$$
$$= \exp(-H_0(\psi_1(X)t)\psi_2(X)) \tag{3.26}$$

Si $\alpha_0(t)$ est positive pour tout t, alors $H_0(t)$ peut être inversé et le temps de survie de chacun des modèles considérés (Cox, AFT et AH) peut être exprimé à partir de $H_0^{-1}(u)$. L'expression des temps de survie pour chacun des modèles est présentée dans la Table 3.3 et s'écrit de manière générale :

$$T = \frac{1}{\psi_1(X)} H_0^{-1}\left(\frac{\log(1-U)}{\psi_2(X)}\right). \tag{3.27}$$

Deux lois ont été utilisées pour la fonction des risques cumulés $H_0(t)$ afin de générer les données de survie. Si les temps de survie sont distribués selon une loi de Weibull $\mathcal{W}(a, \lambda)$, le risque de base est de la forme :

$$\alpha_0(t) = a\lambda t^{a-1}, \lambda > 0, a > 0. \tag{3.28}$$

La fonction des risques cumulés s'écrit donc :

$$H_0(t) = \lambda t^a, \lambda > 0, a > 0 \tag{3.29}$$

et l'inverse de cette fonction s'exprime de la façon suivante :

$$H_0^{-1}(u) = \left(\frac{u}{\lambda}\right)^{1/a}. \tag{3.30}$$

Dans un second temps, nous avons considéré que les temps de survie suivaient une loi log-normale $\mathcal{LN}(\mu, \sigma)$ de moyenne μ et d'écart-type σ. La fonction du risque de base s'écrit donc :

$$\alpha_0(t) = \frac{\frac{1}{\sigma\sqrt{2\pi t}} \exp\left[-\frac{(\log t - \mu)^2}{2\sigma^2}\right]}{1 - \Phi\left[\frac{\log t - \mu}{\sigma}\right]}, \tag{3.31}$$

avec $\Phi(t)$ la fonction de répartition d'une loi normale centrée et réduite. La fonction des risques cumulés s'écrit :

$$H_0(t) = -\log\left[1 - \Phi\left(\frac{\log t - \mu}{\sigma}\right)\right] \tag{3.32}$$

et donc l'inverse de cette fonction s'exprime par :

$$H_0^{-1}(u) = \exp(\sigma\Phi^{-1}(1 - \exp(-u)) + \mu), \tag{3.33}$$

avec $\Phi^{-1}(t)$ l'inverse de la fonction de répartition d'une loi normale centrée et réduite.

Comme nous l'avons précisé dans la section 3.1, nous avons simulé les données de survie à partir de trois modèles différents. Le premier modèle considéré est le modèle de Cox dont la loi de la fonction de risque base est une loi de Weibull. La loi de Weibull partage l'hypothèse des risques proportionnels avec le modèle de Cox. C'est pour cette raison que nous l'avons associé avec le modèle de Cox et que nous avons choisi une autre loi pour les modèles AFT et AH. Le deuxième modèle considéré pour la simulation de données est le modèle AFT associé à la loi log-normale pour la fonction du risque de base. Avec le modèle AFT, les risques ne sont pas proportionnels. Mais les courbes de survie sont parallèles comme pour le modèle de Cox. Nous voulions étudier une simulation plus complexe avec des courbes de survie se croisant. C'est pourquoi nous avons simulé des données supplémentaires à partir de deux autres scénarios. Tout d'abord, nous avons modifié la simulation à partir d'un modèle AFT/log-normale. Ensuite, nous avons considéré un modèle AH dont la distribution du risque de base est la loi log-normale.

3 Simulation Cox/Weibull

Pour simuler les données à partir du modèle Cox/Weibull, nous nous sommes basés sur Bender et al. (2005). Nous avons choisi de réaliser cette simulation afin de générer des données de survie respectant l'hypothèse des risques proportionnels. Nous rappelons que la génération des données de survie à partir d'un modèle de Cox se fait à partir de :

$$T = H_0^{-1}\left[\frac{-\log(1-U)}{\exp(\beta^T X_{i.})}\right], \tag{3.34}$$

où $U \sim \mathcal{U}[0, 1]$. Pour cette simulation, nous considérons que les temps de survie suivent une loi de Weibull $\mathcal{W}(a, \lambda)$. Dans ce cas, nous avons la fonction des risques cumulés qui s'exprime par :

$$H_0(t) = -\log\left[1 - \Phi\left(\frac{\log t - \mu}{\sigma}\right)\right] \tag{3.35}$$

et les temps de survie peuvent être donc simulés à partir de :

$$T = \frac{1}{\lambda^{1/a}} \left(\frac{-\log(1-U)}{\exp(\beta^T X_{i.})} \right)^{1/a}. \tag{3.36}$$

4 Simulation AFT/Log-normale

Pour simuler les données à partir du modèle AFT/Log-normale, nous nous sommes basés sur Leemis et al. (1990). Nous avons choisi de réaliser cette simulation afin de générer des données de survie ne respectant pas l'hypothèse des risques proportionnels. Nous rappelons que la génération des données de survie à partir d'un modèle AFT se fait à partir de :

$$T = \frac{H_0^{-1}\left[-\log(1-U)\right]}{\exp(\beta^T X_{i.})}, \tag{3.37}$$

où $U \sim \mathcal{U}[0,1]$. Pour cette simulation, nous considérons que les temps de survie suivent une loi log-normale $\mathcal{LN}(\mu, \sigma)$. Dans ce cas, nous avons la fonction des risques cumulés qui s'exprime par :

$$H_0(t) = -\log\left[1 - \Phi\left(\frac{\log t - \mu}{\sigma}\right)\right], \tag{3.38}$$

où $\Phi(t)$ est la fonction de répartition de la loi normale centrée et réduite. Les temps de survie peuvent être donc simulés à partir de :

$$T = \frac{1}{\exp(\beta^T X_{i.})} \exp(\sigma\phi^{-1}(U) + \mu). \tag{3.39}$$

5 Simulation modifiée AFT/Log-normale

Comme nous l'avons évoqué précédemment, nous avons simulé des données de survie à partir d'un modèle AFT modifié afin d'obtenir des courbes de survie qui se croisent. Nous réécrivons la fonction des risques cumulés en y ajoutant un terme $\phi_2(X)$:

$$H_0(t\exp(\beta^T X_{i.}) + \phi_2(X)), \tag{3.40}$$

avec $\phi_2(X) = -\beta_2^T X_{i.}$. À partir de l'équation (3.25), nous obtenons :

$$1 - U = \exp(-H_0(t\exp(\beta^T X_{i.})) - \beta_2^T X_{i.}) \sim \mathcal{U}[0,1] \tag{3.41}$$

car $U \sim \mathcal{U}[0,1]$. Si $\alpha_0(t)$ est positive pour tout t, alors $H_0(t)$ peut être inversé et le temps de survie peut être exprimé à partir de $H_0^{-1}(u)$:

$$T = \frac{1}{\exp(\beta^T X_{i.})} \left[H_0^{-1}\left(-\log(1-U)\right) + \beta_2^T X_{i.}\right]. \tag{3.42}$$

De plus, nous supposons que les temps de survie suivent une loi log-normale et l'inverse de la fonction des risques cumulés s'écrit alors $H_0^{-1}(u) = \exp(\sigma\Phi^{-1}(1-\exp(-u)) + \mu)$ (cf. Table 3.2) avec $\Phi^{-1}(t)$ l'inverse de la fonction de répartition d'une loi normale centrée et réduite. Les temps de survie peuvent donc être générés de la façon suivante :

$$T = \frac{1}{\exp(\beta^T X_{i.})} \left(\exp(\sigma\Phi^{-1}(U) + \mu) + \beta_2^T X_{i.}\right), \tag{3.43}$$

avec $U \sim \mathcal{U}[0,1]$.

6 Simulation AH/Log-normale

En s'inspirant du travail de Bender et al. (2005) et Leemis et al. (1990), nous avons également simulé les données de survie à partir d'un autre modèle, le modèle AH. Nous avons effectué cette simulation afin de générer des données dont les courbes de survie vont se croiser. Nous rappelons que la génération des données de survie à partir d'un modèle AH se fait à partir de :

$$T = \frac{1}{\exp(\beta^T X_{i.})} H_0^{-1}\left[-\frac{\log(1-U)}{\exp(-\beta^T X_{i.})}\right], \tag{3.44}$$

avec $U \sim \mathcal{U}([0,1])$ Pour cette simulation, nous considérons que les temps de survie suivent une loi log-normale $\mathcal{LN}(\mu,\sigma)$. Dans ce cas, nous avons la fonction des risques cumulés qui s'exprime par :

$$H_0(t) = -\log\left[1-\Phi\left(\frac{\log t-\mu}{\sigma}\right)\right], \tag{3.45}$$

où $\Phi(t)$ est la fonction de répartition de la loi normale centrée et réduite. Les temps de survie peuvent être donc simulés à partir de :

$$T = \frac{1}{\exp(\beta^T X_{i.})} \exp\left[\sigma\Phi^{-1}\left(\frac{\log(1-U)}{\exp(-\beta^T X_{i.})}\right)+\mu\right] \tag{3.46}$$

avec $\Phi^{-1}(t)$ l'inverse de la fonction de répartition d'une loi normale centrée et réduite.

3.3.2 Plan de simulations

Nous développons dans cette section le plan de simulations que nous avons réalisé pour évaluer les différentes méthodes de prédiction de la survie étudiées dans le cadre de la grande dimension. Ce plan de simulations peut être divisé en différentes configurations, mais des points communs existent entre elles. Dans chacune des configurations, nous avons considéré deux tailles d'échantillons ($n \in \{200, 1000\}$) et un nombre total de variables prenant plusieurs valeurs ($p_{total} \in \{10, 100, 1000\}$). Je précise maintenant les trois configurations possibles :

1. Dans la première configuration, le nombre de variables pertinentes est égal au nombre total de variables où le nombre de variables total prend les valeurs : $p_{total} \in \{10, 100, 1000\}$;
2. La deuxième configuration permet de prendre en compte l'effet de la censure. Dans cette configuration, nous conservons le même nombre de variables total ($p_{total} \in \{10, 100, 1000\}$) et le nombre de variables pertinentes est également égal au nombre total de variables, mais des temps censurés sont simulés en plus des temps de survie. La démarche utilisée est détaillée en section 4 . La génération des données à partir de cette configuration nous permet d'étudier l'effet de la censure sur les différentes méthodes comparées dans ce manuscrit pour la prédiction de survie en grande dimension.
3. La troisième configuration permet de prendre en compte l'effet de la sparsité. Pour obtenir des données sparses, nous avons changé le nombre de variables pertinentes. Ce nombre de variables pertinentes n'est plus égal au nombre de variables total, il est maintenant fixé à 3 variables ($p = 3$ et $p_{total} \in \{10, 100, 1\,000\}$). La démarche utilisée est détaillée en section 5 . La génération des données à partir de cette configuration va nous permettre d'étudier l'effet de la sparsité.

Pour chacune des simulations présentes dans les différentes configurations, un jeu de données test est simulé dont la taille est égale à 1 000.

1 Indices de référence

Pour comparer les méthodes étudiées à partir du plan de simulation, nous avons utilisé deux métriques différentes : le score de Brier Intégré (IBS) présenté en section 1.3.2 et l'indice de concordance (C_{td} *index*) présenté en section 1.3.1. Nous connaissons de manière exacte le modèle dans le cadre de nos simulations, nous avons donc calculé le C_{td} *index* de référence et l'IBS de référence. L'indice de concordance (C_{td} *Index*) a été introduit dans le chapitre 1 et se calcule lorsque le modèle est connu de la manière suivante :

$$\widehat{C}^{\star}_{td} = \frac{\sum_{i=1}^{n}\sum_{j\neq i}\mathbb{1}_{\{S(t_i|X_{i.})<S(t_j|X_j)\}}\mathbb{1}_{\{(t_i<t_j;\delta_i=1)\cup(t_i=t_j;\delta_i=1,\delta_j=0)\}}}{\sum_{i=1}^{n}\sum_{j\neq i}\mathbb{1}_{\{(t_i<t_j;\delta_i=1)\cup(t_i=t_j;\delta_i=1,\delta_j=0)\}}}, \tag{3.47}$$

$\{(t_1,\delta_1,S(t_{(k)},X_1);k=1,\ldots,K),\ldots,(t_n,\delta_n,S(t_{(k)},X_n);k=1,\ldots,K))\}$ sont respectivement les temps observés, l'indicateur de censure et la fonction de survie connue. Le score de Brier intégré a été introduit dans le chapitre 1 et le score de Brier de référence doit être calculé dans un premier temps :

$$\widehat{BS}^{\star}(t,S) = \frac{1}{n}\sum_{i=1}^{n}\widehat{W}_i(t)(\widetilde{Y}_i(t)-S(t|X_{i.}))^2, \tag{3.48}$$

avec $\widetilde{Y}_i(t)$ le statut observé de l'individu i, $S(t|X_{i.})$ la probabilité de survie connue au temps t pour l'individu i et n le nombre d'individus dans l'ensemble de test. Le score de Brier intégré de référence se calcule donc :

$$\widehat{IBS}^{\star} = \frac{1}{\tau}\int_0^{\tau}\widehat{BS}^{\star}(t,S)dt, \tag{3.49}$$

où $\widehat{BS}^{\star}(t,S)$ est le score de Brier de référence estimé et $\tau>0$. On fixe $\tau>0$ qui peut être par exemple le maximum des temps observés et le score de Brier est moyenné sur l'intervalle $[0,\tau[$.

2 Répartition des données

J'évoque dans ce paragraphe la distribution des données de survie. Nous avons travaillé sur la distribution des temps de survie afin qu'elle soit réaliste, c'est-à-dire proche de données réelles. Nous nous sommes basés sur la distribution des temps de survie des données du cancer du sein. Ces données sont présentées de manière plus approfondie en section 3.5 et sont accessibles à l'adresse : `www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE6532`. La moyenne, la variance et la médiane des temps de survie de ces données sont respectivement de 2325, 1619996 et 2280. L'histogramme des temps de survie est présenté sur la FIGURE 3.4. Pour chacune des simulations des configurations présentées ci-dessous, nous avons donc joué sur les paramètres de la loi utilisée afin d'obtenir une distribution des temps de survie proche de celle des données réelles. Mais la réalisation d'une distribution proche pour certaines simulations était difficile dû à l'augmentation du nombre de variables, au souhait d'avoir une valeur du C-index C_{td} de référence élevée et au souhait d'avoir des données avec des courbes de survie se croisant.

3 Configuration 1

Nous détaillons la génération des données dans la configuration 1 pour les modèles présentés en section 1.2.

Simulation Cox/Weibull

— Taille d'échantillon n et nombre total de variables p_{total} : Plusieurs choix de n ont été condidérés. La taille d'échantillon prend les valeurs $n=200$ et $n=1000$ et le nombre de variables varie $p_{total}=10$, $p_{total}=100$ et $p_{total}=1000$.

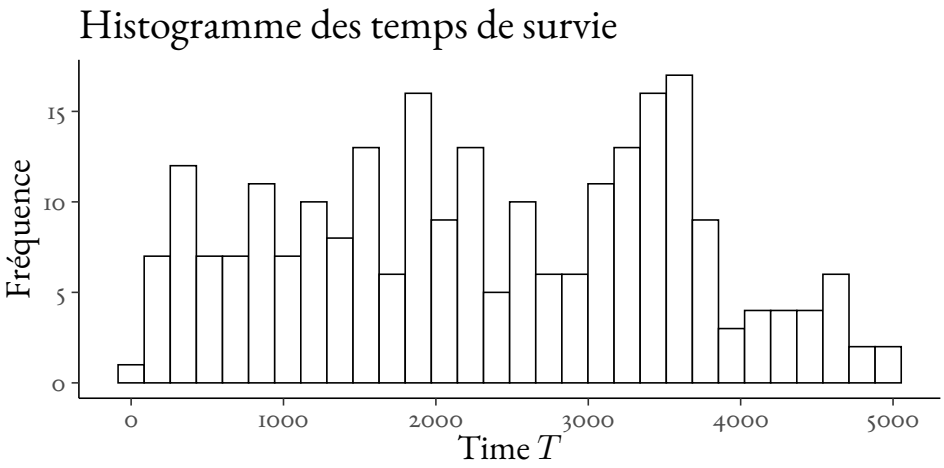


Figure 3.4 – Distribution des temps de survie des données réelles du cancer du sein

— Matrice X : La matrice $X = (X_{i,j})_{1 \leq i \leq n, 1 \leq j \leq p}$ est simulée à partir d'une loi normale centrée et réduite $\mathcal{N}(0, 1)$.

— Paramètre de régression β : Le paramètre de régression β est un vecteur de taille p_{total} dont les coefficients sont égaux à 1 : $\beta = (1, 1, 1, \ldots, 1)^T \in \mathbb{R}^{p_{total}}$.

— Normalisation : Nous avons pu remarquer que l'augmentation du nombre de variables produisait une augmentation de la valeur de la moyenne et de la variance des temps de survie simulés. Pour corriger cette augmentation, un terme de normalisation en $1/\sqrt{p}$ doit être ajouté dans l'équation (3.36) :

$$T = \left(-\frac{1}{\lambda} \log(1 - U) \exp(-\frac{1}{\sqrt{p}} \beta^T X) \right)^{\frac{1}{a}}. \tag{3.50}$$

— Paramètres a et λ de la loi Weibull : Le choix des paramètres a et λ s'est fait pour que la distribution des temps de survie soit similaire à la distribution des temps de survie des données réelles. Afin d'avoir une espérance et une médiane valant respectivement 2325 et 2280, les valeurs de a et λ sont respectivement 2.67 et $7.5e - 10$. Pour calculer ces valeurs, les formules de l'espérance et de la variance résumées dans la Table 3.2 ont été utilisées. Tout d'abord, nous avons cherché a tel que $m = 2280$, où m est la médiane des temps de survie :

$$m = \frac{\mathbb{E}(T)}{\Gamma(1 + 1/a)} \log(2)^{\frac{1}{a}},$$

avec $\mathbb{E}(T) = 2325$. Une fois a estimé, il suffit d'injecter sa valeur dans la formule suivante pour calculer λ :

$$\lambda = \left[\frac{1}{\mathbb{E}(T)} \Gamma\left(1 + \frac{1}{a}\right) \right]^a.$$

Sur la Figure 3.5, nous pouvons voir la distribution diffère légèrement. Le maximum des temps de survie (12000) est différent par rapport à celui des temps de survie des données réelles (5000). Mais cela était difficile à gérer avec notre plan de simulations et cette distribution était le meilleur compromis entre garder les mêmes valeurs de moyennes et de variances et des temps de survie qui pouvaient avoir une signification. De plus, nous pouvons voir que la normalisation a permis d'avoir une distribution assez similaire selon le nombre de variables considérées.

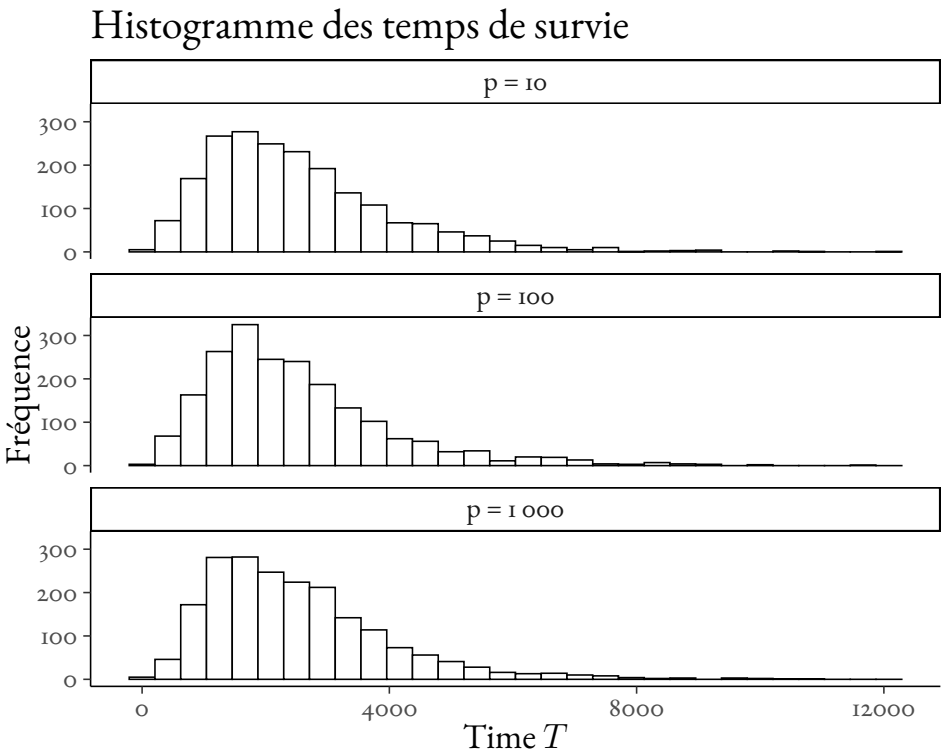


Figure 3.5 – Distribution des temps de survie simulés par un modèle Cox/Weibull

Simulation AFT/Log-normale

- Taille d'échantillon n et nombre de variables total p_{total} : $n = 200$ et $n = 1000$; $p_{total} = 10$, $p_{total} = 100$ et $p_{total} = 1000$.
- Matrice X : $X = (X_{i,j})_{1 \leq i \leq n, 1 \leq j \leq p}$ est simulée à partir d'une loi uniforme sur $[-1, 1]$.
- Paramètre de régression β : β est un vecteur de taille p_{total} dont les coefficients sont égaux à 1 : $\beta = (1, 1, 1, \ldots, 1)^T \in \mathbb{R}^{p_{total}}$.
- Normalisation : Comme pour la simulation Cox/Weibull présentée en section 3 , l'augmentation du nombre de variables produit une augmentation des valeurs de l'espérance et de la variance des temps de survie simulés. Nous avons également ajouté un terme de normalisation en $1/\sqrt{p}$ dans l'équation (3.39) :

$$T = \frac{1}{\exp((1/\sqrt{p})\beta^T X)} \left(\exp(\sigma \Phi^{-1}(U) + \mu)\right). \tag{3.51}$$

- Paramètres μ et σ de la loi log-normale : Nous avons adopté la même démarche pour choisir les paramètres σ et μ de la distribution des temps de survie que celle utilisée pour la simulation Cox/Weibull présentée en section 3 . Nous voulons que cette distribution soit proche de la distribution des temps de survie des données réelles. L'obtention de la valeur des paramètres se fait à partir des formules explicites (cf. Table 3.2) :

$$\mu = \ln(\mathrm{E}(T)) - \frac{1}{2}\sigma^2 \tag{3.52}$$

$$\sigma^2 = \ln\left(1 + \frac{\mathrm{Var}(T)}{(\mathrm{E}(T))^2}\right). \tag{3.53}$$

Le calcul de σ^2 se fait dans un premier temps à partir des valeurs que l'on souhaite pour l'espérance et la variance des temps de survie. Les valeurs souhaitées sont respectivement 2325 et 170000 pour

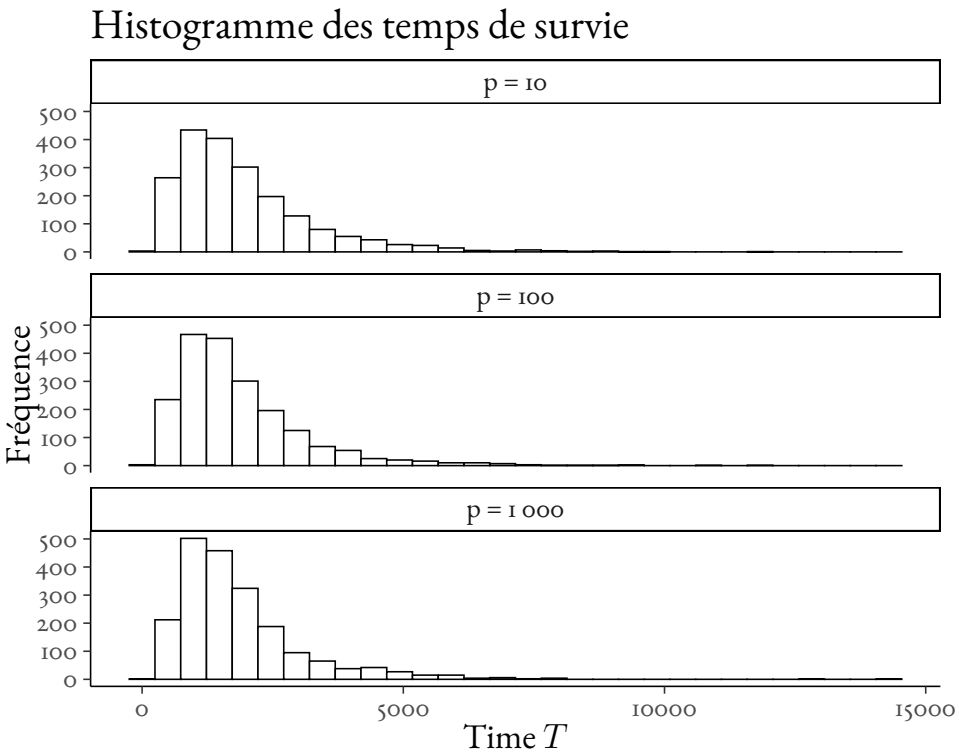


Figure 3.6 – Distribution des temps de survie simulés par un modèle AFT/Log-normale

l'espérance et la variance. Les valeurs de μ et σ utilisées pour la simulation des données de survie sont $\mu = 7.73$ et $\sigma = 0.1760$. Sur la Figure 3.6, nous pouvons voir que la distribution diffère légèrement des données réelles (cf. Figure 3.4) mais est assez proche de celle de la simulation Cox/Weibull (cf. Figure 3.5). Le maximum des temps de survie (15000) est différent par rapport à celui des temps de survie des données réelles (5000). Comme pour la simulation Cox/Weibull, nous avons essayé d'avoir le meilleur compromis. Mais le maximum est plus grand que celui de la simulation Cox/Weibull. Enfin, nous pouvons voir que la normalisation a permis d'avoir une distribution assez similaire selon le nombre de variables considérées.

Simulation AH/Log-normale

- Taille d'échantillon n et nombre de variables total p_{total} : $n = 200$ et $n = 1000$; $p_{total} = 10$, $p_{total} = 100$ et $p_{total} = 1000$.
- Matrice X : $X = (X_{i,j})_{1 \leq i \leq n, 1 \leq j \leq p}$ est simulée à partir d'une loi uniforme sur $[-1, 1]$.
- Paramètre de régression β : β est un vecteur de taille p_{total} dont les coefficients sont égaux à 1.5 : $\beta = (1.5, 1.5, 1.5, \ldots, 1.5)^T \in \mathbb{R}^{p_{total}}$.
- Normalisation : Comme pour la simulation Cox/Weibull présentée en section 3 et la simulation AFT/log-normale présentée en section 3 , l'augmentation du nombre de variables produit une augmentation des valeurs de l'espérance et de la variance des temps de survie simulés. Nous avons également ajouté un terme de normalisation en $1/\sqrt{p}$ dans l'équation (3.46) :

$$T = \frac{1}{\exp((1/\sqrt{p})\beta^T X)} \exp\left[\sigma \Phi^{-1}\left(\frac{\log(1-U)}{\exp(-(1/\sqrt{p})\beta^T X)}\right) + \mu\right]. \tag{3.54}$$

- Paramètres μ et σ de la loi log-normale : Nous utilisons pour la simulation AH/log-normale la même loi que la simulation AFT/log-normale afin de générer les temps de survie. La démarche pour obtenir

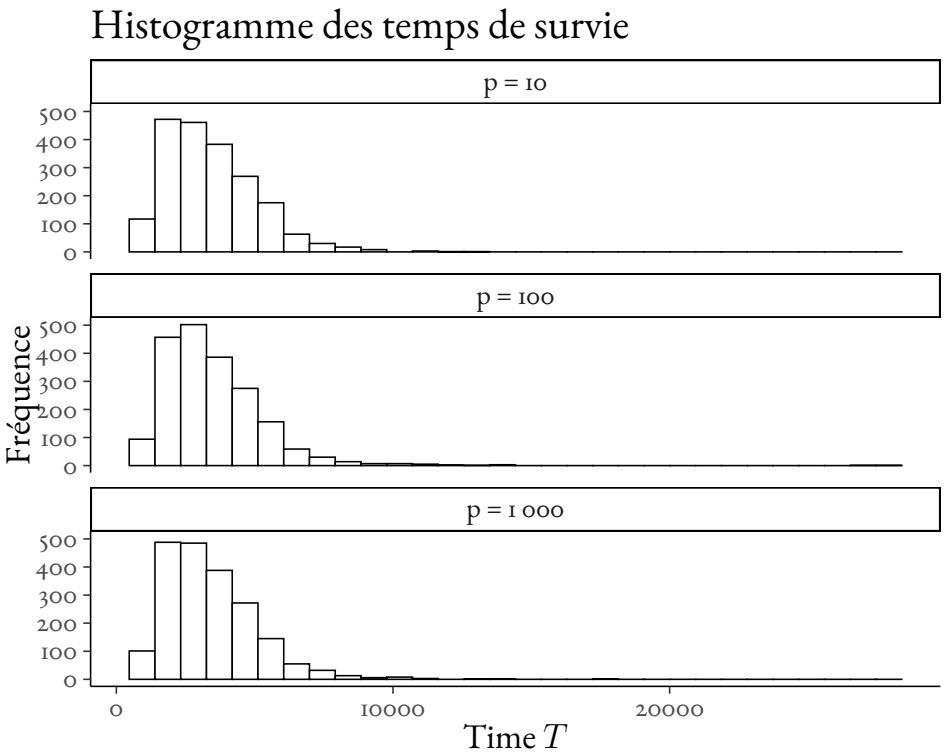


Figure 3.7 – Distribution des temps de survie simulés par un modèle AH/Log-normale

les paramètres σ et μ est donc la même que celle présentée dans les paragraphes précédents. Cependant, si nous gardions les valeurs obtenues pour μ et σ les courbes de survie se croisaient un peu tard. Nous avons donc multiplié par 4 la valeur de σ. Les valeurs de μ et σ utilisées pour la simulation des données de survie sont $\mu = 7.73$ et $\sigma = 0.7$. Sur la Figure 3.7, nous pouvons voir que la distribution est proche de la simulation AFT/Log-normale. Mais nous pouvons observer sur la Figure 3.7 que les temps de survie sont plus grands. Nous avons modifié les valeurs des paramètres de la loi log-normale afin de gérer les courbes de survie ainsi que le C_{td} de référence en essayant d'avoir un maximum des temps de survie le plus petit possible. Comme pour les simulations précédentes, nous avons joué sur les paramètres μ et σ afin d'avoir le meilleur compromis entre la distribution des temps de survie, le C_{td} de référence et le croisement des courbes de survie.

Simulation modifiée AFT/log-normale

- Taille d'échantillon n et nombre de variables total p_{total} : $n = 200$ et $n = 1000$; $p_{total} = 10$, $p_{total} = 100$ et $p_{total} = 1000$.
- Matrice X : $X = (X_{i,j})_{1 \leq i \leq n, 1 \leq j \leq p}$ est simulée à partir d'une loi uniforme sur $[-1, 1]$.
- Paramètre de régression β : β est un vecteur de taille p_{total} dont les coefficients sont égaux à 1 : $\beta = (1, 1, 1, \ldots, 1)^T \in \mathbb{R}^{p_{total}}$.
- Paramètre β_2 : β_2 est un vecteur de taille p_{total} qui suit une loi uniforme entre $[-1.5, 1.5]$.
- Normalisation : Comme pour la simulation Cox/Weibull et la simulation AFT/log-normale, l'augmentation du nombre de variables produit une augmentation des valeurs de l'espérance et de la variance des temps de survie simulés. Nous avons également ajouté un terme de normalisation en $1/\sqrt{p}$ dans l'équation (3.39) :

$$T = \frac{1}{\exp((1/\sqrt{p})\beta^T X)} \left(\exp(\sigma \Phi^{-1}(U) + \mu) + \beta_2^T X\right). \tag{3.55}$$

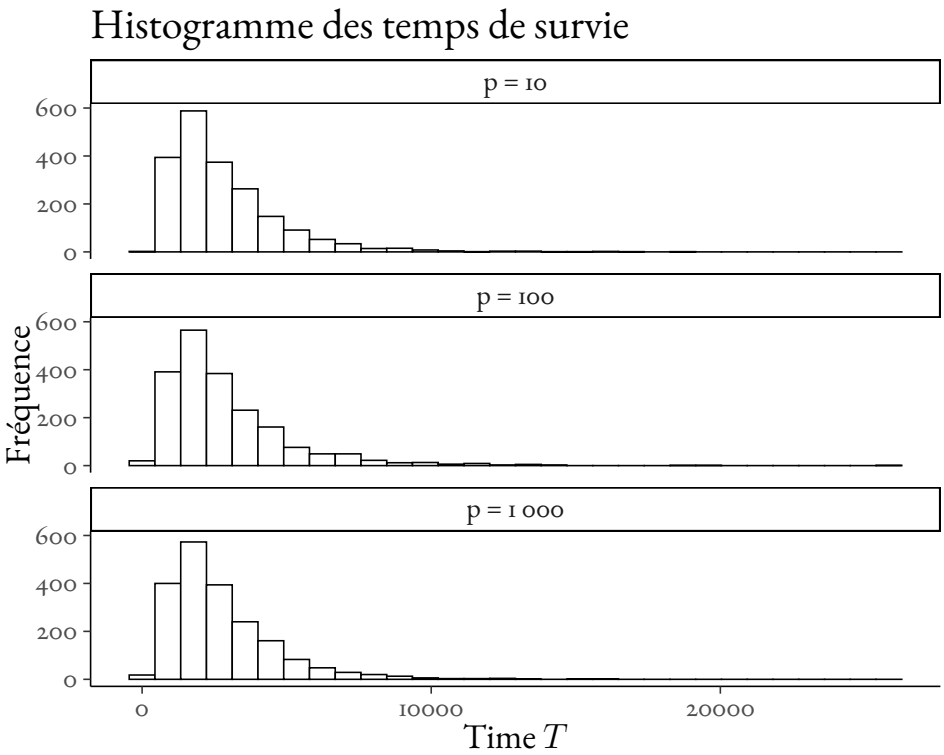


Figure 3.8 – Distribution des temps de survie simulés par un modèle AFT/Log-normale avec un ajout d'un terme $\phi_2(X)$

— Paramètres μ et σ de la loi log-normale : Nous utilisons pour la simulation modifiée AFT/log-normale la même loi que la simulation AFT/log-normale pour générer les temps de survie. La démarche pour obtenir les paramètres σ et μ est donc la même que celle présentée dans les paragraphes précédents. Cependant, si nous gardions les valeurs obtenues pour μ et σ les courbes de survie se croisaient un peu tard. Nous avons donc joué sur la valeur de σ afin d'obtenir des courbes de survie qui se croisent plus rapidement. Les valeurs de μ et σ utilisées pour la simulation des données de survie sont $\mu = 7.73$ et $\sigma = 0.3$. Sur la Figure 3.8, nous pouvons voir que la distribution est proche de la simulation AFT/Log-normale. Mais nous pouvons observer sur la Figure 3.8 que les temps de survie sont plus grands. Le maximum est supérieur à 20000. Il a été difficile de gérer le maximum des temps de survie avec l'ajout du terme ϕ_2 tout en gardant une valeur pour le C_{td} de référence la plus proche de 1.

4 Configuration 2 : effet de la censure dans le modèle modifié AFT/Log-normale

Nous détaillons dans cette section la deuxième configuration correspondant à des données de survie simulées dont le pourcentage de censure est fixé afin de quantifier l'effet de celle-ci sur les performances des méthodes pour la prédiction de survie étudiée dans ce manuscrit.

Simulation modifiée AFT/Log-normale

Le plan de cette simulation reste le même que celui présenté dans la section 3 , mais nous considérons en plus la censure. Nous avons décidé d'utiliser le modèle modifié AFT/Log-normale pour quantifier l'effet de la censure car c'est un des modèles qui permet d'avoir des courbes de survie qui se croisent. Cela permet donc d'avoir des données de survies plus complexes. Pour simuler les temps censurés, nous avons mis un taux de censure de la forme $1/\gamma\mathbb{E}[T]$, où $\gamma > 0$ est une constante ajustée au taux de censure. Nous considérons

un grand taux de censure autour de 50% en prenant $\gamma = 1.1$. Les temps censurés $(C_i)_{1\leq i\leq n}$ sont simulés indépendemment des temps de survie *via* une loi exponentielle $\mathcal{E}(1/\gamma\mathbb{E}[T])$ de paramètre $1/\gamma\mathbb{E}[T]$.

5 Configuration 3 : effet de la sparsité dans le modèle modifié AFT/Log-normale

Nous détaillons dans cette section la troisième configuration correspondant à des données de survie simulées dont peu de variables ont un effet sur la survie des individus. Cette simulation a été créée afin de mesurer l'effet de la sparsité sur les performances des méthodes comparées dans cette thèse pour la prédiction de survie en grande dimension.

Simulation modifiée AFT/Log-normale

La taille de l'échantillon et le nombre de variables totales restent identiques aux simulations précédentes, mais le nombre de variables pertinentes va être fixé à 3.

- Taille des échantillons et nombre de variables : La taille des échantillons prend les valeurs 200 et 1 000 et le nombre total de variables est 10, 100 et 1 000. Le nombre de variables pertinentes (non nulles) est fixé à 3 ($p = 3$ 3 variables pertinentes sur $\{10, 100, 1\,000\}$ variables au total).
- Matrice X : $X = (X_{i,j})_{1\leq i\leq n, 1\leq j\leq p}$ est simulée à partir d'une loi uniforme sur $[-1, 1]$.
- Paramètre de régression β : β est un vecteur de dimension p_{total} défini de la façon suivante : $\beta = (0.5, 0.7, 0.9, 0, \ldots, 0)^T \in \mathbb{R}^{p_{total}}$.
- Paramètre β_2 : β_2 est un vecteur de taille p_{total} qui suit une loi uniforme entre $[-1.5, 1.5]$.
- Normalisation : Comme pour la simulation Cox/Weibull présentée en section 3 et la simulation AFT/log-normale présentée en section 3 , l'augmentation du nombre de variables produit une augmentation des valeurs de l'espérance et de la variance des temps de survie simulés. Nous avons également ajouté un terme de normalisation en $1/\sqrt{p}$ dans l'équation (3.39) :

$$T = \frac{1}{\exp((1/\sqrt{p})\beta^T X)} \left(\exp(\sigma\Phi^{-1}(U) + \mu) + \beta_2^T X\right). \tag{3.56}$$

- Paramètres μ et σ de la loi log-normale : Nous utilisons pour la simulation modifiée AFT/log-normale prenant en compte la sparsité la même loi que celle de la simulation modifiée AFT/log-normale pour générer les temps de survie. La démarche pour obtenir les paramètres σ et μ est donc la même que celle présentée en section 3 pour la première configuration. Les valeurs de μ et σ utilisées pour la simulation des données de survie sont $\mu = 7.73$ et $\sigma = 0.3$.

L'histogramme des distributions des configurations 2 et 3 est présenté en Annexe F. Mais celui-ci diffère très peu de la simulation AFT/Log-normale (cf. Figure 3.6).

3.3.3 Comportement des données simulées

1 Vérification de l'hypothèse de proportionnalité

L'hypothèse des risques proportionnels signifie que les fonctions de risques pour deux individus sont proportionnels et que leur rapport est indépendant du temps. Pour vérifier cette hypothèse, il existe plusieurs méthodes soit graphiques, soit basées sur des tests statistiques. Mais ces méthodes ont toutes des limites (Grambsch et al., 1994). Dans ce manuscrit, nous introduisons tout d'abord les résidus de Schoenfeld et nous présentons ensuite le test de proportionnalité proposé par Grambsch et al. (1994).

Résidus de Schoenfeld

L'idée des résidus de Schoenfeld (Schoenfeld, 1980; Schoenfeld, 1982) est de calculer pour chaque temps de décès t_i la différence entre les variables de l'individu décédé et une moyenne pondérée des variables

des individus à risque au temps t_i :

$$\hat{r}_{ij} = X_{ij} - \bar{X}_{ij},$$

avec $\hat{r}_{ij}$ les résidus au temps t_i pour la variable j, X_{ij} la valeur de la variable j pour l'individu décédé au temps t_i et $\bar{X}_{ij}$ la moyenne pondérée de la variable j pour les individus à risque au temps t_i. Ces résidus sont standardisés par leur variance $\hat{r}^{\star}_{ij} = [\mathbb{V}(\hat{r}_{ij})]^{-1}\,\hat{r}_{ij}$.

Pour vérifier l'hypothèse de proportionnalité à l'aide de ces résidus standardisés, le principe est de les représenter graphiquement en fonction du temps (ou d'une transformation du temps). Si les résidus sont distribués de la même manière au cours du temps alors l'hypothèse des risques proportionnels est vérifiée. Une courbe représentant l'évolution moyenne des résidus au cours du temps peut également être ajoutée sur le graphe. On peut alors observer si une différence existe entre cette courbe et la droite $y = 0$. Si c'est le cas, cela signifie que le risque n'est pas constant au cours du temps et l'hypothèse des risques proportionnels n'est pas vérifiée.

À partir de ces résidus, on peut également effectuer un test. Ce test permet de tester la corrélation entre les résidus et le temps (ou la transformation du temps). De nombreux travaux ont été réalisés pour le test des risques proportionnels qui se différencient par le choix de la fonction de transformation du temps dans les différents tests (Wei, 1984; Gill et al., 1987; O'Quigley et al., 1989). Le choix de la transformation du temps peut amener à une mauvaise spécification de modèle. Grambsch et al. (1994) a proposé afin de tenir compte du problème de la transformation du temps, un test global et un test pour chaque variable des résidus standardisés de Schoenfeld.

Test des risques proportionnels

Le modèle à risques proportionnels de Cox est défini à partir du risque instantané donné par :

$$\lambda(t) = \alpha_0(t)\exp(\beta^T X_i), \tag{3.57}$$

où $\alpha_0(t)$ est le risque de base et β est le vecteur des coefficients de régression. Ce modèle est dit à risques proportionnels car $\beta(t)$ est constant ($\beta = \beta(t)$) et donc le rapport des risques instantanés entre deux individus ne dépend pas du temps. Pour tester que les risques proportionels sont vérifiés, il est nécessaire de montrer que $\beta(t)$ est constant. Dans ce but, Grambsch et al. (1994) a considéré $\beta(t) = \beta + \theta g(t)$ et a proposé à la fois une classe de tests pour une fonction g déterministe ou aléatoire, ainsi qu'un estimateur du paramètre θ. Le test des risques proportionnels proposé par Grambsch et al. (1994) consiste donc à tester l'hypothèse nulle des risques proportionnels comme dans (3.57) *versus* l'alternative des coefficients dépendants du temps avec le risque instantané donné par :

$$\lambda_i(t) = \alpha_0(t)\exp(\{\beta + g(t)\theta\}^T X_i). \tag{3.58}$$

Le test utilisé est un test du χ^2 à p degrés de liberté avec pour hypothèse nulle $H_0 : \theta = 0$ et la statistique du test est définie de la manière suivante :

$$T(G) = (\sum G_i ri)^T (\sum G_i V_i G_i)^{-1} (\sum G_i r_i), \tag{3.59}$$

avec $G_i = G(t_i)$ est une matrice avec les coefficients $g(t_i) - \bar{g}$, r_i les résidus standardisés de Schoenfeld et V_i la variance des résidus. Les détails de ce test sont précisés en Annexe E.

De plus, nous avons été confrontés à une autre problématique que celle évoquée dans la littérature et détaillée en Annexe E en utilisant ce test pour vérifier la proportionnalité de nos données simulées. Pour la simulation réalisée à partir d'un modèle de Cox avec un risque de base qui suit un loi de Weibull, nous avons remarqué que quand le nombre de variables était supérieur à un seuil le résultat du test correspondait au rejet des risques proportionnels alors que ces données devraient les vérifier. Nous avons pu observer que ce seuil était

environ égal à 100 variables. Nous avons donc décidé d'utiliser la méthode de Grambsch et al. (1994) avec une régularisation Lasso afin de faire une sélection de variables. Cette sélection de variables a été initialement réalisée afin de voir si les variables sélectionnées respectaient l'hypothèse des risques proportionnels. Quand le nombre de variables sélectionnées est inférieur à 100, le résultat du test est l'acceptation de l'hypothèse des risques proportionnels. La démarche est précisée dans le paragraphe suivant.

Simulation Cox/Weibull

L'hypothèse des risques non-proportionnels est vérifiée pour la simulation Cox/Weibbull pour les deux premières configurations (p=10, p=100), mais l'hypothèse des risques proportionnels n'est plus vérifiée pour la dernière configurations (p = 1000) comme nous pouvons le voir sur la Figure 3.9. Nous nous sommes intéressés plus attentivement au résultat du test des risques proportionnels de cette configuration, nous pouvons voir que la première variable ne respecte pas les risques proportionnels en considérant sa p-valeur. Cependant, lorsque nous observons les résidus standardisés tracés, ils semblent être répartis de manière homogène.

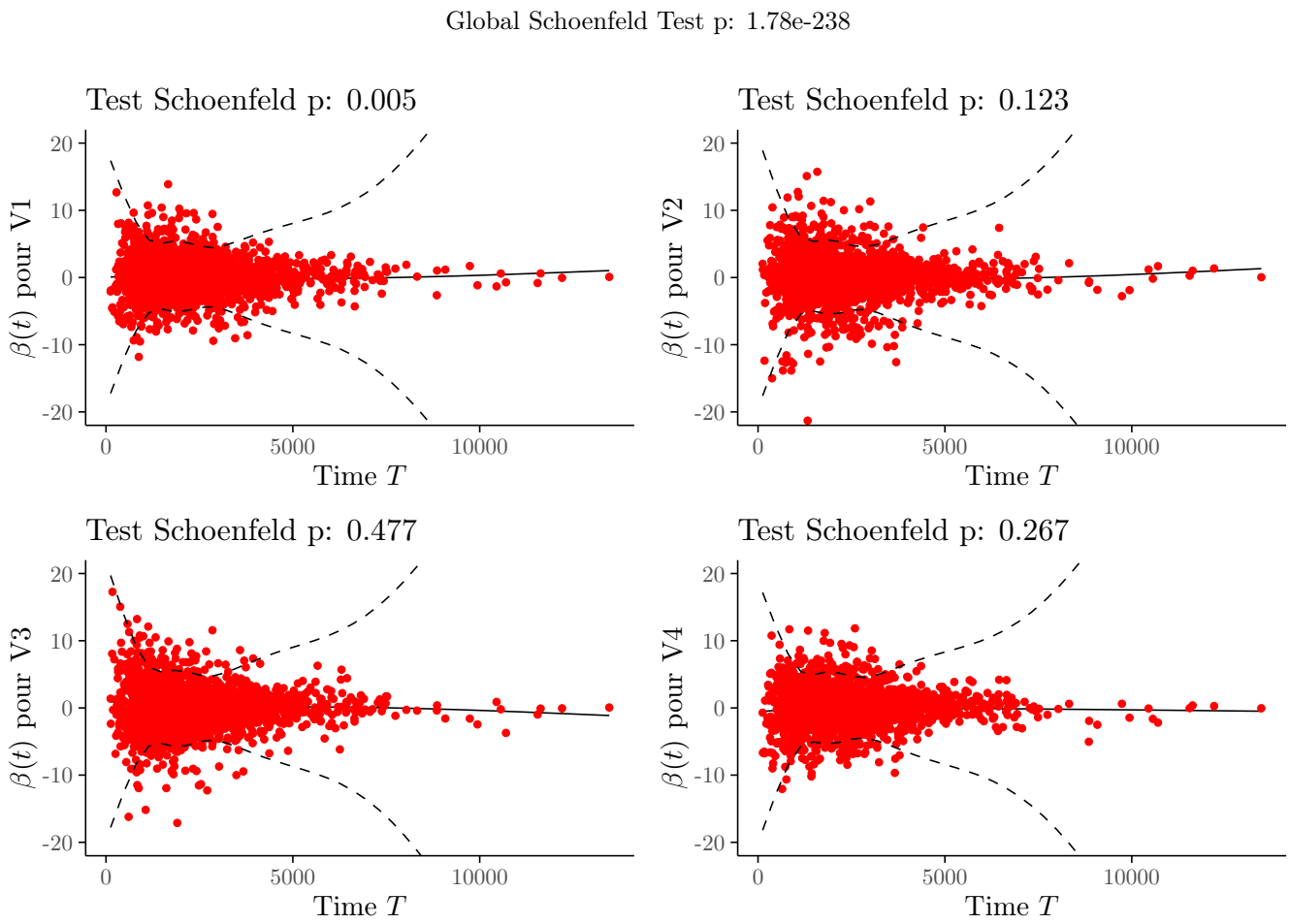


Figure 3.9 – Résidus standardisés pour 1000 variables

Nous avons effectué une procédure Lasso et Adaptive Lasso afin de voir si le résultat du test était différent dans les différents cas :

— Quand moins de 100 variables sont sélectionnées par la procédure Lasso, le résultat du test des risques proportionnels donne une p-valeur supérieure à 0.05. Cela signifie que l'hypothèse des risques proportionnels peut être considérée comme vérifiée.

	p-valeur du test global	Risques proportionnels vérifiés
Cox/weibull	0.35	OUI
AFT/Log-normale	8.9×10^{-09}	NON
AH/Log-normale	5.9×10^{-24}	NON
AFT/Log-normale modifié	5.4×10^{-12}	NON
AFT/Log-normale sparse	9.1×10^{-06}	NON
AFT/Log-normale censuré	2.9×10^{-10}	NON

Table 3.4 – Résultats du test global de Grambsch et al. (1994) sur les données simulées avec 100 variables

— Nous avons par la suite réalisé une procédure Adaptive-Lasso (car la procédure Lasso est instable en sélection) et une procédure modifiée de l'Adaptive-Lasso (la pénalisation des poids est obtenue à partir d'une procédure Ridge). Les résultats nous montrent comme précédemment que si le nombre de variables est supérieur à 100 alors le test conclura à rejeter l'hypothèse nulle et donc les risques peuvent être considérés non-proportionnels.

Les tableaux du test des risques proportionnels sont présentés en Annexe E.

Enfin, nous concluons bien que l'hypothèse des risques proportionnels n'est pas vérifiée pour les données issues des simulations basées sur un modèle AFT ou AH avec un risque de base qui suit une loi log-normale. Les résultats du test global sur chaque simulation sont présentés dans la Table 3.4 pour le cas où on a 100 variables. Les résultats pour les autres cas ($p = 10$ et $p = 1000$) sont présentés dans les Tables E.1 et E.2 en Annexe E.

2 Courbes de survie

Dans cette section, nous étudions le comportement des courbes de survie dans le cadre de la configuration 1 de la simulation des données. Dans cette configuration, les données de survie sont simulées à partir de différents modèles et les courbes de survie peuvent se croiser dans certains modèles. Les courbes de survie des configurations 2 et 3 sont présentées en Annexe F.

Simulation Cox/Weibull Pour cette simulation, le modèle utilisé est le modèle de Cox introduit en section 1.2.3. C'est un modèle à risques proportionnels, ce qui signifie que le rapport des risques ne dépend pas du temps. Les courbes des risques instantanés sont donc parallèles entre les individus, elles ne se croisent jamais. Les courbes de survie sont tracées sur la Figure 3.10.

Simulation AFT/Log-normale Pour cette deuxième simulation, le modèle utilisé est le modèle AFT présenté en section 1.2.4. L'effet des variables agit de manière multiplicative sur l'échelle du temps pour la fonction de survie dans ce modèle. Autrement dit, les courbes de survie de deux individus auront la même forme, mais une des courbes aura du retard ou de l'avance sur le temps de survie. En revanche, les courbes des risques instantanés peuvent se croiser. La Figure 3.11 montre que les courbes de survie ne se croisent pas dans cette simulation. Pour illustrer cela, nous avons pris 5 individus.

Simulation modifiée AFT/Log-normale Nous avons modifié la simulation précédente afin d'avoir des courbes de survie qui se croisent dans le modèle AFT. La démarche utilisée pour cela est présentée en section 3.3.1. Les courbes de survie obtenues pour cette simulation ont été tracées pour 10 individus sur le graphe de la Figure 3.12. Nous pouvons observer que certaines des courbes de survie des individus se croisent.

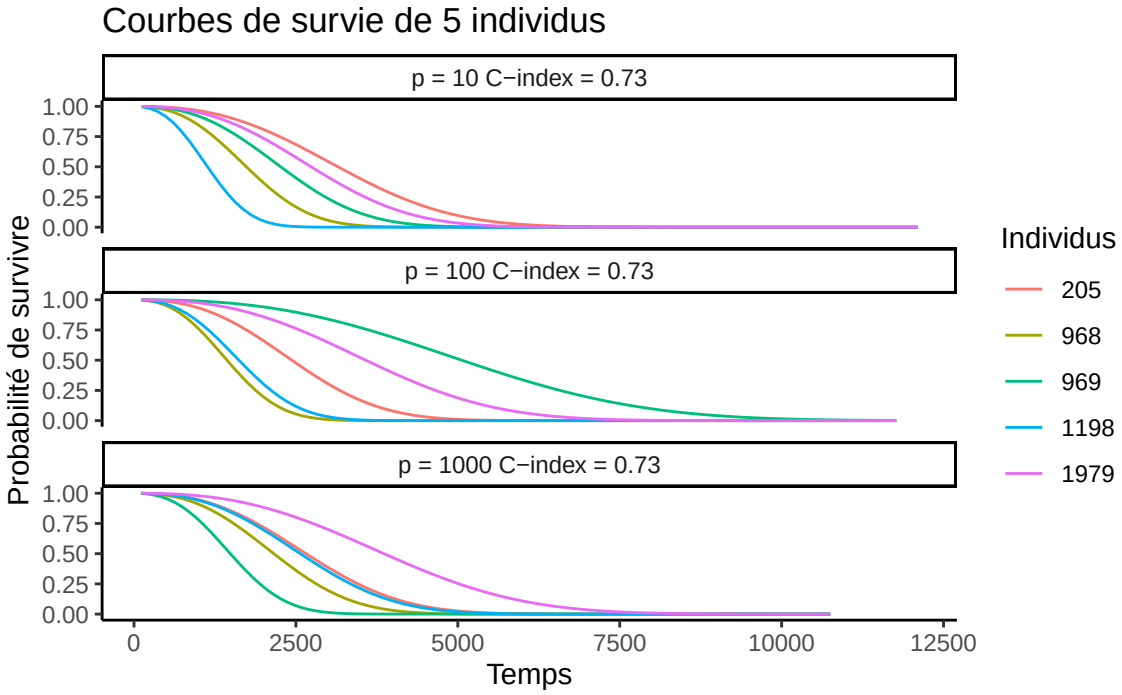


Figure 3.10 – Courbes de survie de différents individus pour la simulation Cox/Weibull

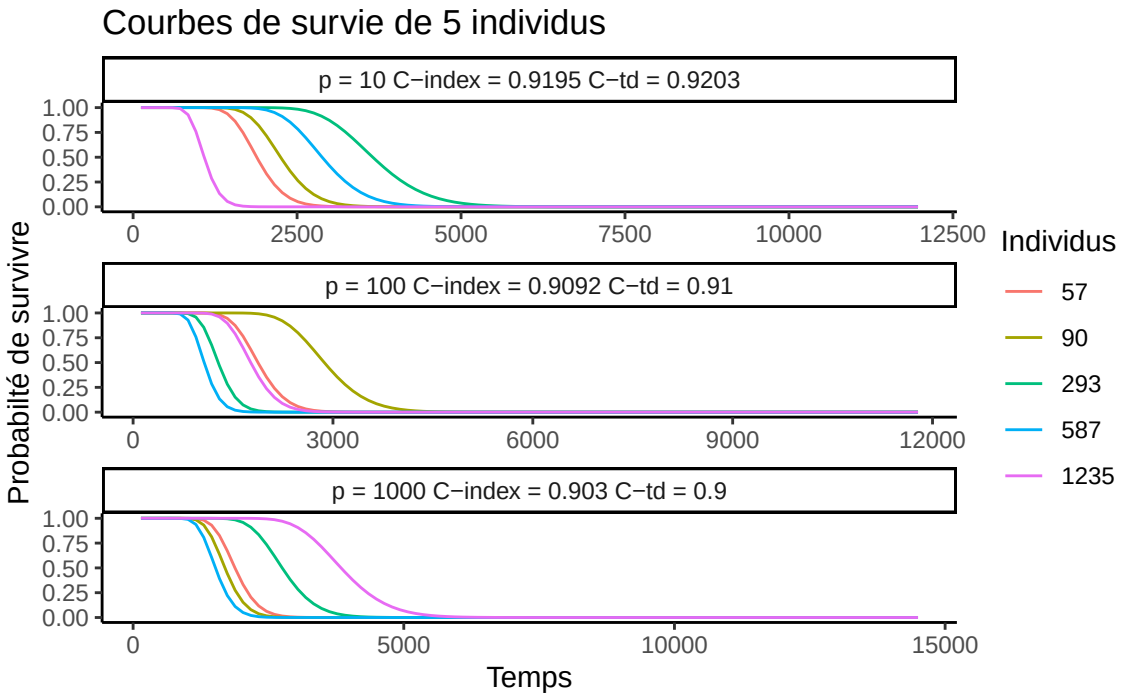


Figure 3.11 – Courbes de survie de différents individus pour la simulation AFT/Log-normale

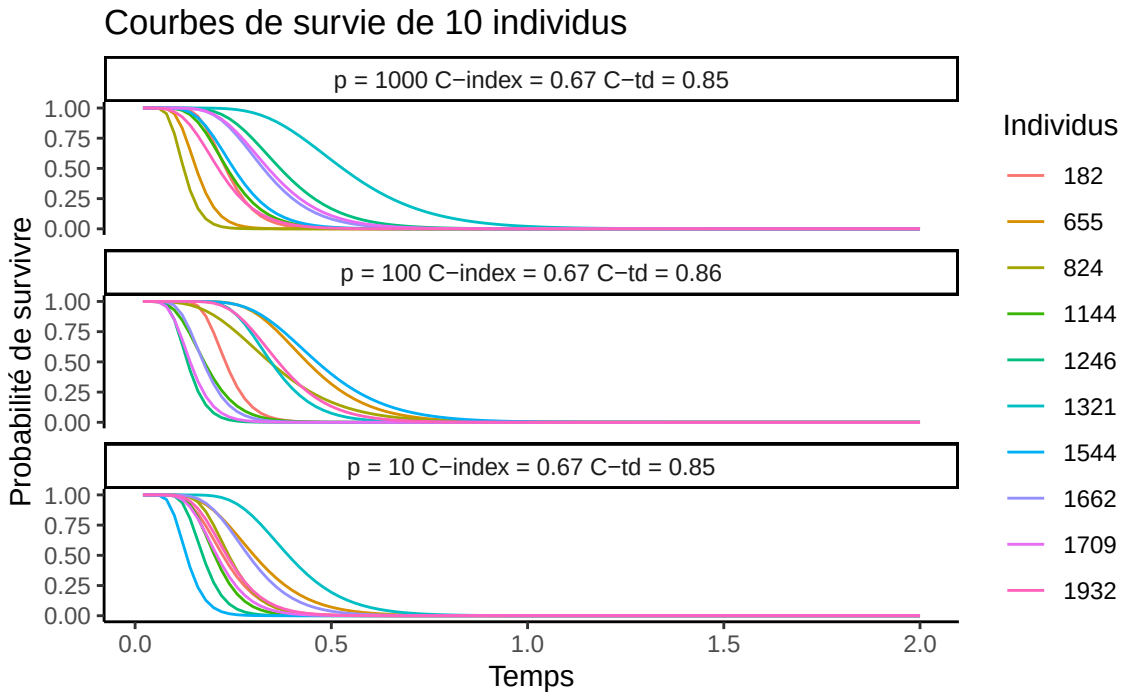


Figure 3.12 – Courbes de survie de différents individus pour la simulation AFT/Log-normale modifiée

Simulation AH/Log-normale Nous avons considéré pour notre dernière simulation le modèle AH. Dans ce modèle, contrairement aux deux modèles précédents, les courbes de survie et des risques instantanés peuvent se croiser. La Figure 3.13 représente les courbes de survie de 5 individus. Nous pouvons voir que certaines courbes d'individus se croisent. De plus, nous avons joué sur les paramètres de la loi du risque de base, comme je l'ai mentionné dans la section 3 , afin d'avoir des courbes de survie qui se croisent plus tôt.

3.4 Résultats sur les données simulées

Nous présentons dans cette section, les résultats de la comparaison sur l'ensemble des simulations réalisées et présentées en section 3.3. Cette comparaison est réalisée afin de voir le comportement des réseaux de neurones pour la prédiction de la survie en grande dimension. Les méthodes comparées concernent deux types de réseaux de neurones et une procédure Lasso utilisant la vraisemblance partielle de Cox. La première approche de réseaux de neurones est basée sur une modélisation de Cox, c'est le modèle Cox-nnet développé par Ching et al. (2018) et présenté en section 3.2.1. La seconde approche de réseau de neurones concerne une modélisation à temps discret. Pour cette seconde approche, nous avons étudié deux réseaux de neurones NNsurv et NNsurvK ainsi que leurs versions *profondes* (plus de couches cachées dans le réseau) que nous avons appelées NNsurv deep et NNsurvK deep. NNsurv est une adaptation du modèle de Biganzoli et al. (1998) et il est présenté en section 3.2.2. NNsurvK est une version multisites que nous proposons en section 3.2.3. Pour implémenter ces deux derniers réseaux de neurones, les hyperparamètres du réseau de neurones qui sont la taille du batch, du paramètre de régularisation sont déterminés par validation croisée. Nous avons choisi d'utiliser une fonction d'activation ReLU pour les couches cachées et les versions *profondes* ont deux couches cachées alors que les versions classiques ont une seule couche cachée.

Les réseaux de neurones basés sur un modèle à temps discret NNsurv, NNsurv deep, NNsurvK, NNsurvK

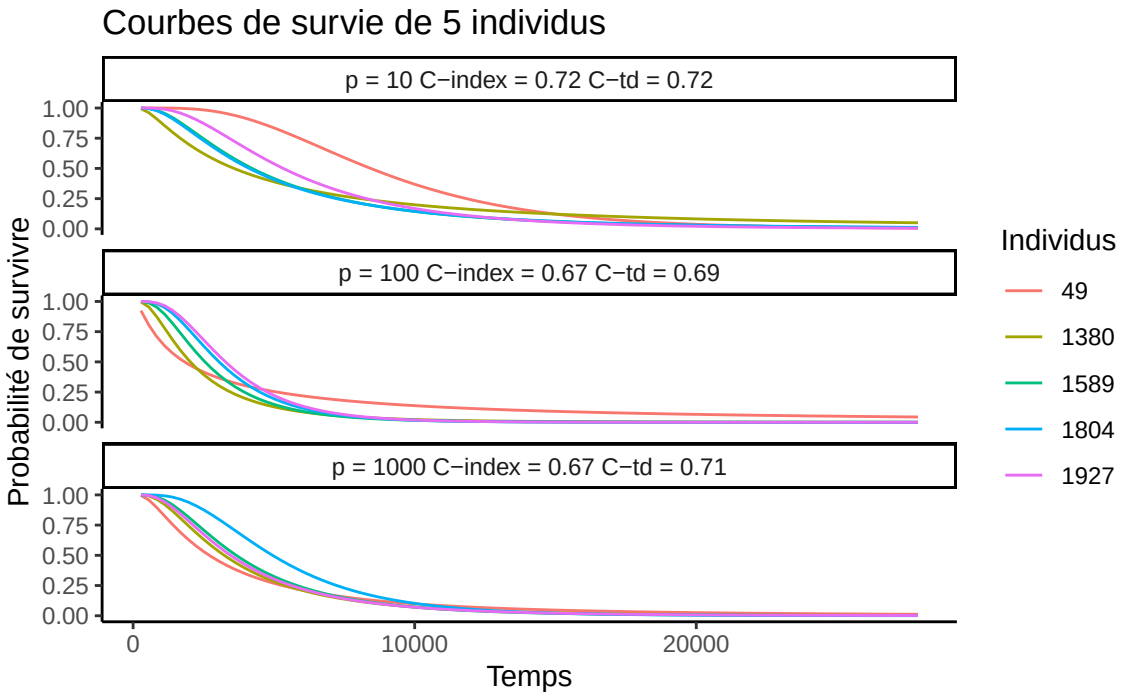


Figure 3.13 – Courbes de survie de différents individus pour la simulation AH/Log-normale

deep, le réseau de neurones Cox-nnet et le modèle de Cox précédé par une sélection de variables par une procédure Lasso (qu'on appelle par la suite coxL1) ont été exécutés sur chaque simulation présentée en section 3.3, où nous avons fait varier la taille de l'échantillon et le nombre de variables. Pour étudier les performances des différentes méthodes, nous avons utilisé les deux métriques présentées en section 1.3. La première métrique est le C_{td} permettant d'évaluer la discrimination, elle évalue si l'ordre est conservé entre les prédictions estimées par le modèle et les valeurs observées. Plus la valeur du C_{td} est proche de 1, meilleur est le modèle. La deuxième métrique est le Score de Brier Intégré permettant d'évaluer à la fois la discrimination et la calibration du modèle. La calibration correspond à la différence entre les événements observés et les prédictions, la calibration évalue la précision de la prédiction. Si la calibration d'un modèle est correcte pour le risque de décès, alors on s'attendra à ce que x individus sur 100 décèdent parmi les individus qui ont un risque prédit de $x\%$. Plus la valeur de l'IBS est proche de 0, meilleur est le modèle. De plus, nous avons calculé l'IBS de référence et le C_{td} de référence car dans le cadre de nos simulations, nous connaissons de façon exacte le modèle. Remarquons que les valeurs de ces métriques peuvent cependant être supérieures (dans le cas du C_{td}) ou inférieures (dans le cas de l'IBS) à celles des prédictions obtenues selon les différentes méthodes à cause de la génération aléatoire des données correspondant à la courbe de survie modélisée.

3.4.1 Configuration 1

1 Résultats pour la simulation Cox/Weibull

La simulation Cox/Weibull correspond aux données simulées à partir d'un modèle de Cox dont le risque de base est modélisé par une loi de Weibull. Dans cette simulation, les données respectent l'hypothèse des risques proportionnels. Les résultats de cette simulation montrent que Cox-nnet a les meilleures performances pour ce qui est du C_{td} dans toutes les configurations (quels que soient le nombre de variables ou la taille de l'échantillon) et dans la plupart des configurations en ce qui concerne l'IBS. Les meilleures valeurs de l'IBS de Cox-nnet, comme nous pouvons le voir sur la Table 3.5, sont pour une taille d'échantillon égale à 200 et un nombre de variables égal à 10 et 100 ou une taille d'échantillon égale à 1000 et un nombre de

variables égal à 100 et 1000. CoxLi a également le meilleur IBS (*i.e.* le plus faible) pour une taille d'échantillon égale à 1000 et un nombre de variables égal à 10. Ces bons résultats de CoxLi et Cox-nnet ne sont pas étonnants car cette simulation est réalisée à partir d'un modèle de Cox et ces deux méthodes sont basées sur une modélisation de Cox. Nous pouvons observer sur la Table 3.5 que NNsurv deep obtient la plus faible valeur de l'IBS pour un nombre d'individus égal à 200 et un nombre de variables égal à 1000. Nous pouvons également voir que les valeurs de l'IBS de NNsurv et NNsurv deep sont très proches des valeurs de l'IBS de référence. C'est notamment le cas quand la taille d'échantillons est égale à 1000 et le nombre de variables est égal à 100. De plus, nous pouvons observer sur la Table 3.5 que certaines de les valeurs de C_{td} obtenues pour NNsurv et NNsurv deep sont proches de celles de Cox-nnet. C'est notamment le cas quand la taille d'échantillon est égale à 200 et le nombre de variables est égal à 10 ainsi que quand le nombre d'échantillons est égal à 1000 et le nombre de variables est égal à 10 et 100. Nous pouvons voir que certaines valeurs de C_{td} pour les réseaux de neurones à temps discret sont meilleures que celles obtenues à partir du modèle de Cox, comme par exemple pour une taille d'échantillon égale à 200 et un nombre de variables égal à 100 ou à 1000 et pour une taille d'échantillon égale à 1000 et quel que soit le nombre de variables.

	n	200			1000		
Méthode	p	10	100	1000	10	100	1000
Référence	$C^{\star}_{td}$	0.7442	0.7428	0.7309	0.7442	0.7428	0.7309
	IBS*	0.0471	0.0549	0.0582	0.0471	0.0549	0.0582
NNsurv	C_{td}	0.7137	0.6224	0.5036	0.7398	0.7282	0.5700
	IBS	0.0980	0.0646	0.1359	0.0759	0.0537	0.1007
NNsurvK	C_{td}	0.6261	0.5135	0.5173	0.7312	0.6504	0.5699
	IBS	0.1310	0.1121	0.1137	0.1178	0.1011	0.1130
NNsurv deep	C_{td}	0.7225	0.5982	0.5054	0.7424	0.7236	0.5741
	IBS	0.0878	0.0689	**0.1080**	0.0591	0.0555	0.1185
NNsurvK deep	C_{td}	0.6178	0.4784	0.4112	0.7112	0.5772	0.4748
	IBS	0.1324	0.1122	0.1561	0.1179	0.1023	0.1260
Cox -nnet	C_{td}	**0.7313**	**0.6481**	**0.5351**	**0.7427**	**0.7309**	**0.6110**
	IBS	**0.0688**	**0.0622**	0.1402	0.0640	**0.0498**	**0.0710**
CoxLi	C_{td}	0.7292	0.5330	0.5011	0.7419	0.7243	0.5
	IBS	0.0715	0.0672	0.1175	**0.0541**	0.0509	0.0770

Table 3.5 – Résultats pour l'ensemble des méthodes sur la simulation Cox/Weibull

Les courbes présentées sur la figure 3.14 vont dans le même sens que les résultats de l'IBS observés dans la Table 3.5. Nous pouvons voir sur le premier graphe à gauche de la Figure 3.14 que CoxLi et Cox-nnet ont les courbes proches de la courbe de l'estimateur de Kaplan-Meier. Nous pouvons également remarquer que la courbe de NNsurv est proche de celle de l'estimateur de Kaplan-Meier, mais s'éloigne de celle-ci pour des temps plus longs. Dans ce cas, la version *deep learning* de NNsurv ne permet pas d'améliorer les performances de NNsurv. Les courbes de NNsurvK et NNsurvK deep sont vraiment très éloignées de la courbe de l'estimateur de Kaplan-Meier. Nous avons un comportement assez similaire sur le deuxième graphe à gauche correspondant à la configuration pour une taille d'échantillon égale à 200 et un nombre de variables égal à 100. Les courbes de CoxLi et Cox-nnet se superposent à celle de l'estimateur de Kaplan-Meier. Les courbes de NNsurv et NNsurv deep sont proches de celles de l'estimateur de Kaplan-Meier. Les courbes de NNsurvK et NNsurvK deep sont toujours éloignées de celle de Kaplan-Meier. Pour le troisième graphe de la Figure 3.14 représentant les résultats des différentes méthodes pour une taille d'échantillon égale à 200 et un nombre

de variables égal à 1000, toutes les méthodes ont de grandes difficultés à s'approcher de la courbe de l'estimateur de Kaplan-Meier. Ce sont les courbes de NNsurvK et NNsurvK deep qui sont étonnament la plus proche de la courbe de l'estimateur de Kaplan-Meier et notamment pour des temps plus longs. Une explication possible de ce résultat est que NNsurvK et NNsurvK deep permettent de réduire les variations entre les coefficients des variables. En effet, NNsurv permet de gérer les variables dépendantes du temps par son architecture. Mais la pénalité utilisée pour NNsurvK permet de réduire les variations entre variables et permet donc d'obtenir les mêmes coefficients pour des temps différents. Si nous nous intéressons aux résultats quand la taille d'échantillon est égale à 1000 correspondant aux graphes présentés à droite de la Figure 3.14, nous pouvons observer que le comportement des courbes de l'ensemble des méthodes est similaire pour un nombre de variables égal à 10 ou à 100. NNsurv, NNsurv deep, CoxLi et Cox-nnet ont des courbes de survie qui se superposent avec celles de Kaplan-Meier. Nous pouvons observer que les courbes de CoxLi et Cox-nnet sont légèrement plus proches à la courbe de l'estimateur de Kaplan-Meier que celles de NNsurv et NNsurv deep. Comme pour un nombre d'individus égal à 200, les courbes de NNsurvK et NNsurvK deep sont très éloignées de la courbe de l'estimateur de Kaplan-Meier. Pour un nombre de variables égal à 1000, les courbes des méthodes que nous comparons ont également des difficultés à s'approcher de la courbe de l'estimateur de Kaplan-Meier. Mais ces différences entre les courbes est moindre pour 1000 individus comparé à 200 individus. CoxLi et Cox-nnet ont des courbes les plus proches de celle de Kaplan-Meier, leurs courbes arrivent à se superposer sur la courbe de l'estimateur de Kaplan-Meier à la fin des temps de survie. La Figure 3.14 semble montrer que CoxLi et Cox-nnet ont les meilleurs résultats dans cette simulation, ce qui n'est pas surprenant car elle est basée sur un modèle de Cox. On constate néanmoins que les performances de NNsurv sont très proches de celles basées sur le modèle de Cox. Par ailleurs, le réseau plus profond a de meilleures performances.

Synthèse :
Sans surprise, Cox-nnet a les meilleurs résultats sur ce jeu de données simulées à partir d'un modèle Cox avec une loi de Weibull. Nous avons également pu voir que les réseaux de neurones basés sur un modèle à temps discret (NNsurv et NNsurv deep) ont des performances proches dans certains cas. Avec 1000 individus et une centaine de covariables, NNsurv et NNsurv deep semblent très performants.

2 Résultats pour la simulation AFT/Log-normale

Cette section présente les résultats pour la simulation réalisée à partir d'un modèle AFT avec un risque de base modélisé par une loi log-normale. La Table 3.6 présente les résultats des différentes méthodes étudiées dans ce manuscrit (NNsurv, NNsurv deep, NNsurvK, NNsurvK deep, Cox-nnet et CoxLi) en s'appuyant sur deux métriques (C_{td} et IBS) dans différents cas (différentes tailles d'échantillons et un nombre de variables différents). La spécifité de ces données simulées est qu'elles ne vérifient pas l'hypothèse des risques proportionnels, mais les courbes de survie ne se croisent pas.

La Table 3.6 montre que CoxLi et Cox-nnet ont les meilleurs résultats dans la plupart des configurations en considérant le C_{td} ou l'IBS. C'est notamment le cas pour le C_{td} quand la taille de l'échantillon est égale à 200 ou quand la taille d'échantillon est égale à 1000 et le nombre de variables est égal à 10 et 100. Le C_{td} obtenu par le modèle de CoxLi est égal à 0.9867 pour 200 individus et 10 variables et celui de Cox-nnet est égal à 0.9060 pour 1000 individus et 100 variables. Nous pouvons voir dans la Table 3.6 que le C_{td} obtenu pour les réseaux de neurones basés sur un modèle à temps discret est très proche de ceux obtenus par CoxLi et Cox-nnet et qu'il est soit supérieur à celui de référence ou soit faiblement en dessous. Par exemple, pour une taille d'échantillon égale à 200 et un nombre de variables égal à 10 le C_{td} de NNsurv est égal à 0.9832, celui de Cox-nnet est égal à 0.9867 et celui de référence est égal à 0.9203. Nous avons le même comportement pour 100 variables et une même taille d'échantillon ou pour 100 variables et une taille d'échantillon égale à 1000.

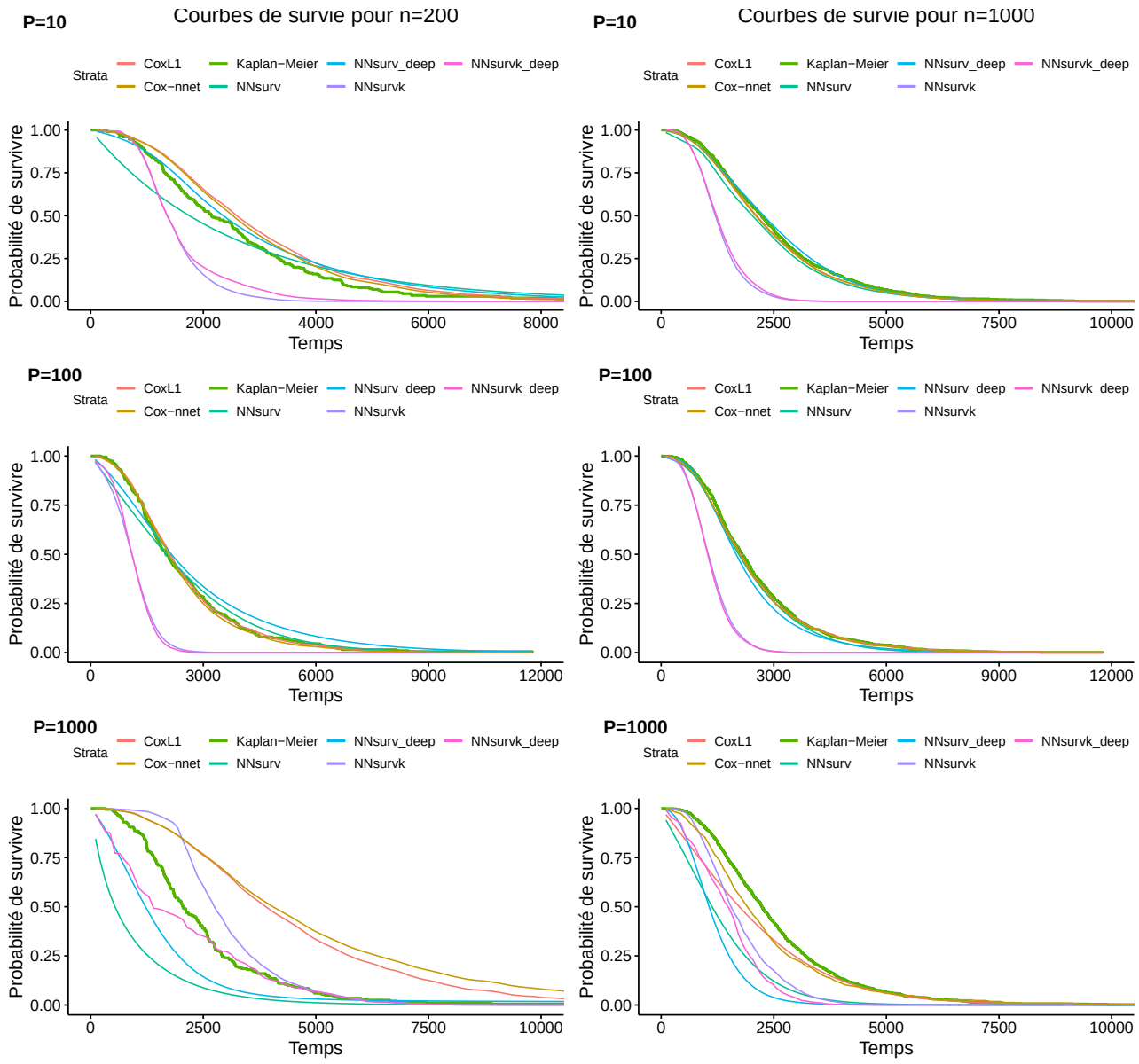


Figure 3.14 – Courbes de survie obtenues à partir des méthodes pour la simulation Cox/Weibull

NNsurvK deep obtient la meilleure valeur de C_{td} pour une taille d'échantillon égale à 1000 et un nombre de variables égal à 10. Ces bons résultats de NNsurvK pourrait venir de la pénalisation utilisée. En effet, NNsurvK utilise une pénalisation de type fused Lasso, permettant de gérer l'absence ou la présence des risques proportionnels. De plus, les valeurs de l'IBS sont les plus faibles pour les méthodes basées sur une modélisation de Cox dans le plupart des situations. Mais les valeurs de l'IBS pour NNsurv et NNsurv deep sont également très bonnes. Elles sont plus faibles que celles de l'IBS de référence dans de nombreuses situations et sont très proches de celles de CoxL1 et Cox-nnet. Nous pouvons observer ces résultats quand le nombre de variables est inférieur ou égal à 100 quel que soit la taille de l'échantillon. Les bons résultats de CoxL1 et Cox-nnet pourraient nous paraître surprenants, mais cela est expliqué par le fait que nous avons simulés ces données à partir d'un modèle AFT dont les courbes de survie ne se croisent pas. Pour cette simulation, la prédiction de la fonction de survie obtenue par CoxL1 et Cox-nnet ne se croiseront pas et est sûrement plus proche de la fonction de survie de la simulation AFT comparée aux réseaux de neurones à temps discret.

	n	200			1000		
Méthode	p	10	100	1000	10	100	1000
Référence	$C^{\star}_{td}$	0.9203	0.9136	0.9037	0.9203	0.9136	0.9037
	IBS*	0.0504	0.0604	0.0417	0.0504	0.0604	0.0417
NNsurv	C_{td}	0.9832	0.8349	0.5425	0.9851	0.9038	0.7426
	IBS	0.0265	**0.0560**	0.2577	0.0247	0.0188	0.0642
NNsurvK	C_{td}	0.9802	0.7118	0.5575	0.9856	0.8707	0.6049
	IBS	0.1425	0.1043	0.1468	0.1319	0.0820	0.0979
NNsurv deep	C_{td}	0.9786	0.8275	0.5576	0.9857	**0.9060**	**0.7500**
	IBS	0.0295	0.0561	0.1886	0.0261	0.0207	**0.0631**
NNsurvK deep	C_{td}	0.9791	0.6976	0.5694	**0.9861**	0.8716	0.6090
	IBS	0.1079	0.1049	0.1905	0.0984	0.0657	0.1334
Cox -nnet	C_{td}	0.9825	**0.8558**	**0.5979**	0.9844	**0.9060**	0.7085
	IBS	**0.0122**	0.0906	**0.0959**	0.0126	0.0374	0.0808
CoxL1	C_{td}	**0.9867**	0.7827	0.5091	0.9856	0.9028	0.5349
	IBS	0.0146	0.0965	0.0960	**0.0077**	**0.0182**	0.0827

Table 3.6 – Résultats pour l'ensemble des méthodes sur la simulation AFT/Log-normale

Synthèse :
Pour les données simulées à partir d'un modèle AFT avec une loi log-normale, Cox-nnet est les réseau de neurones avec les meilleurs résultats dans la plupart des situations quand la taille de l'échantillon est petite. Quand la taille de l'échantillon augmente, NNsurv deep est le meilleur modèle en considérant le C_{td} dans la plupart des situations. De plus, NNsurv et NNsurv deep semblent également avoir de bonnes performances quand le nombre de variables est inférieur ou égale à 100. Nous pensons que les bons résultats de Cox-nnet peuvent s'expliquer par le faible niveau de complexité de ces données. En effet, les courbes de survie des individus de ce jeu de données ne se croisent jamais.

3 Résultats pour la simulation AH/Log-normale

Les résultats présentés dans la Table 3.7 et sur la Figure 3.16 sont ceux obtenus sur la simulation AH avec le risque de base qui suit une loi log-normale. Dans cette simulation, les risques ne sont pas proportionnels et les fonctions de survie se croisent.

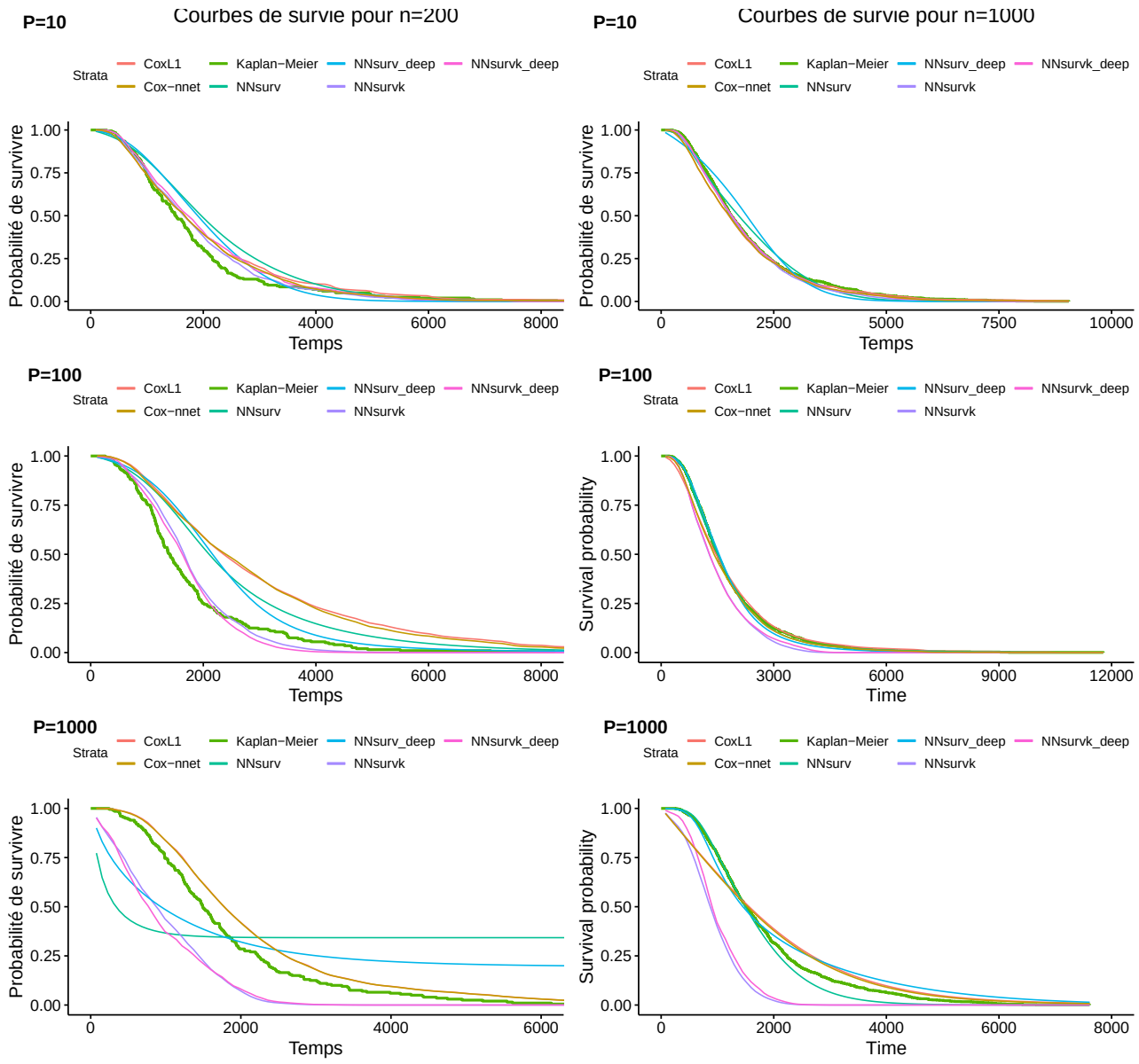


Figure 3.15 – Courbes de survie obtenues à partir des méthodes pour la simulation AFT/Log-normale

Nous pouvons observer que les réseaux de neurones basés sur un modèle à temps discret ont les meilleures performances au regard du C_{td} et de l'IBS. En effet, leurs valeurs sont proches des valeurs de C_{td} et de l'IBS de référence. C'est notamment le cas pour l'IBS quand la taille d'échantillon est égale à 1000, les valeurs de l'IBS de NNsurv et NNsurv deep sont inférieures à celles de l'IBS de référence. En revanche, les méthodes basées sur la vraisemblance partielle de Cox ont les plus hautes valeurs de C_{td} pour une petite taille d'échantillon (n=200) et un petit nombre de variables (p=10) ou au contraire pour une grande taille d'échantillon (n=1000) et un grand nombre de variables (p=1000). Pour une taille d'échantillon égale à 200, les réseaux de neurones basés sur un modèle à temps discret ont des valeurs de C_{td} plus élevées que celles obtenues par CoxL1 et Cox-nnet. De plus, nous pouvons également voir sur la FIGURE 3.16 que les courbes des réseaux de neurones basés sur un modèle à temps discret sont plus proches de la courbe de l'estimateur de Kaplan-Meier et notamment pour la configuration où le nombre de variables est égal à 1000. Pour une taille d'échantillon plus grande, cela semble moins clair. Nous pouvons voir sur la FIGURE 3.16 que la courbe de NNsurv est plus proche de la courbe de l'estimateur de Kaplan-Meier quand le nombre de variables est égal à 100 et à 1000. Mais Cox-nnet semble plus proche quand le nombre de variables est égal à 10. Cependant, nous pouvons observer que les résultats de CoxL1 et Cox-nnet arrivent à conserver des performances honorables, leurs courbes ne sont pas si éloignés de la courbe de Kaplan-Meier. Cela est confirmé par les valeurs obtenues de l'IBS par les deux méthodes utilisant la vraisemblance partielle de Cox. Pour un faible nombre d'individus (n=200), les valeurs de l'IBS de CoxL1 et Cox-nnet sont très hautes. Par exemple, Cox-nnet obtient des valeurs de l'IBS égales respectivement à 0.2243 et 0.1609 pour un nombre de variables égal à 10 et à 100 et CoxL1 obtient des valeurs de l'IBS égales respectivement à 0.2278 et 0.1614. Ces valeurs sont très hautes comparées à celles de l'IBS de référence. CoxL1 et Cox-nnet ont donc plus de difficultés avec un faible nombre d'échantillon. Les prédictions de ces deux méthodes sont plus éloignées que celles données par les réseaux de neurones qui ne sont pas construits à partir du modèle de Cox.

	n	200			1000		
Méthode	p	10	100	1000	10	100	1000
Référence	$C_{td}^{\star}$	0.7225	0.6857	0.7070	0.7225	0.6867	0.7070
	IBS*	0.0755	0.0316	0.0651	0.0755	0.0316	0.0651
NNsurv	C_{td}	0.6863	**0.5971**	**0.5358**	0.7084	0.6088	0.5654
	IBS	0.1247	**0.0780**	**0.0859**	0.0699	0.0347	0.0533
NNsurvK	C_{td}	0.6151	0.5258	0.5025	0.7107	0.6214	0.5159
	IBS	0.1267	0.1087	0.1396	0.1020	0.0459	0.0790
NNsurv deep	C_{td}	0.7042	0.5793	0.5325	**0.7155**	**0.6450**	0.5702
	IBS	0.1789	0.2529	0.1554	**0.0602**	**0.0303**	**0.0484**
NNsurvK deep	C_{td}	0.6067	0.4847	0.5025	0.7138	0.5570	0.5199
	IBS	**0.1234**	0.1058	0.1328	0.1048	0.0451	0.0558
Cox -nnet	C_{td}	**0.7128**	0.5812	0.5356	0.7097	0.6047	**0.5720**
	IBS	0.1342	0.2243	0.1609	0.0843	0.0875	0.0553
CoxL1	C_{td}	0.7042	0.5219	0.5112	0.7088	0.5597	0.5
	IBS	0.1350	0.2278	0.1614	0.0608	0.0408	0.0553

TABLE 3.7 – Résultats pour l'ensemble des méthodes sur la simulation AH/Log-normale

Synthèse :
Sur le jeu de données simulé à partir d'un modèle AH avec une loi log-normale, les réseaux de neurones basés sur le modèle à temps discret ont les meilleures performances dans la majorité des situations. Les réseaux de

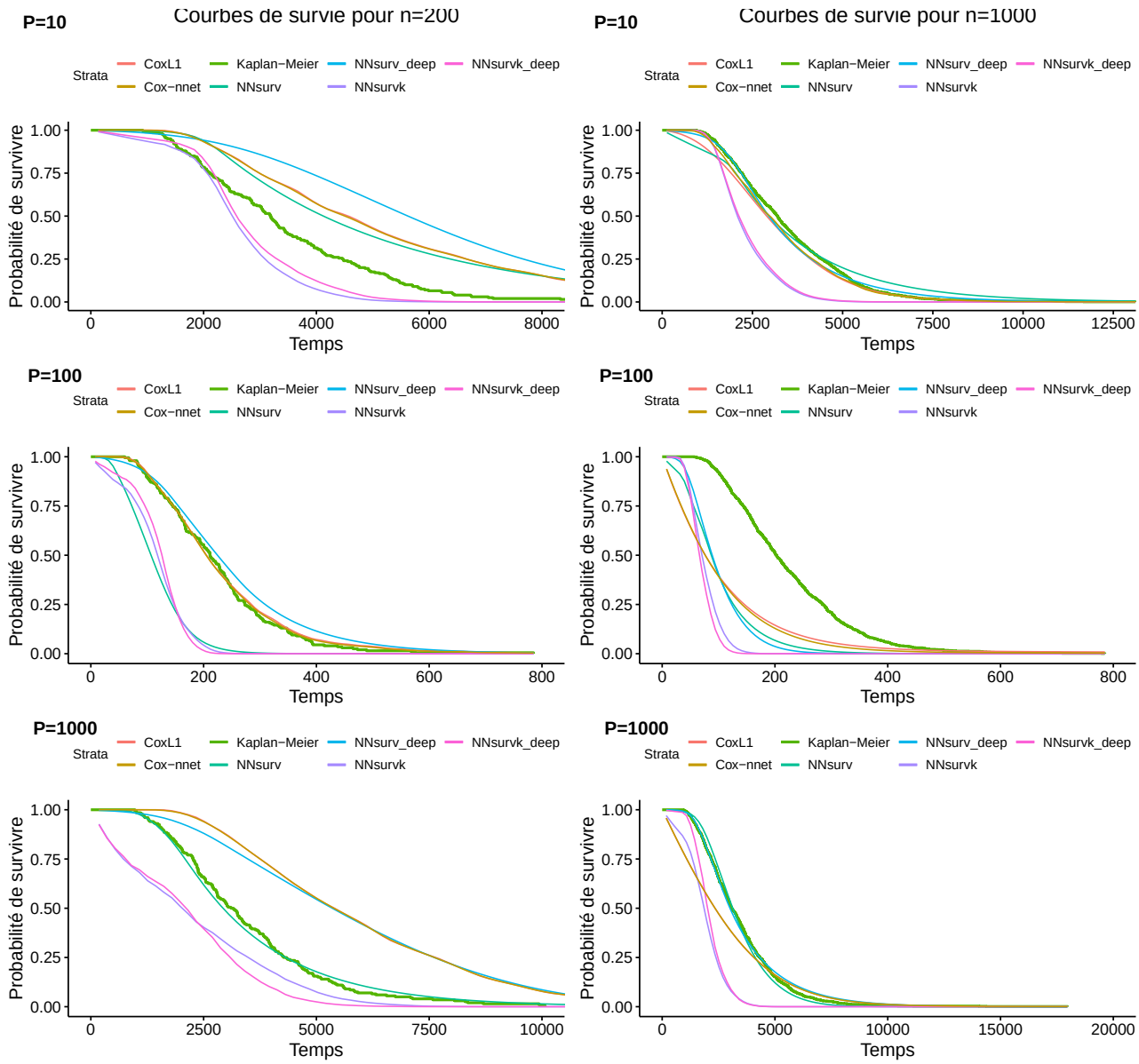


Figure 3.16 – Courbes de survie obtenues à partir des méthodes pour la simulation AH/Log-normale

neurones basés sur un modèle à temps discret ont toujours les meilleures valeurs de l'IBS, ce qui signifie que la précision de la prédiction est la meilleure avec ces derniers. Quand les données sont complexes, les réseaux de neurones basés sur un modèle à temps discret ont de meilleures performances que les méthodes basées sur la modélisation de Cox. Dans ce jeu de données, les risques proportionnels ne sont pas vérifiés et les courbes de survie se croisent.

4 Résultats pour la simulation modifiée AFT/Log-normale

Dans cette section, nous présentons les résultats des différentes méthodes pour la simulation basée sur un modèle AFT modifié avec un risque de base qui suit une loi log-normale. Nous avons modifié le modèle AFT afin que les courbes de survie se croisent. La Table 3.8 détaille les résultats des diférentes méthodes sur cette simulation.

Les méthodes NNsurv et NNsurv deep sont les méthodes qui réussissent le mieux à discriminer les individus dans la plupart des cas, c'est-à-dire que si le temps d'événement d'un individu i est plus grande que celui d'un individu j alors un bon modèle prédira une survie plus grande pour l'individu i. Il y a cependant deux configurations où CoxLi et Cox-nnet ont des bons résultats de discrimination. Par exemple, quand le nombre de variables et la taille d'échantillon sont faibles le C_{td} de CoxLi et Cox-nnet valent respectivement 0.8449 et 0.8448. Celles-ci sont très proche de la valeur du C_{td} de référence. Ce sont également les plus hautes valeurs de C_{td} parmi toutes les méthodes, mais le C_{td} de NNsurv deep est assez proche du C_{td} de ces dernières et de celui de référence, il est égal à 0.8385 (cf.Table 3.8). Comme pour le C_{td}, nous pouvons observer que les valeurs de l'IBS sont les plus faibles pour CoxLi et Cox-nnet quand le nombre de variables est faible, bien que celle de NNsurv deep soit proche. Sur la Table 3.8, nous pouvons voir que la valeur de l'IBS pour CoxLi et Cox-nnet sont respectivement égales à 0.0354 et 0.0347 pour 10 variables et 200 individus et sont également égales à 0.0267 et 0.0323 pour 10 variables et 1000 individus. Les valeur de l'IBS de NNsurv deep sont égales à 0.0487 pour 10 variables et 200 individus et à 0.0363 pour 10 variables et 1000 individus. Nous pouvons remarquer sur la Table 3.8 que NNsurv deep a les valeurs les plus faibles de l'IBS quand le nombre de variables augmente et que la taille d'échantillon est grande (0.0312 pour 100 variables et 0.0510 pour 1000 variables). De plus, ces valeurs sont également très proches de la valeur de l'IBS de référence. Dans un cadre de grande dimension, NNsurv Deep semble avoir un meilleur pouvoir de calibration.

La Figure 3.17 va dans le sens des résultats de la Table 3.8 en ce qui concerne CoxLi et Cox-nnet. En effet, nous remarquons sur les deux graphes en haut de la Figure 3.17 que les courbes de CoxLi et Cox-nnet sont les plus proches de la courbe de l'estimateur de Kaplan-Meier. Nous pouvons également voir que les courbes de NNsurv et NNsurv deep sont plus proches de la courbe de l'estimateur de Kaplan-Meier que les courbes de CoxLi et Cox-nnet quand le nombre de variables augmente. C'est notamment la cas pour les configurations où le nombre de variables est égal à 1000 quelle que soit la taille d'échantillon et pour 1000 individus et 10 variables. De plus, nous pouvons remarquer que NNsurv deep semble avoir de meilleures performances quand le nombre d'échantillons est grand alors qu'au contraire NNsurv semble avoir de meilleures performances quand ce nombre d'échantillons est petit. Ceci peut s'expliquer par le plus grand nombre de paramètres à calibrer pour le modèle profond.

Synthèse :
Pour la simulation des données réalisée à partir d'un modèle modifié AFT avec un loi log-normale, le comportement des réseaux de neurones est le même que celui de ces méthodes sur la simulation précédente concernant le modèle AH. Dans la plupart des situations, les réseaux de neurones basés sur un modèle à temps discret ont les meilleures performances mais Cox-nnet reste proche. De même, le modèle de CoxLi a de bonnes performances quand le nombre d'échantillons est grand par rapport au nombre de variables.

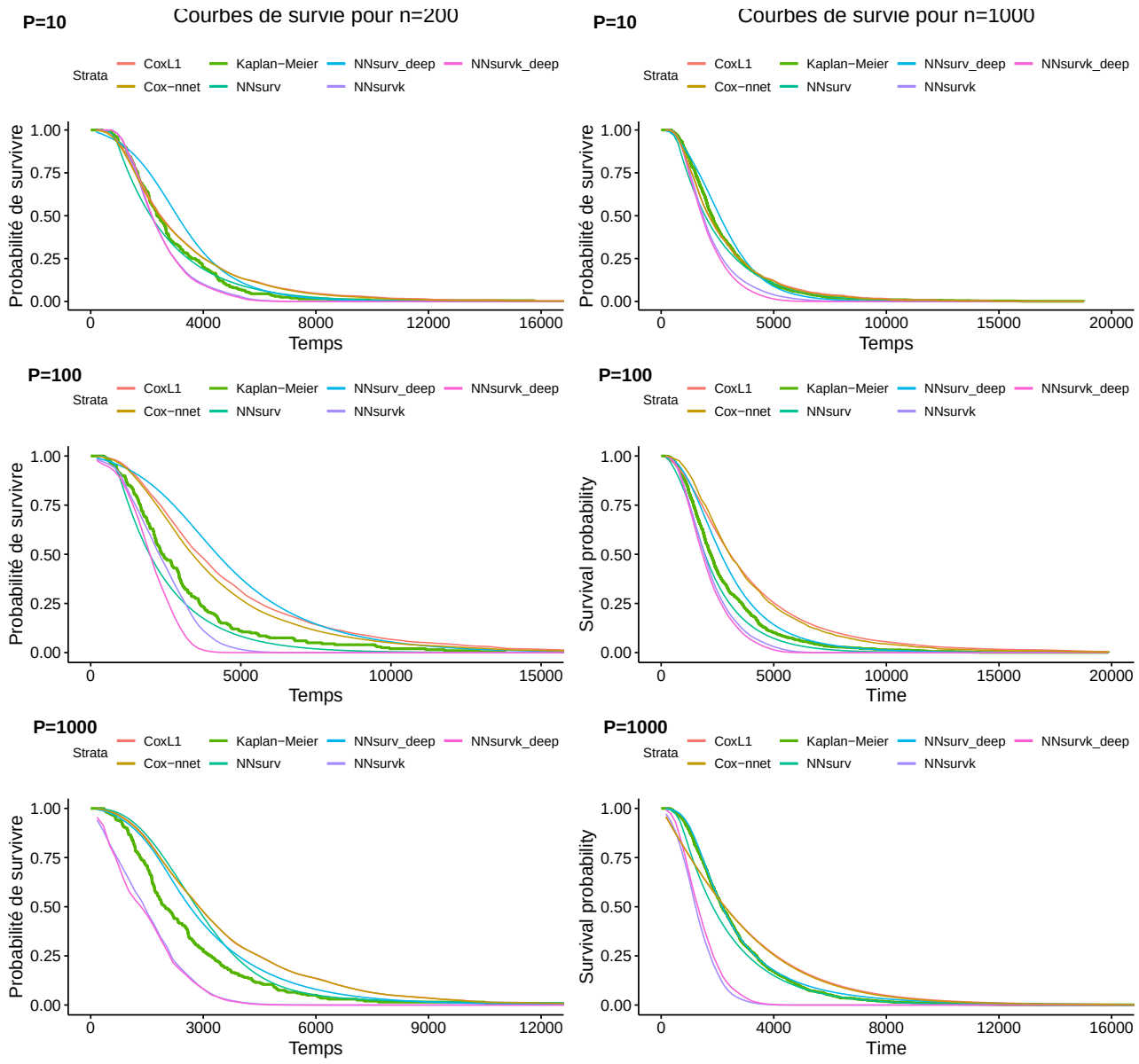


Figure 3.17 – Courbes de survie obtenues à partir des méthodes pour la simulation AFT/Log-normale modifiée

	n	200			1000		
Méthode	p	10	100	1000	10	100	1000
Référence	$C^{\star}_{td}$	0.8468	0.8589	0.8459	0.8468	0.8589	0.8459
	IBS*	0.0294	0.0199	0.0305	0.0294	0.0199	0.0305
NNsurv	C_{td}	0.8080	**0.7764**	0.5607	0.8404	0.8391	**0.7098**
	IBS	0.0624	0.0775	**0.0669**	0.0532	0.0564	0.0651
NNsurvK	C_{td}	0.8197	0.5870	0.5610	0.8404	0.7990	0.6154
	IBS	0.0859	0.1003	0.1235	0.0771	0.0759	0.0856
NNsurv deep	C_{td}	0.8385	0.7746	**0.6028**	**0.8463**	0.8361	0.7021
	IBS	0.0487	0.0897	0.0759	0.0363	**0.0312**	**0.0510**
NNsurvK deep	C_{td}	0.7941	0.4673	0.5559	0.8394	0.7716	0.6011
	IBS	0.0838	0.0942	0.1237	0.0735	0.0744	0.0843
Cox -nnet	C_{td}	0.8448	0.7747	0.5916	0.8441	**0.8410**	0.6678
	IBS	**0.0347**	**0.0717**	0.0819	0.0323	0.0680	0.0622
CoxL1	C_{td}	**0.8449**	0.5893	0.5168	0.8457	0.8381	0.5456
	IBS	0.0354	0.0933	0.0818	**0.0267**	0.0429	0.0628

Table 3.8 – Résultats pour l'ensemble des méthodes sur la simulation modifiée AFT/Log-normale

3.4.2 Configuration 2 : effet de la censure

1 Résultats pour la simulation modifiée AFT/Log-normale censurée

Nous introduisons maintenant les résultats des différentes méthodes étudiées dans ce manuscrit à partir de la simulation d'un modèle AFT modifié avec un risque qui suit une loi log-normale. Mais cette simulation a été créée afin de prendre en compte la censure des données. Le taux de censure dans cette simulation est d'environ 50%. Les résultats sont donnés dans la table 3.9.

Nous pouvons observer sur la Table 3.9 qu'aucune des méthodes n'est la plus performante dans tous les cas quand la taille d'échantillons est petite (n=200). Les réseaux de neurones basés sur le modèle à temps discret semblent avoir un meilleur pouvoir de discrimination quand le nombre de variables augmente. En effet, les valeurs de C_{td} de NNsurv deep sont les plus proches du C_{td} de référence. En revanche, les méthodes basées sur la modélisation de Cox ont de meilleures performances de calibration et de discrimination quand le nombre de variables est petit. En effet, les valeurs de C_{td} et de l'IBS sont respectivement de 0.8643 et de 0.0602 pour Cox-nnet. Quand la taille d'échantillon augmente, le réseau de neurones avec les meilleures performances est NNsurv deep. Les valeurs du C_{td} de ce dernier sont celles qui sont les plus proches de l'IBS de référence quel que soit le nombre de variables. De plus, quand le nombre de variables est plus important (supérieur ou égale à 100) les réseaux de neurones basés sur un modèle à temps discret ont un meilleur pouvoir de calibration. Nous pouvons observer cela, par exemple, pour 100 variables et 1000 individus où l'IBS de NNsurv deep est égal à 0.0587. Cette valeur est très proche de celle de l'IBS de référence (0.0569). Cependant, Cox-nnet réussit à avoir un meilleur pouvoir de calibration que celui de NNsurv deep, notamment quand le nombre de variables est petit. L'IBS de Cox-nnet est égal à 0.0529 et celui-ci est très proche du C_{td} de référence (0.0473). Une meilleure précision de la prédiction de la survie par Cox-nnet peut s'expliquer par le fait que la modélisation de Cox est connue pour sa capacité à gérer les données censurées.

Synthèse :

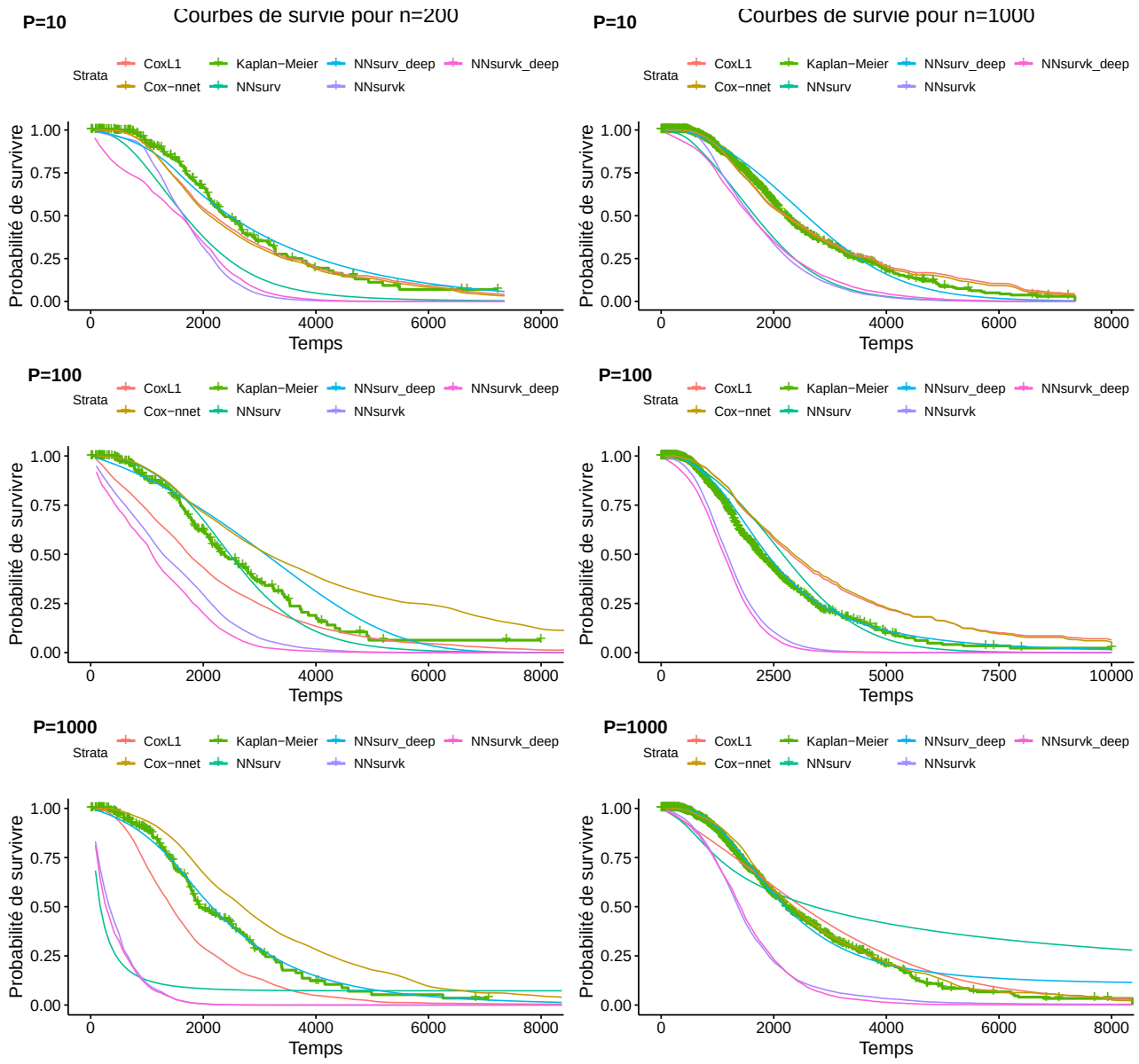


Figure 3.18 – Courbes de survie obtenues à partir des méthodes pour la simulation AFT/Log-normale censurée

	n	200			1000		
Méthode	p	10	100	1000	10	100	1000
Référence	$C^{\star}_{td}$	0.8718	0.8917	0.8765	0.8718	0.8917	0.8765
	IBS*	0.0473	0.0569	0.0482	0.0473	0.0569	0.0482
NNsurv	C_{td}	0.8600	**0.8086**	0.5175	0.8697	0.8706	0.6990
	IBS	0.1064	**0.1009**	0.2866	0.1335	0.0673	0.1952
NNsurvK	C_{td}	0.8063	0.6810	0.5422	0.8591	0.7866	0.6063
	IBS	0.1704	0.1946	0.2856	0.1961	0.1550	0.1523
NNsurv deep	C_{td}	0.8431	0.7168	**0.5463**	**0.8710**	**0.8739**	**0.7155**
	IBS	0.1212	0.1268	**0.1142**	0.0869	**0.0587**	0.1013
NNsurvK deep	C_{td}	0.8193	0.5633	0.5217	0.8435	0.7466	0.5921
	IBS	0.1925	0.2038	0.2883	0.2018	0.1593	0.1520
Cox -nnet	C_{td}	**0.8643**	0.8038	0.5	0.8697	0.8730	0.7145
	IBS	0.0613	0.1233	0.1192	**0.0529**	0.0844	**0.0961**
CoxL1	C_{td}	0.8623	0.6107	0.5309	0.8694	0.8659	0.5160
	IBS	**0.0602**	0.1340	0.1394	0.0667	0.0799	0.1142

Table 3.9 – Résultats pour l'ensemble des méthodes sur la simulation modifiée AFT/Log-normale censurée

Sur ce jeu de donnés simulées prenant en compte la censure, les résultats sont assez similaires à ceux obtenus à partir d'un modèle AFT avec une loi log-normale sans censure, mais légèrement plus faible que ceux de la simulation modifiée AFT avec une loi log-normale sans censure. Cette baisse de performance des réseaux de neurones basés sur un modèle à temps discret sur ce jeu de données peut s'expliquer par la présence de la censure. Le défaut de ces réseaux de neurones est qu'une partie de l'information est perdue quand les données sont censurées. En effet, ces réseaux de neurones, qui sont des modèles utilisant les intervalles de temps pour estimer le risque instantané de décès, négligent des informations quand les données sont censurées à cause de la discrétisation des intervalles de temps. Les temps observés sont classés en intervalles et il est indiqué pour chaque individu si l'évènement d'intérêt a lieu dans l'intervalle. Les individus qui sont censurés dans la seconde partie de l'intervalle sont donc supposés vivant jusqu'à la fin de l'intervalle. La modélisation de CoxL1 est connue pour sa capacité à gérer les données censurées de façon optimale.

3.4.3 Configuration 3 : effet de la sparsité

1 Résultats pour la simulation modifiée AFT/Log-normale sparse

Nous introduisons maintenant les résultats des différentes méthodes étudiées dans ce manuscrit à partir de la simulation d'un modèle AFT modifié avec un risque qui suit une loi log-normale. Mais cette simulation a été créée afin de prendre en compte la sparsité des données. Les résultats sont donnés dans la table 3.10.

Sur cette simulation, la procédure Lasso utilisant la vraisemblance partielle de Cox (CoxL1) a les meilleurs résultats au regard du C_{td} pour tous les cas à l'exception du cas où il y a 1000 individus et seulement 10 variables. La méthode CoxL1 a également les meilleures performances quand on s'intéresse à la métrique IBS quand la taille d'échantillon est grande. NNsurv deep a les valeurs les plus faibles de l'IBS quand la taille d'échantillon est petite et a des valeurs de l'IBS proches de celles de CoxL1 ainsi que celles de référence quand la taille d'échantillon est grande. Par exemple, la valeur de l'IBS de NNsurv deep est de 0.0304, celle de CoxL1 est de 0.0291 et celle de référence est égale à 0.0284 pour 1000 individus et 100 variables. Nous

pouvons également remarquer que les valeurs de C_{td} obtenues pour NNsurv et NNsurv deep sont proches de celles de Cox-nnet. Seules les deux versions de NNsurvK ont des difficultés quand le nombre de variables devient important. Ces résultats, notamment la bonne performance de CoxL1, ne sont pas surprenants car le Lasso est une méthode de régularisation en norme l_1 qui permet de sélectionner les variables pertinentes et la méthode retrouve donc nos variables non nulles de notre simulation.

	n	200			1000		
Méthode	p	10	100	1000	10	100	1000
Référence	$C^{\star}_{td}$	0.8673	0.8673	0.8673	0.8673	0.8673	0.8673
	IBS*	0.0284	0.0284	0.0284	0.0284	0.0284	0.0284
NNsurv	C_{td}	0.8684	0.8012	0.5902	0.8766	0.8646	0.7436
	IBS	0.1254	0.1129	0.0738	0.0621	0.1566	0.0622
NNsurvK	C_{td}	0.8648	0.5215	0.5581	**0.8770**	0.8511	0.6566
	IBS	0.1094	0.0987	0.0995	0.0899	0.0872	0.0835
NNsurv deep	C_{td}	0.8744	0.8062	0.5938	0.8761	0.8664	0.7284
	IBS	**0.0474**	**0.0488**	**0.0739**	0.0378	0.0304	0.0487
NNsurvK deep	C_{td}	0.8610	0.5100	0.5263	0.8746	0.8227	0.5835
	IBS	0.1099	0.0992	0.1091	0.0913	0.0848	0.0869
Cox -nnet	C_{td}	0.8742	0.7922	0.5832	0.8757	0.8683	0.6952
	IBS	0.0885	0.0773	0.1015	0.0532	0.0519	0.0699
CoxL1	C_{td}	**0.8759**	**0.8686**	**0.8733**	0.8739	**0.8743**	**0.8726**
	IBS	0.0904	0.0805	0.0754	**0.0300**	**0.0291**	**0.0290**

Table 3.10 – Résultats pour l'ensemble des méthodes sur la simulation modifiée AFT/Log-normale sparse

La première remarque que nous pouvons faire sur les deux premiers graphes en haut de la Figure 3.19 est que les courbes de NNsurvK et NNsurvK deep sont proches de la courbe de l'estimateur de Kaplan-Meier. Nous pouvons voir que sur tous les graphes à droite de la Figure 3.19 la courbe de CoxL1 est très proche de la courbe de l'estimateur Kaplan-Meier, ce qui va dans le sens des bons résultats de calibration (IBS). Enfin, nous pouvons observer sur les deux derniers graphes que la courbe de NNsurv est proche de celle de Kaplan-Meier. Cela est aussi en adéquation avec les résultats de l'IBS obtenus dans la Table 3.10 pour NNsurv.

Synthèse :

Comme nous pouvions nous y attendre, les résultats sur la simulation sparse à partir d'un modèle modifié AFT avec une loi log-normale montre une très bonne performance de CoxL1. Quand on utilise CoxL1 pour prédire la survie, la première étape consiste à appliquer une procédure Lasso qui met à zéro toutes les variables non-informatives. Seules les variables pertinentes sont donc gardées pour prédire la survie et permettent d'améliorer la prédiction. Mais nous avons pu observer que dans certains cas, NNsurv deep réussissait à surpasser les performances de CoxL1 en considérant l'IBS. Cela peut s'expliquer par la complexité des données. Les courbes de survie entre plusieurs individus peuvent se croiser et comme le modèle de Cox est un modèle à risques proportionnels, les courbes de survie estimées seront parallèles entre individus.

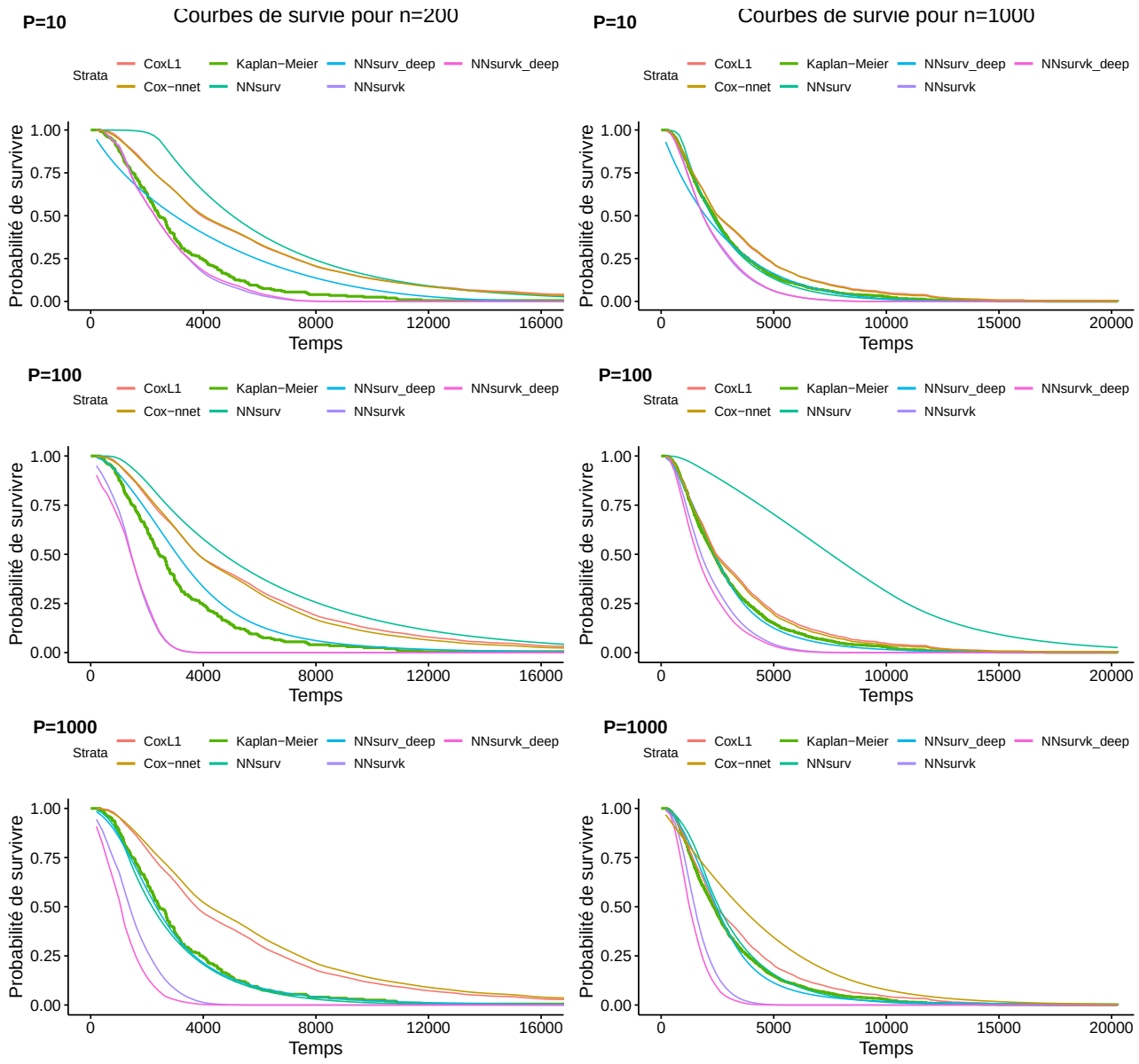


Figure 3.19 – Courbes de survie obtenues à partir des méthodes pour la simulation AFT/Log-normale sparse

3.4.4 Discussion des résultats sur les données simulées

Pour conclure sur la configuration 1, nous avons pu voir que les méthodes basées sur la log-vraisemblance partielle de Cox avaient de bons résultats sur les données de la simulation Cox/Weibull et les bonnes performances de ces modèles comparées à celles des réseaux de neurones à temps discret étaient attendues. Quand le nombre de variables est grand, la performance des réseaux de neurones est bien meilleure que celle de la procédure CoxL1. Quand les données de survie deviennent plus complexes comme c'est le cas pour la simulation AFT/Log-normale modifiée, les réseaux de neurones à temps discret s'en sortent mieux et plus particulièrement pour des plus grands échantillons. Cependant, nous avons pu également observer que CoxL1 et Cox-nnet avaient de bons résultats sur cette simulation. Pour la simulation AH/Log-normale, la conclusion est assez similaire à la simulation précédente, mais CoxL1 et Cox-nnet semblent avoir plus de difficultés. Quand les données de survie ont un niveau élevé de complexité, les réseaux de neurones basés sur un modèle à temps discret sont les plus performants. Ils ont notamment de très bonnes performances quand la taille d'échantillon est élevée et que le nombre de variables est d'environ 100. Parmi ces réseaux de neurones, NNsurv deep est celui avec les meilleurs résultats.

L'étude concernant l'effet de la censure a montré des résultats similaires à ceux que nous avons obtenus sur la simulation sans censure. Les réseaux de neurones basés sur un modèle à temps discret ont de bonnes performances pour les simulations avec censure mais celles-ci sont légèrement plus faibles que sur la simulation sans censure. En revanche, les performances pour les méthodes basées sur la modélisation de Cox restent similaires entre les données censurées et non-censurées. Ces résultats peuvent s'expliquer par le fait que les réseaux de neurones basés sur un modèle à temps discret réduisent les temps observés en intervalle et perdent donc une partie de l'information quand les données sont censurées.

L'étude concernant l'effet de la sparsité (configuration 3) a montré que Cox précédé par une procédure Lasso avait les meilleures performances. Ce résultat confirme la très bonne capacité de la procédure Lasso en sélection quand le nombre de variables pertinentes est faible. Les résultats des réseaux de neurones montrent que NNsurv deep est le plus performant sur cette simulation.

3.5 Illustration sur données réelles

3.5.1 Jeu de données du cancer du rein à cellules claires

Pour illustrer les performance des réseaux de neurones sur les données réelles, nous nous sommes tout d'abord intéressés aux données réelles du cancer du rein à cellules claires (ccRCC). Celles-ci sont les données utilisées dans le chapitre 2 pour l'étude de détection de marqueurs provenant de la base de données TCGA (*The Cancer Genome Atlas*). Les données sont disponibles sur le site `https://www.cancer.gov/tcga`. Dans ce jeu de données, nous avions accès à 533 patients et le nombre de variables, correspondant est de 17 789. Ce nombre correspond à un ensemble filtré (où les gènes avec une expression nulle sont supprimés) des gènes codants présents chez l'humain. De plus, nous avons utilisé les résultats obtenus dans la section 2.2 en appliquant les différentes méthodes de régularisation et de *Screening* étudiées. Nous avons donc pris l'intersection des résultats des méthodes SIS, ISIS et PSIS pour créer une première sélection et nous avons pris les résultats de coxCS pour faire une seconde sélection. Nous avons séparé les méthodes de cette manière car la méthode coxCS effectue une pré-sélection en utilisant la connaissance biologique alors que les autres méthodes SIS, ISIS et PSIS font une pré-sélection en utilisant un score basé sur la vraisemblance partielle de Cox. Pour évaluer les performances des différentes méthodes, nous avons séparé le jeu de données en trois ensembles un d'entraînement, un de validation et un de test. Une procédure de validation croisée a également été réalisée pour obtenir les hyperparamètres des méthodes. Nous commençons par présenter les résultats sur les données du cancer du rein à cellules claires sans sélection en section 1 et nous montrons ensuite les résultats sur

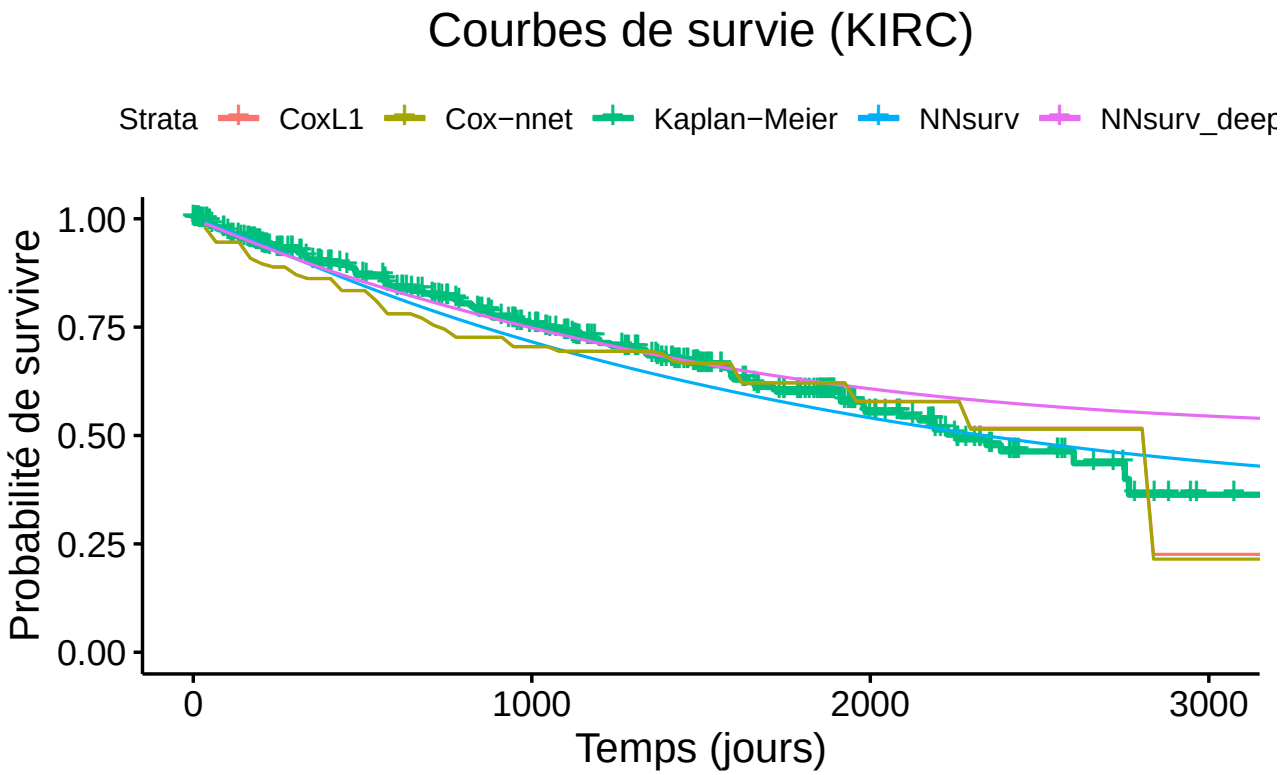


Figure 3.20 – Courbes de survie obtenues à partir des méthodes sur le jeu de données du cancer du rein à cellules claires (ccRCC/KIRC)

les données du rein à cellules claires avec sélection en section 2 .

1 Sans sélection

		CoxL1	Cox-nnet	NNsurv Deep	NNsurv
KIRC	C_{td}	0.5115	0.5277	**0.5741**	**0.5741**
	IBS	**0.2069**	0.2076	0.2869	0.2491

Table 3.11 – Résultats sur les données du cancer du rein à cellules claires

La Table 3.11 résume les résultats de l'ensemble des méthodes pour le jeu de données ccRCC (également appelé KIRC). Nous pouvons observer que NNsurv et NNsurv deep ont les meilleures performances en considérant le C_{td}. Leurs valeurs sont égales à 0.5741. Sur ce jeu de données, les réseaux de neurones basés sur un modèle à temps discret sont les méthodes qui ont le meilleur pouvoir de discrimination. En revanche, CoxL1 a la meilleure performance en considérant l'IBS. La précision de la prédiction est meilleure pour CoxL1. Ce résultat peut s'expliquer par le fait qu'une procédure Lasso est utilisée avant d'appliquer le modèle de Cox. Les variables les plus pertinentes sont donc gardées et cela améliore la prédiction. De plus, le taux de censure de ce jeu de données est élevé, ce qui pourrait expliquer la différence des résultats entre les réseaux de neurones basés sur un modèle à temps discret et celui basé sur la modélisation de Cox.

2 Avec sélection

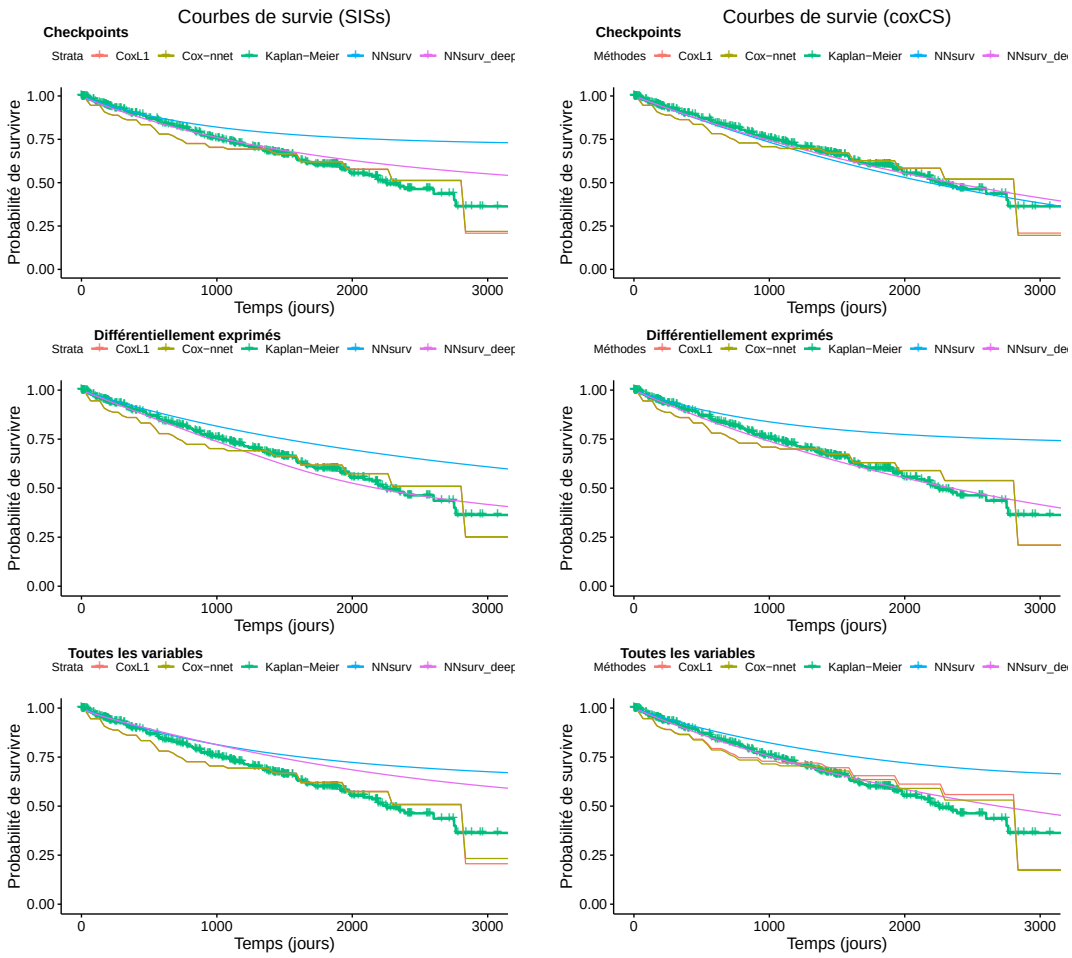
Nous présentons dans cette section les résultats des différentes méthodes étudiées dans ce manuscrit qui sont des méthodes basées sur la vraisemblance partielle de Cox et des réseaux de neurones basés sur un modèle à

temps discret. Les deux méthodes utilisant la vraisemblance partielle de Cox sont une méthode de régularisation Lasso et le réseau de neurones Cox-nnet.

		CoxL1	Cox-nnet	NNsurv Deep	NNsurv
SISs checkpoints	C_{td}	0.4807	0.5090	0.4988	**0.5239**
	IBS	0.2098	**0.2077**	0.2508	0.3067
SISs Différentiellement exprimés	C_{td}	0.5841	**0.6018**	0.6006	0.5720
	IBS	0.2004	**0.1983**	0.2314	0.2533
SISs Toutes les variables	C_{td}	0.5831	0.5929	**0.5950**	0.5835
	IBS	0.1945	**0.1926**	0.2515	0.2815
coxCS checkpoints	C_{td}	0.4670	0.4959	0.5056	0.5024
	IBS	0.2081	**0.2071**	0.2299	0.2275
coxCS Différentiellement exprimés	C_{td}	0.5647	0.5718	**0.5957**	0.5637
	IBS	0.2093	**0.2070**	0.2305	0.3073
coxCS Toutes les variables	C_{td}	0.5069	0.5160	0.5148	0.5019
	IBS	0.2368	**0.2131**	0.2372	0.3245

Table 3.12 – Résultats des différentes méthodes sur une partie des données du cancer du rein à cellules claires. La partie des données correspond au résultat de sélection de la méthode de *Screening* utilisée.

Nous pouvons voir dans la Table 3.12 que Cox-nnet est la méthode avec les meilleurs résultats de calibration. Les valeurs de l'IBS pour Cox-nnet sont les plus faibles quelle que soit la sélection réalisée (SISs ou CoxCS) et le jeu de données sur lequel est réalisé la sélection (*Immune Checkpoints*, gènes différentiellement exprimés ou l'ensemble des gènes). Nous pouvons également observer que les valeurs de l'IBS sont très proches de celles de Cox-nnet et celles de NNsurv deep restent dans le même ordre de grandeur. Par exemple, la valeur de l'IBS pour Cox-nnet est égale 0.2071 pour la cas coxCS *checkpoints* et celles de CoxL1 et NNsurv deep sont respectivement de 0.2081 et 0.2299. En revanche, en s'intéressant au pouvoir de discrimination nous pouvons voir que les réseaux de neurones basés sur un modèle à temps discret ont de meilleurs performances. Seul le cas où la sélection réalisée par SISs et faite sur les gènes différentiellement exprimés donne une valeur plus élevée de C_{td} pour Cox-nnet. Sa valeur est égale à 0.6018 alors que la valeur de C_{td} de NNsurv deep est égale à 0.6006. Ces deux valeurs sont très proches. De plus, nous pouvons remarquer la version *deep learning* de NNsurv n'améliore pas de manière spectaculaire le C_{td}. En revanche, nous pouvons voir une amélioration des résultats de l'IBS. Il passe d'environ 0.30 pour NNsurv à 0.20 pour NNsurv deep. Par exemple, si nous nous intéressons à la dernière ligne de la Table 3.12 nous pouvons voir que la valeur de l'IBS de NNsurv est de 0.3245 contre 0.2372 pour la valeur de l'IBS de NNsurv deep. Il semble avoir le même comportement pour NNsurv et NNsurv deep, mais seulement en considérant le C_{td}. Il y aurait un bénéfice à utiliser la procédure SISs en amont pour améliorer les performances de discrimination. Mais cela ne semble pas vrai pour améliorer le pouvoir de calibration des méthodes. Finalement, le nombre de variables d'entrée ne semble pas avoir d'impacts sur les performances des différentes méthodes. La Figure 3.21a confirme les bons résultats de CoxL1 et Cox-nnet. Nous pouvons observer que leurs courbes de survie sont proches de celle de Kaplan-Meier et cela renforce le fait qu'il est plutôt judicieux d'utiliser la sélection SISs pour de meilleures performances de CoxL1 et Cox-nnet. Nous pouvons voir que NNsurv deep est bon sur les gènes différentiellement exprimés. La Figure 3.21b montre que les courbes des différentes méthodes sont proches de la courbe de Kaplan-Meier excepté pour NNsurvK. Nous pouvons également observer sur la Figure 3.21b que NNsurv deep maintient de bonnes performances dans toutes les configurations ce qui n'est pas le cas de NNsurv.



(a) Méthodes de sélection utilisée SISs (intersection des méthodes SIS, ISIS et PSIS)

(b) Méthodes de sélection utilisées coxCS

Figure 3.21 – Résultats de l'ensemble des méthodes étudiées pour la prédiction de la survie en grande dimension sur les données du cancer du rein à cellules claires avec 2 sélections différentes

3 Comparaison entre "avec sélection" et "sans sélection" pour le jeu de données KIRC

Dans cette section, nous comparons les résultats des méthodes sur le jeu de données du rein à cellules claires avec ou sans sélection. Nous pouvons constater sur la Table 3.11 et la Table 3.12 que les résultats de CoxL1 sont améliorés quand une sélection de variables est réalisée. Seule une sélection avec SIS permet d'améliorer les résultats de Cox-nnet. Sur la Table 3.12, on peut voir que la valeur de Cox-nnet est égale à 0.5929 contre 0.5277 sans sélection (cf.Table 3.11). NNsurv deep a également de meilleures performances quand une sélection avec SIS est réalisée en amont. Il semble malgré tout que sur ce cas test le signal reste relativement faible puisque les meilleures performances sont de l'ordre de 0.6 pour le C_{td}, contre 0.5 dans le cas d'une prévision au hasard, et de 0.19 pour l'IBS contre 0.25 quand on prédit par la moyenne.

3.5.2 Jeu de données du cancer du sein

Description des données : Les données METABRIC (pour *Molecular Taxonomy of Breast Cancer International Consortiulm*)(Curtis et al., 2012) proviennent d'un projet entre le Canada et le Royaume-Uni qui a inclus 2509 patients atteint d'un cancer du sein à un stade précoce. Ces données sont accessibles à l'adresse suivante : `https://www.synapse.org/#!Synapse:syn1688369/wiki/27311`. La durée de

survie, les variables cliniques et les données d'expressions étaient présentes pour 1981 patients. Le jeu de données METABRIC utilisé est constitué de 1981 patients, de 6 variables cliniques (âge, taille de la tumeur, hormonothérapie, chimiothérapie, grades de la tumeur) et de 863 gènes. Le pourcentage d'individus censurés est égal à 55%.

Résultats :
Les résultats du jeu de données METABRIC sont résumés dans la Table 3.13. Nous pouvons voir que NNsurv deep réussit à obtenir la valeur la plus haute du C_{td}. Le C_{td} de NNsurv est équivalent à celui de Cox, mais Cox-nnet a une valeur plus faible. Le score de Brier intégré est très proche pour les méthodes NNsurv deep, Cox-nnet et Cox précédé d'un procédure Lasso, bien que c'est cette dernière qui a la valeur de l'IBS la plus faible. Elle vaut 0.1937 contre 0.1965 pour Cox-nnet et 0.1972 pour NNsurv deep. Enfin, nous avons tracé sur la Figure 3.22 les courbes de survie moyenne des différentes méthodes étudiées et la courbe de survie estimée par l'estimateur de Kaplan-Meier. Nous pouvons voir que les courbes de CoxL1 et Cox-nnet se superposent et sont très proches de celles de Kaplan-Meier. Le fait que les courbes de CoxL1 et Cox-nnet se superposent avait déjà été remarqué sur les résultats des données simulées. De la même façon, les courbes de NNsurv et NNSurv deep se superposent, mais NNsurv deep est légèrement plus proche de la courbe de Kaplan-Meier. Enfin, nous pouvons observer que les courbes de CoxL1 et Cox-nnet sont plus proches de celle de Kaplan-Meier quand les temps de survie sont faibles et qu'au contraire les courbes de NNsurv et NNsurv deep sont proches de celle de l'estimateur de Kaplan-Meier quand les temps de survie sont longs.

		CoxL1	Cox-nnet	NNsurv Deep	NNsurv
Metabric	C_{td}	0.6757	0.6676	**0.6853**	0.6728
	IBS	**0.1937**	0.1965	0.1972	0.2038

Table 3.13 – Résultats des différentes méthodes sur le jeu de données du cancer du sein (METABRIC)

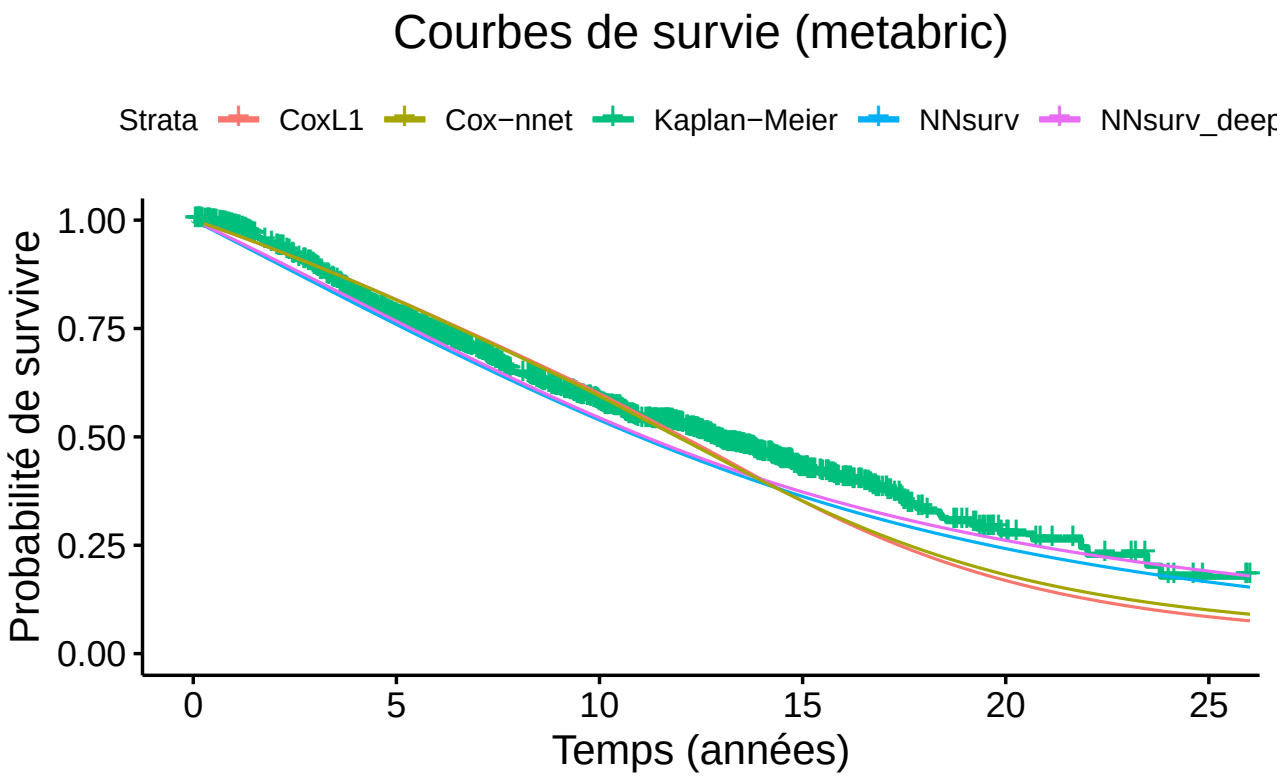


Figure 3.22 – Courbes de survie obtenues à partir des méthodes sur le jeu de données du cancer du sein provenant du projet METABRIC

Conclusion et perspectives

Dans cette thèse, nous avons étudié d'une part la détection de marqueurs en grande dimension, et d'autre part la prédiction de la durée de survie en grande dimension. Les contributions proposées dans cette thèse permettent de donner un nouvel éclairage sur certaines questions, mais certaines restent en suspens et de nouvelles apparaissent. Nous proposons des extensions possibles.

Contributions

Sélection de variables

L'un des objectifs de cette thèse concernait la sélection de variables en oncologie. Cet objectif s'est décliné en deux questions. La première a porté sur la détection de marqueurs caractérisant les cellules tumorales chez les patients atteints du cancer du rein à cellules claires (ccRCC). La seconde concernait la détection de marqueurs influençant la durée de survie.

Sélection de variables pour la détection de marqueurs

Dans le chapitre 2, nous avons commencé par nous intéresser à la découverte de nouveaux marqueurs pour caractériser le cancer du rein à cellules claires (ccRCC). Cette première partie a été le travail d'une collaboration avec l'équipe de l'hôpital Saint-Louis/CEA dont les recherches se concentrent sur l'investigation de nouvelles immunothérapies pour les patients atteints de ccRCC. Pour trouver de nouveaux marqueurs, nous avons tout d'abord réalisé une analyse différentielle. Nous avons par la suite proposé une seconde approche plus originale qui consiste à appliquer une méthode d'apprentissage supervisée (RFE) sur une sélection de gènes. Cette sélection préliminaire est obtenue à partir des résultats de l'analyse différentielle. Cette méthode nous a permis d'obtenir les gènes les plus importants pour caractériser le ccRCC. L'utilisation de ces deux approches a permis de souligner des nouveaux gènes qui semblent impliqués dans le ccRCC. De plus, la seconde analyse que nous avons proposée a permis de mettre en évidence l'expression simultanée de certains gènes dans les cellules tumorales. Ces résultats ont donné un nouvel éclairage sur les cibles potentielles pour traiter le ccRCC.

Sélection de variables pour la survie

Cette seconde partie a concerné la détection de marqueurs impactant la durée de survie des patients. Depuis l'arrivée du séquençage à haut-débit, les variables utilisées en analyse de survie ne sont plus seulement cliniques mais également génomiques. La grande dimension est donc désormais à considérer dans le cadre de l'analyse de survie. Pour prendre en compte ce grand nombre de variables auxquelles nous avons accès, les méthodes de régularisation et de *Screening* ont été développées. Nous nous sommes concentrés sur ces méthodes dans la seconde partie du chapitre 2 en les appliquant sur les données du ccRCC pour sélectionner les gènes influençant la survie des patients. Des gènes potentiellement intéressants pour expliquer la survie ont

été mis en évidence. De plus, certaines méthodes nécessitent de la connaissance biologique. Nous avons proposé une démarche basée sur l'extraction d'informations à partir d'*abstracts* pour obtenir cette connaissance quand celle-ci était manquante. Enfin, ces méthodes de sélection sont connues pour avoir des problèmes d'instabilité quand la dimension des données augmente. Nous avons donc proposé un indice de similarité pour quantifier la stabilité des méthodes. Nous avons pu conclure que les méthodes de *Screening* étaient plus stables que les méthodes de régularisation sur le jeu de données du ccRCC et les deux méthodes dont la stabilité est la plus forte sont Ridge pour les méthodes de régularisation et ISIS pour les méthodes de *Screening*.

Étude des réseaux de neurones pour la prédiction de survie

La dernière contribution de mon travail de thèse a consisté à étudier les réseaux de neurones pour la prédiction de la survie en grande dimension. Dans ce contexte, les méthodes usuelles comme par exemple le modèle de Cox ne peuvent plus être exécutées. Plusieurs types de méthodes (comme les méthodes de réduction de dimensions ou les méthodes d'apprentissage) ont été proposés, mais notre intérêt s'est dirigé vers les réseaux de neurones. L'analyse de survie a vu deux approches de réseaux de neurones se développer pour prédire la survie. La première approche est basée sur le modèle de Cox, mais introduit un réseau de neurones pour la détermination du risque. De nombreux réseaux de neurones appartenant à cette approche ont été proposées. La seconde approche est basée sur un modèle à temps discret. Les réseaux de neurones basés sur cette approche ont peu été étudiés en grande dimension. Nous nous sommes donc particulièrement concentré sur ce type de réseau de neurones. Nous avons adapté le réseau de neurones proposé par Biganzoli et al. (1998) à la grande dimension et nous avons ensuite proposé un nouveau réseau de neurones en modifiant ce dernier. Nous avons modifié l'architecture du réseau de neurones (plusieurs sorties au lieu d'une) ainsi que l'indicateur de censure et la pénalisation. Nous avons également ajouté une couche cachée dans le réseau de neurones afin de l'adapter à la grande dimension. Dans le chapitre 3, nous avons présenté une étude comparative pour observer l'impact des différents modèles pour la prédiction de la survie dans le contexte de la grande dimension. Nous avons donc comparé un réseau de neurones basé sur le modèle de Cox (Cox-nnet) avec ceux basés sur un modèle à temps discret adapté à la grande dimension (NNsurv, NNsurv deep et NNsurvK). La procédure Lasso utilisant la log-vraisemblance partielle de Cox a été utilisée comme référence. Pour évaluer cette comparaison de manière rigoureuse, nous avons créé un plan de simulations. Les données ont été simulées à partir de différents modèles (Cox, AFT et AH). Cela a permis d'avoir des données avec des différents niveaux de complexité et nous avons fait varier le nombre de variables ainsi que la taille de l'échantillon. L'effet de la censure et de la sparsité a également été pris en compte dans cette étude de comparaison. Nous avons pu conclure, à partir de cette étude, que le meilleur réseau de neurones dans la plupart des situations est celui basé sur la modélisation de Cox. Le réseau de neurones utilisant cette modélisation (Cox-nnet) permet de gérer les effets non-linéaires ainsi que les interactions. Cependant, le réseau de neurones basé sur la modélisation à temps discret permettant de prédire directement le risque instantané, avec plusieurs couches cachées (NNsurv deep), a montré sa supériorité dans les situations les plus complexes, notamment en présence de risques non-proportionnels et de courbes de survie se croisant. Typiquement, pour une cohorte de l'ordre du millier, les simulations montrent que les réseaux de neurones ont de très bonnes performances pour une centaine de variables, ce qui laisse supposer la nécessité de sélectionner les variables en amont pour réduire la dimension. C'est effectivement ce qui est confirmé sur le premier jeu de données réelles concernant le ccRCC (également appelé KIRC). Sur le deuxième cas test concernant les données réelles Metabric, les modèles basés sur les réseaux de neurones sont performants, mais seulement marginalement supérieurs à la procédure d'estimation Lasso basé sur la log-vraisemblance partielle de Cox, ce qui laisse supposer peu de non-linéarité et d'interactions dans ce cas.

Perspectives

Nous évoquons dans cette section les perspectives possibles pour les deux problématiques abordées dans cette thèse.

Sélection de variables pour la détection de marqueurs

Dans le deuxième chapitre, nous nous sommes intéressés à la détection de marqueurs en grande dimensions à la fois pour caractériser le cancer du rein à cellules claires et pour expliquer la survie des patients. Pour caractériser le cancer du rein à cellules claires, nous avons proposé une démarche utilisant la connaissance biologique qui provenait de la collaboration avec l'équipe de l'hôpital Saint-Louis/CEA. Les résultats obtenus donnent surtout des perspectives aux biologistes, ils doivent maintenant valider les conclusions de l'étude à partir d'expérimentations biologiques. Une contrainte de cette démarche est d'avoir de la connaissance biologique. Une perspective quand celle-ci est manquante serait d'extraire de l'information biologique à partir d'*abstracts* comme cela a été réalisé pour l'étude de détection de marqueurs en survie.

Pour expliquer la durée de survie des patients, nous avons étudié des méthodes de régularisation et de *Screening* qui ont été développées pour répondre à la problématique de la grande dimension. Mais des problèmes d'instabilité ont été identifiés pour ces méthodes. Les résultats de notre étude ont montré que les méthodes de *Screening* étaient plus stables que les méthodes de régularisation. Pourtant, elles souffrent aussi d'instabilité. Le travail doit encore être poursuivi dans ce domaine afin de rendre ces méthodes plus stables. Une première idée pourrait concerner le choix du seuil. En effet, Fan et al. (2009) recommandait d'utiliser une certaine valeur de seuil dépendant du modèle considéré. Les auteurs de PSIS (Zhao et al., 2012) ont proposé une solution, il serait pertinent d'approfondir ce travail car nos résultats montrent que PSIS n'arrive pas à maintenir une stabilité avec l'augmentation de la dimension des données. Enfin, nous n'avons pas étudié les méthodes d'ensemble comme par exemple les forêts aléatoires (Breiman, 2001). En effet, les forêts aléatoires ont été développées par Breiman (2001) et ont été adaptées au cadre de la survie par Ishwaran et al. (2008). L'avantage de ces modèles est leur stabilité, il serait donc intéressant de faire une étude plus large de comparaison des méthodes pour la sélection de variables en grande dimension. L'indice de similarité que nous avons proposé dans cette thèse pourrait être utilisé mais semble toutefois pouvoir être amélioré lui aussi. De plus, notre étude de comparaison a été effectuée sur des données réelles, mais cela serait pertinent de faire cette même étude sur des données simulées où la vérité est connue. Enfin, une dernière perspective sur les méthodes de sélection serait de prendre en compte les interactions entre les variables. Un travail a été réalisé dans le cadre des modèles linéaires par Hao et al. (2014), mais aucun n'a été proposé jusqu'à présent dans le cadre de la survie à notre connaissance.

Étude des réseaux de neurones pour la prédiction de survie

Dans cette thèse, nous avons étudié à partir d'un plan de simulation le potentiel des réseaux de neurones pour prédire la durée de survie des patients en grande dimension. Les résultats de cette étude montre que le réseau de neurones basé sur le modèle de Cox est performant dans la plupart des cas. Cependant, quand le niveau de complexité des données augmente et notamment en présence de risques non-proportionnels et de courbes de survie se croisant, les réseaux de neurones basés sur une modélisation à temps discret, avec plusieurs couches cachées, prouvent leur supériorité. Une perspective à laquelle nous avions pensé était d'insérer une variable de temps en entrée du réseau de neurones. Mais des travaux ont été publiés par Kvamme et al. (2019a) et Kvamme et al. (2019b) cette dernière année. Kvamme et al. (2019a) a donc proposé d'ajouter dans un réseau de neurones basé sur la modélisation de Cox une variable qui va correspondre aux temps observés.

Dans notre étude de comparaison, nous avons étudié de manière succincte l'effet de la censure. Nous avons évalué les modèles avec seulement un jeu de données dont le pourcentage de censure était d'environ 50%. Mais l'étude pourrait être approfondie en considérant différents pourcentages de censure. Il en est de même en ce qui concerne la sparsité, nous avons également simulé un seul jeu de données avec seulement 3 variables importantes. Mais il serait intéressant de voir le comportement des modèles en prenant plus de variables importantes et de faire varier leur niveau d'importance.

De plus, nous avons précisé précédemment que le réseau de neurones basé sur un modèle de Cox était mis en défaut quand les risques proportionnels n'étaient plus respectés comparé aux réseaux de neurones basés sur un modèle à temps discret. Une perspective d'un réseau de neurones basé sur un modèle à temps discret serait de considérer un modèle multi-tâche. Ces méthodes peuvent être vues comme une série de régressions logistiques construites sur des intervalles de temps différents pour estimer la probabilité que l'évènement se produise à l'intérieur de chacun. Fotso (2018) a donc proposé un réseau de neurones basé sur ce type de modèles, mais celui-ci a seulement été comparé aux modèles de Cox. Cependant, les modèles, utilisant les intervalles de temps pour estimer le risque instantané de décès, négligent des informations quand les données sont censurées. En effet, les temps observés sont classés en intervalles et il est indiqué pour chaque individu si l'évènement d'intérêt a lieu dans l'intervalle. Les individus qui sont censurés dans la seconde partie de l'intervalle sont donc supposés vivant jusqu'à la fin de l'intervalle. Un autre type de réseau de neurones a donc fait son apparition et son approche est basé sur les *pseudo*-observations (Zhao et al., 2020 ; Roblin et al., 2020). L'ajout de ce modèle pour l'étude de comparaison serait une perspective de ce travail. Par manque de temps, nous nous sommes concentrés sur les effets des risques non proportionnels et des croisements des courbes de survie dans notre plan de simulation. Il serait bien sûr très intéressant de modifier les fonctions de dépendance en X (ψ_1 et ψ_2 cf. Section 3.3). Nous avons pris $\psi_1(X) = \beta^T X$, mais nous devions aussi considérer des fonctions non-linéaires (par exemples par paliers ou avec des splines) et également avec des termes d'interactions. L'intérêt des modèles basés sur les réseaux de neurones devrait être encore plus souligné dans ces cas.

Nous avons évoqué jusqu'à maintenant des perspectives concernant seulement des méthodes de *deep learning*, mais d'autres modèles sembleraient également pertinent à étudier. Par exemple, Bussy et al. (2019) a proposé un modèle de mélange pour les données de survie censurées. Le principe de la méthode est de détecter des groupes de patients avec des risques élevés ou faibles. Les auteurs de ce modèle montrent que cela améliore la prédiction de la survie comparé au modèle de Cox.

Enfin, une dernière perspective intéressante de ce travail serait d'approfondir l'étude du couplage d'une sélection de variables préalable avec des modèles de réseaux de neurones afin d'étudier une amélioration possible des performances de ceux-ci sur la prédiction de la durée de survie. La démarche évoquée est la même que celle réalisée par Jardillier et al. (2020). Cependant, il a seulement étudié les performances de prédiction à partir d'un modèle de Cox alors que nous voudrions l'étendre aux réseaux de neurones, après les premiers résultats intéressants trouvés dans cette thèse.

De façon générale, les réseaux de neurones profonds sont connus pour être performants grâce à leurs méthodes d'apprentissage efficaces dans le cas où il existe de nombreuses données, permettant de paramétrer les nombreux poids des différentes couches. En médecine personnalisée, quand on souhaite utiliser les données génomiques des patients pour une meilleure prévision, plus individualisée, les cohortes étudiées restent généralement d'un nombre relativement restreint, quelques milliers de patients dans les meilleurs cas, ce qui explique que nos résultats préliminaires semblent montrer la nécessité de réaliser une présélection des variables avant l'utilisation des modèles basés sur les réseaux de neurones.

Malgré tout, il existe des méthodes développées pour ce cadre précis que nous n'avons pas explorées et qui pourraient être intéressantes. Par exemple, l'apprentissage non supervisée de représentation (cf. Bengio et al., 2013), notamment par les autoencodeurs ou les autoencodeurs variationnels (Kingma et al., 2013), est une

méthode qui généralise l'analyse en composante principale et permet de trouver une représentation latente des données dans un espace de représentation de dimension réduite. Il existe aussi les cartes autoadaptives (SOM pour *Kohonen's Self-Organizing Map*) (Kohonen, 1990) qui sont des réseaux de neurones également fondés sur des méthodes d'apprentissage non supervisée. Cette méthode permet de représenter un ensemble de données par un ensemble de neurones organisé dans une structure de plus faible dimension. Cottrell et al. (2012) en propose une extension pour prendre en compte des données plus complexes en intégrant des mesures de dissimilarité des données soit à l'aide d'une fonction de dissimilarité soit à l'aide d'un noyau. Ces types de représentation peuvent ensuite être utilisés en entrée d'un modèle de survie et permettre un apprentissage plus facile.

Par ailleurs, l'apprentissage multi-tâche peut permettre de gérer plusieurs types de jeu de données différents mais comparables de façon conjointes, avec dans le modèle des parties communes (qui donc peuvent profiter de jeux de données plus importants pour l'apprentissage) et de parties spécifiques permettant de s'adapter à chaque cas d'étude. Goncalves et al. (2020) a montré une amélioration de performance de la prédiction de survie à 5 ans en utilisant un apprentissage multi-tâche appliqué sur un ensemble de 10 jeux de données différents dont chaque jeu concerne un type de cancer. Mais ces différents types de cancer sont tous liés aux papillomavirus. Par exemple, pour le cancer du sein, nous pourrions combiner des cohortes correspondants aux différents sous-types HER+, ER+/HER-, Triple Négatif, avec une partie commune du modèle et une partie spécifique.

Dans la même idée, des essais cliniques comparables (même type de cancer étudié, avec même type de données génomiques) ont été menés dans différents centres dans le monde. Nous pourrions imaginer mettre en oeuvre des méthodes de correction d'effets-batch (Johnson et al., 2007) sur les différents jeux de données en amont, puis combiner ces jeux de données pour entraîner un modèle unique.

Dans un contexte similaire où plusieurs sources de données sont disponibles pour les mêmes individus. Nous pourrions nous baser sur la méthode de Imbert et al. (2017) développée pour l'inférence des réseaux et qui propose d'imputer des échantillons non-disponibles de RNA-seq en se basant sur un second jeu de données où l'échantillon est présent. Les réseaux de neurones pourraient donc être appliqués sur des jeux de données avec un nombre d'individus plus important.

L'acquisition d'un grand nombre de données moléculaires permet désormais de caractériser très finement les patients, ce qui suscite un grand espoir pour développer des nouvelles voies de diagnostic, de pronostic, ou de traitement. De même, les nouveaux modèles et méthodes de l'apprentissage profond ouvrent des perspectives en terme de précision et de capacité. Malgré tout, la taille limitée des cohortes pose des problématiques statistiques et numériques d'intérêt pour les années à venir.

Annexe A

Résultats détaillés de l'analyse différentielle

IC	baseMean	log2FoldChange	lfcSE	stat	pvalue	padj
CD70	2464.64	7.16	0.33	21.74	7.95e-105	4.05e-102
ILT2	588.19	2.97	0.16	18.83	3.93e-79	6.11e-77
CD40	2304.35	1.88	0.10	18.37	2.25e-75	2.84e-73
OX40	274.55	3.02	0.17	17.92	7.72e-72	8.33e-70
HVEM	4188.37	1.66	0.09	17.54	7.26e-69	6.89e-67
4-1BB	378.36	4.29	0.24	17.52	1.08e-68	1.01e-66
CD200	1901.29	1.84	0.11	16.67	2.18e-62	1.49e-60
HLA-G	1537.52	3.97	0.24	16.65	2.97e-62	2.01e-60
B7-H4	1622.03	-4.87	0.29	-16.64	3.48e-62	2.36e-60
Galectin9	2088.86	2.21	0.14	16.33	5.81e-60	3.57e-58
4-1BBL	390.44	3.66	0.23	15.97	1.97e-57	1.05e-55
CD39	4650.97	0.99	0.07	14.96	1.43e-50	5.47e-49
ILT4	560.97	2.21	0.15	14.83	9.46e-50	3.46e-48
B7-H3	3453.01	1.23	0.08	14.61	2.47e-48	8.38e-47
LAG3	405.28	3.66	0.26	14.16	1.54e-45	4.51e-44
B7-H5	4082.56	1.32	0.09	13.87	9.85e-44	2.60e-42
CD244	116.30	2.36	0.18	13.46	2.71e-41	6.34e-40
MHC-II	37011.58	1.79	0.13	13.39	7.18e-41	1.64e-39
CD27	356.17	3.32	0.25	13.29	2.81e-40	6.23e-39
CD86	814.05	2.39	0.18	13.04	6.83e-39	1.42e-37
PD-1	172.62	3.19	0.25	12.76	2.72e-37	5.22e-36
ICOs	58.41	3.23	0.27	11.97	5.10e-33	7.84e-32
CD80	37.03	2.40	0.20	11.91	1.07e-32	1.62e-31
GITR	46.43	2.60	0.22	11.82	3.23e-32	4.74e-31
CD160	48.24	1.60	0.14	11.65	2.17e-31	3.03e-30
CD200R	174.61	2.40	0.21	11.52	1.10e-30	1.48e-29
CTLA-4	57.77	2.75	0.24	11.36	6.50e-30	8.43e-29
Kir2DL	8.52	2.57	0.24	10.53	6.22e-26	6.44e-25
CD38	376.18	1.92	0.19	10.19	2.14e-24	2.04e-23
CD48	1004.59	2.02	0.21	9.68	3.65e-22	3.04e-21
PDL2	257.56	1.25	0.13	9.54	1.44e-21	1.16e-20

OX40L	243.84	0.89	0.11	8.31	9.34e-17	5.64e-16
VHL	1525.31	-0.62	0.08	-7.97	1.54e-15	8.58e-15
CD28	233.00	1.49	0.19	7.93	2.25e-15	1.24e-14
TIM-3	3686.16	1.91	0.24	7.90	2.72e-15	1.49e-14
Bap1	4258.35	-0.45	0.06	-7.50	6.50e-14	3.25e-13
CD40L	82.55	1.53	0.22	7.12	1.08e-12	4.99e-12
S100A8	432.27	1.43	0.20	7.10	1.22e-12	5.62e-12
BTLA	53.07	1.45	0.21	6.80	1.02e-11	4.39e-11
CD203a	1110.10	-0.99	0.15	-6.45	1.11e-10	4.40e-10
S100A9	1091.46	1.16	0.19	5.99	2.07e-09	7.49e-09
PBRM1	2622.04	-0.41	0.08	-5.25	1.49e-07	4.62e-07
PDL1	377.62	0.58	0.12	4.95	7.26e-07	2.10e-06
ADOR	725.90	-0.77	0.23	-3.42	6.30e-04	1.32e-03
CD73	2137.36	0.54	0.18	3.04	2.36e-03	4.57e-03
GITRL	35.42	0.38	0.18	2.08	3.72e-02	5.87e-02
IcosL	132.13	-0.34	0.20	-1.71	8.77e-02	1.28e-01
CD157	619.91	-0.01	0.15	-0.06	9.51e-01	9.63e-01

Table A.1 – Table des résultats de DESeq2 concernant les niveaux d'expression des 44 ICs (baseMean représente la moyenne des comptages normalisés pour tous les échantillons, log2FoldChange est le logarithme du rapport du niveau des expressions de gènes entre le tissu sain et tumoral, lfcSE est l'erreur standardisée du logarithme fold change, stat représente la valeur de la statistique de test, pvalue est la p-valeur du test et padj est la p-valeur ajustée du test).

Annexe B

Présentation des indices de similarité

B.1 Diversité β

La diversité β est une mesure de biodiversité introduite par Whittaker (1960) et Whittaker (1972) en écologie. Celle-ci consiste à comparer la variation de la composition des espèces entre les sites et elle peut être calculée à partir d'indices de similarité, comme celui de Jaccard ou de Sørensen. Ces deux indices sont les plus connus. Dans la section 2.2 du chapitre 2, nous avons donc utilisé cette métrique pour étudier la stabilité des méthodes de régularisation et de *Screening*. Nous voulons comparer la variation de la composition des gènes entre les différentes *seeds*. Nous détaillons dans cette annexe les indices de similarité comme définis en écologie et nous donnons les résultats de l'indice de Jaccard pour les deux types de méthodes sur notre jeu de données.

En écologie, il peut y avoir deux types de données soit qualitatives (binaires) correspondant à la présence/absence des espèces soit quantitatives correspondant à l'abondance des espèces. Dans notre étude de stabilité, nous avons accès à des données qualitatives. Nous avons une matrice de données où les colonnes correspondent à l'ensemble des gènes, les lignes correspondent aux différents *seeds* et les coefficients de la matrice valent 1 si le gène est présent dans le *seed* de simulation aléatoire et 0 sinon.

B.2 Indices de similarité en écologie

Nous présentons donc les indices de similarité qualitatives et asymétriques. Ces indices utilisent l'information concernant le nombre d'espèces partagées entre deux sites et le nombre d'espèces observées seulement dans le premier site ou seulement dans le second site (cf Table B.1). Dans la Table B.1, a représente le nombre d'espèces partagées par les deux sites, b, c le nombre d'espèces observé dans un des deux sites et d le nombre d'espèces présentes dans aucun des sites. Ils ne se préoccupent pas du problème des "double zéro" (on ignore la case d qui correspond au double zéro).

Dans le cas de deux sites, la région R peut être divisé en deux ensembles avec les tailles n_1 et n_2 où $n_1 = a+b$ est le nombre d'espèces observées dans le site 1 et $n_2 = a + c$ est le nombre d'espèces observées dans le site 2. On note S le nombre de sites et N le nombre d'espèces observées dans au moins un site. Si l'espèce j est dans l'aire de chevauchement des deux ensembles alors le nombre de sites dans lequel elle est observée est $s_j = 2$ et il y a a espèces et si l'espèce j n'est pas dans l'aire de chevauchement alors $s_j = 1$ et il y a soit b espèces ou

		Site 2	
		Présent	Absent
Site 1	Présent	a	b
	Absent	c	d

Table B.1 – Table schématisant l'utilisation de l'information pour les indices de similarité : a représente le nombre d'espèces présentes dans les deux sites, b, c le nombre d'espèces présentes dans un des deux sites et d le nombre d'espèces présentes dans aucun des sites (fraction d ignoré).

c espèces (cf Figure B.1). La somme du nombre de sites observés pour toutes les espèces est donc :

$$\begin{aligned}\sum_{j=1}^{N} s_j &= 2a + b + c \\ &= n_1 + n_2 = \sum_{i=1}^{S} n_i\end{aligned} \tag{B.1}$$

et est égale à la somme de la taille des ensembles combinés.

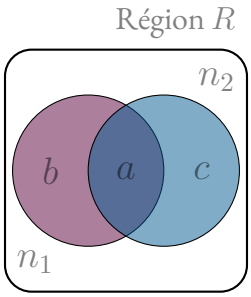


Figure B.1 – Schéma de la région R quand 2 sites sont considérés $S = 2$

L'indice de Jaccard divise le nombre d'espèces partagées par l'ensemble des deux échantillons avec le nombre total d'espèces présentes dans l'ensemble des échantillons :

$$J_2 = \frac{a}{a + b + c}. \tag{B.2}$$

L'indice de Sørensen considère que le nombre d'espèces partagées par les deux échantillons est plus important, il est donc compté deux fois :

$$S_2 = \frac{2a}{2a + b + c} = \frac{a}{\frac{1}{2}(n_1 + n_2)}. \tag{B.3}$$

Nous présentons maintenant ces deux indices dans le cas multiple. Nous rappelons que S est le nombre de sites et N est le nombre d'espèces observées dans au moins un site. On appelle E_i l'ensemble des espèces observées dans le site i, on note $|E_i| = n_i$ son cardinal. Réciproquement, on note s_j le nombre de sites dans lesquels l'espèce j est présente.

S'il n'y a que deux sites, l'indice de Sørensen est facilement interprété en terme ensembliste comme le rapport entre la taille de l'intersection des ensembles des espèces observées et la taille moyenne des ensembles. L'indice

de Sørensen est ainsi donné par :

$$S_2 = \frac{|E_1 \cap E_2|}{\frac{1}{2}\left(|E_1| + |E_2|\right)}.$$

Quand deux sites sont considérés, l'indice de Jaccard est interprété comme le rapport entre la taille de l'intersection des ensembles des espèces observées et la taille de l'union des ensembles des espèces observées :

$$J_2 = \frac{|E_1 \cap E_2|}{\left(|E_1| + |E_2| - |E_1 \cap E_2|\right)}.$$

Si une espèce appartient à $E_1 \cap E_2$, on dit qu'il appartient à un recouvrement, et s'il n'appartient pas à $E_1 \cap E_2$, il n'appartient à aucun recouvrement. Cela se généralise à un nombre de sites plus grand. Le nombre de recouvrements d'ensembles d'observation auxquels une espèce j appartient est donc tout simplement $s_j - 1$. Dans le cas optimal, la taille de ce recouvrement serait $S - 1$, et on l'appelle taux de recouvrement pour le site j : $(s_j - 1)/(S - 1)$. Finalement, la mesure de recouvrement est la somme sur toutes les espèces de ce taux de recouvrement. Dans le cas où $S = 2$, ce recouvrement est directement la taille de l'intersection $|E_1 \cap E_2| = \sum_{j=1}^{N}(s_j - 1)/(S - 1)$.

S_2 se récrit alors :

$$S_2 = \frac{\sum_{j=1}^{N}(s_j - 1)}{\frac{1}{2}\left(n_1 + n_2\right)}$$

et J_2 se réécrit :

$$J_2 = \frac{\sum_{j=1}^{N}(s_j - 1)}{N}.$$

Pour la généralisation à un nombre de sites plus grand, on divise la mesure de recouvrement par la taille moyenne sur tous les ensembles d'observation pour l'indice de Sørensen :

$$S_S = \frac{\frac{1}{S-1}\sum_{j=1}^{N}(s_j - 1)}{\frac{1}{S}\sum_{i=1}^{S} n_i}$$

et par la taille de l'union des ensembles pour l'indice de Jaccard :

$$J_S = \frac{\frac{1}{S-1}\sum_{j=1}^{N}(s_j - 1)}{N}.$$

Dans le cas où les mêmes N espèces sont observées dans tous les sites, le dénominateur (c'est à dire le recouvrement) vaut N, et c'est également la taille moyenne des ensembles, l'indice vaut donc 1. A l'inverse si dans chaque espèce n'est observée qu'une fois, $s_j = 1$ pour tout j, et l'indice vaut 0.

B.3 Indice de Jaccard

Nous présentons les résultats de l'indice de Jaccard permettant d'étudier la stabilité en sélection des méthodes de régularisation et de *Screening* étudiées dans le chapitre 2. La démarche que nous avons suivie est l'exécution des méthodes sur 100 *seeds* différentes et nous avons créé une matrice avec en ligne les différentes *seeds* et en colonne les différents gènes. Si le gène j est sélectionné pour la *seed* i, alors le coefficient (i, j) de la matrice vaudra 1 sinon il vaut 0. Nous utilisons alors l'indice de Jaccard qui permet de mesurer la similarité des gènes sélectionnés entre les différentes *seeds*.

1 Résultats concernant les méthodes de régularisation

La Table B.2 présente les indices de Jaccard obtenus pour les méthodes de régularisation.

	Lasso	Ridge	Adaptive-Lasso	Elastic-Net
Immune-Checkpoints	0.73	0.80	0.60	0.62
Sur les variables différentiellement exprimées	0.65	0.58	0.14	0.64
Sur l'ensmble des gènes	0.12	0.62	0.05	0.28

Table B.2 – Résultats de l'indice de similarité (indice de Jaccard) pour l'ensemble des méthodes de régularisation étudiées suivant l'ensemble de gènes donné en entrée. Nous précisons ici que les résultats des méthodes Ridge et Elastic-Net sont calculés à partir du top 20 des gènes.

2 Résultats concernant les méthodes de *Screening*

La Table B.3 présente les indices de Jaccard obtenus pour les méthodes de *Screening*.

	SIS	ISIS	PSIS	coxCS
Immune-Checkpoints	0.25	0.61	0.86	0.80
Sur les variables différentiellement exprimées	0.51	0.38	0.22	0.81
Sur l'ensmble des gènes	0.72	0.90	0.25	0.12

Table B.3 – Résultats de l'indice de similarité (indice de Jaccard) pour l'ensemble des méthodes de *Screening* étudiées suivant l'ensemble de gènes donné en entrée. Nous précisons ici que les résultats des méthodes Ridge et Elastic-Net sont calculés à partir du top 20 des gènes.

Annexe C

Résultats des méthodes de sélection de variables pour la survie

C.1 Méthodes de régularisation

C.1.1 Sur l'ensemble des gènes

	gènes sélectionnés			
Lasso	$ABCB5$(1%)	$ACRC$(2%)	$AMZ2$(1%)	$ANAPC7$(6%)
	$APOBEC2$(2%)	$ARHGEF4$(2%)	**B3GNTL1**(87%)	$BIN1$(1%)
(AIC =	$BIRC6$(7%)	$C10orf90$(25%)	$C14orf165$(1%)	**C19orf76**(80%)
1873.43	$C1R$(1%)	$C20orf112$(1%)	$CACNA1S$(2%)	$CAMK2N2$(1%)
±24.95)	$CAMSAP1$(1%)	$CCDC19$(25%)	$CCDC51$(1%)	$CDC7$(1%)
	CDCA3(62%)	$CENPBD1$(3%)	**CHEK2**(81%)	**CKAP4**(87%)
	$CNN2$(1%)	$CTAGE9$(7%)	**CUBN**(87%)	$CYB5D2$(1%)
	$DAG1$(2%)	$DDX24$(3%)	$DHRS12$(8%)	$EHBP1L1$(3%)
	EIF4EBP2(59%)	$EMILIN3$(1%)	$FAM133A$(7%)	$FAM63A$(3%)
	$FAM64A$(1%)	$FAM66A$(7%)	$FAM95B1$(1%)	**FBXL5**(81%)
	$FLI1$(1%)	$FUT3$(7%)	$GABBR1$(1%)	$GABPB2$(1%)
	GDF5(74%)	$GTPBP2$(1%)	$GZMA$(1%)	**HBP1**(68%)
	$HEG1$(7%)	$HPCAL1$(5%)	$HPCA$(7%)	$IGF2$(7%)
	$IKBIP$(5%)	$IKBKG$(5%)	$ILDR1$(7%)	$ITFG2$(2%)
	$ITPRIPL1$(1%)	$JAGN1$(1%)	$KIAA1109$(1%)	$KIAA1524$(1%)
	$KIF18B$(13%)	$KIF21B$(7%)	$KLHL14$(7%)	$LEO1$(2%)
	$LOC284233$(8%)	$LPIN1$(2%)	$LRRC23$(4%)	$LRRC8E$(13%)
	$LRRN4$(7%)	$MAST4$(22%)	**MBOAT7**(72%)	$MDM4$(2%)
	$MGST1$(2%)	MIA(1%)	$MKRN3$(3%)	$MOGAT2$(7%)
	$MRAP$(2%)	$NAA30$(2%)	$NDUFA8$(7%)	$NEK2$(8%)
	$NUMBL$(1%)	$OSTC$(5%)	$OTOF$(81%)	$P2RX7$(2%)
	$PABPC3$(8%)	$PDCD1$(5%)	$PDCD2$(1%)	$PHTF2$(2%)
	$PIGK$(5%)	$PIGO$(5%)	$PKD1L3$(2%)	$PRDM7$(1%)
	$PRKAA1$(1%)	$PROCA1$(2%)	$PROS1$(2%)	$PRUNE$(8%)
	PTPLA(81%)	**RANGAP1**(69%)	$RGS17$(50%)	$RGS20$(50%)

	RP9P(6%)	*RPL36AL*(8%)	*SCAP*(1%)	*SCO1*(4%)
	SEC61A2(80%)	*SERPINB8*(1%)	*SH2D4A*(6%)	*SHQ1*(1%)
	SLC12A8(6%)	*SLC45A4*(4%)	*SNCA*(6%)	**SORBS2**(81%)
	SYVN1(4%)	*TACC2*(1%)	*TAF9*(1%)	*TARP*(1%)
	THEM4(1%)	*THEMIS*(4%)	*TMEM17*(4%)	*TMEM203*(1%)
	TMEM207(1%)	*TMEM71*(4%)	*TTLL11*(6%)	*TTLL1*(1%)
	TUBGCP5(2%)	*TUSC1*(4%)	*UBE2D2*(1%)	*UMODL1*(2%)
	VWA5B1(1%)	*ZNF148*(1%)	*ZNF232*(1%)	*ZNF252*(1%)
	ZNF626(4%)	*ZNF676*(7%)	*ZNF766*(1%)	*ZNF90*(2%)
Ridge (top 20) Gènes classés par importance de l'estimateur (AIC = 1870.44 ±8.33)	*ANAPC7*(66%)	*APCDD1L*(35%)	*AR*(26%)	**C10orf90**(100%)
	CCDC19(100%)	*CCL7*(39%)	*CKAP4*(62%)	*COL7A1*(14%)
	CUBN(26%)	**DCBLD2**(100%)	**DHRS12**(100%)	*FAM72A*(29%)
	FOXF2(75%)	**GDF5**(100%)	*HJURP*(75%)	*KIF23*(75%)
	LRRC8E(27%)	*MAST4*(26%)	*MBOAT7*(2%)	**NEK2**(75%)
	NUF2(3%)	*OR7E156P*(26%)	**OTOF**(100%)	**PTPLA**(100%)
	RANGAP1(100%)	*RGS17*(81%)	**RGS20**(100%)	**SLC12A8**(100%)
	SORBS2(100%)	*TMEM132A*(46%)	*TRNP1*(76%)	*ZNF704*(26%)
Adaptive-Lasso (AIC = 1870.42 ±40.97)	*ABCA8*(1%)	*ABCF3*(1%)	*ADCY10*(1%)	*ANAPC7*(2%)
	ARHGEF4(1%)	*ATP2A3*(1%)	**B3GNTL1**(82%)	*BIRC6*(2%)
	BMP8B(2%)	*BRSK1*(1%)	*C10orf12*(1%)	*C10orf90*(21%)
	C12orf40(3%)	*C19orf76*(6%)	*C21orf45*(1%)	*C2CD4C*(2%)
	C4orf39(1%)	*C5orf32*(1%)	*CAMSAP1*(2%)	*CASKIN1*(1%)
	CBWD2(1%)	*CCDC14*(1%)	*CCDC19*(19%)	*CCNG1*(1%)
	CENPBD1(5%)	*CHCHD10*(1%)	**CKAP4**(70%)	*CLNK*(2%)
	COL6A2(1%)	**CUBN**(82%)	*CUEDC1*(2%)	*DACH1*(1%)
	DAG1(1%)	*DBF4B*(2%)	*DDX24*(1%)	*DHRS12*(1%)
	DQX1(1%)	*DUSP2*(1%)	*EHBP1L1*(1%)	*EIF4EBP2*(1%)
	ELFN2(1%)	*ENGASE*(1%)	*ENTPD6*(1%)	*FAM133A*(4%)
	FAM63A(1%)	*FAM66A*(3%)	*FAM7A2*(1%)	*FAM95B1*(1%)
	FBXL5(48%)	*FBXO39*(2%)	*FCGR1C*(1%)	*GABPB2*(1%)
	GAS2L3(1%)	**GDF5**(74%)	*GFM2*(1%)	*GJB1*(1%)
	GNAI2(1%)	*GSTO1*(2%)	*GZMA*(1%)	*HBP1*(27%)
	HDAC7(1%)	*HDGF*(1%)	*HEG1*(1%)	*HIST1H2BK*(2%)
	IKBKG(1%)	*ILDR1*(3%)	*JAGN1*(1%)	*JPH3*(1%)
	KIF21B(4%)	*KIF22*(2%)	*KLHL14*(1%)	*LAD1*(1%)
	LAMB3(2%)	*LCA5L*(1%)	*LILRB2*(1%)	*LOC284233*(2%)
	LOC646471(1%)	*LPIN1*(1%)	*LRRC8E*(4%)	*LRRN4*(2%)
	LST1(1%)	*MAST4*(1%)	*MDM4*(2%)	*MED20*(1%)
	MFSD6(1%)	*MIOX*(2%)	*MKI67IP*(1%)	*MME*(1%)
	MOGAT2(2%)	*MOXD1*(2%)	*MRGPRX3*(2%)	*NBPF3*(1%)
	NCRNA00081(1%)	*NDUFA8*(2%)	*NDUFS6*(2%)	*NFAM1*(1%)
	NKRF(2%)	*NXT1*(2%)	*OR9Q1*(1%)	*OSTC*(1%)
	OTOF(79%)	*OXCT1*(1%)	*PARP1*(1%)	*PKD1L3*(1%)
	POLD2(1%)	*POLI*(2%)	*PROS1*(2%)	*PROSC*(1%)

	$PRR12(1\%)$	$PTGER3(1\%)$	$PTPLA(27\%)$	$RAB40AL(5\%)$
	$RANGAP1(2\%)$	$RGS20(24\%)$	$RNF216L(1\%)$	$RPL36AL(3\%)$
	$SCO1(3\%)$	$SEC13(1\%)$	**SEC61A2**(26%)	$SERPINA7(1\%)$
	$SERPINB8(1\%)$	$SETD5(2\%)$	$SFTA1P(1\%)$	$SH2D4A(1\%)$
	$SHQ1(1\%)$	$SKA2(1\%)$	$SLCO1A2(1\%)$	$SLITRK3(1\%)$
	$SMARCA2(1\%)$	$SNCA(1\%)$	$SNORA39(1\%)$	**SORBS2**(78%)
	$SPIC(2\%)$	$STX1A(1\%)$	$SYNC(1\%)$	$TAF1L(1\%)$
	$TAF5L(2\%)$	$TAF9(1\%)$	$TBC1D15(1\%)$	$TMCC1(1\%)$
	$TMEM31(1\%)$	$TMEM71(1\%)$	$TNFSF12(2\%)$	$TRAF7(1\%)$
	$TUSC1(3\%)$	$UAP1(1\%)$	$UBE2R2(1\%)$	$UMODL1(1\%)$
	$USP42(2\%)$	$WNT1(4\%)$	$ZAP70(1\%)$	$ZGPAT(2\%)$
	$ZNF252(1\%)$	$ZNF283(2\%)$	$ZNF766(2\%)$	$ZNF833(1\%)$
	$ZNF90(1\%)$			
Elastic-Net (top 20) Gènes classés par importance de l'estimateur (AIC = 1874.36 ±5.60)	$ANAPC7(2\%)$	$AR(17\%)$	$B3GNTL1(50\%)$	$BIRC6(1\%)$
	$C10orf90(52\%)$	$C19orf40(48\%)$	$C19orf76(48\%)$	$CCDC19(52\%)$
	$CDCA3(33\%)$	**CHEK2**(48%)	**CKAP4**(100%)	$COPS7B(15\%)$
	$CTAGE9(1\%)$	**CUBN**(50%)	$DCBLD2(50\%)$	$DHRS12(51\%)$
	$EIF4EBP2(47\%)$	$FAM133A(1\%)$	$FAM66A(1\%)$	$FAM72A(50\%)$
	$FBXL5(49\%)$	$FOXF2(50\%)$	$FST(1\%)$	$FUT3(1\%)$
	GDF5(99%)	$HEG1(1\%)$	$HES2(1\%)$	$HJURP(50\%)$
	$HPCAL1(1\%)$	$HPCA(1\%)$	$IGF2(1\%)$	$ILDR1(1\%)$
	$KIF18B(33\%)$	$KIF21B(1\%)$	$KIF23(50\%)$	$KLHL14(1\%)$
	$LRRC23(1\%)$	$LRRC8E(2\%)$	$LRRN4(1\%)$	$MAST4(2\%)$
	MBOAT7(97%)	$ME2(1\%)$	$MOGAT2(1\%)$	$NDUFA8(1\%)$
	$NEK2(50\%)$	**OTOF**(99%)	$PABPC3(1\%)$	$PAG1(1\%)$
	$PDCD1(1\%)$	$PRUNE(1\%)$	**PTPLA**(99%)	$PTPRB(47\%)$
	RANGAP1(99%)	$RGS17(50\%)$	$RGS20(52\%)$	$RP9P(1\%)$
	$RPL36AL(1\%)$	$SCRG1(1\%)$	$SEC61A2(49\%)$	$SH2D4A(1\%)$
	$SLC12A8(52\%)$	$SNCA(1\%)$	**SORBS2**(99%)	$STRADA(49\%)$
	$TRNP1(50\%)$	$TUBA3C(1\%)$	$TUBGCP5(1\%)$	$ZNF676(1\%)$
	$ZNF704(49\%)$			

Table C.1 – Résultats des méthodes de régularisation sur l'ensemble de gènes

C.2 Méthodes de *Screening*

C.2.1 Sur l'ensemble des gènes

	gènes sélectionnés			
SIS (AIC = 1873.80± 0.71)	**C5orf23**(100%)	**CAD**(83%)	**CHEK2**(100%)	**CUBN**(100%)
	$DHRS12(27\%)$	**GDF5**(100%)	$SPC24(1\%)$	$TOP2A(69\%)$
ISIS (AIC = 1880.01	$APOBEC2(1\%)$	**C5orf23** (94%)	**CAD**(88%)	**CHEK2**(94%)
	$COMMD5(1\%)$	$CTSO(1\%)$	**CUBN**(94%)	$DHRS12(45\%)$
	$DUOX1(1\%)$	$EHBP1L1(1\%)$	$FAM155B(1\%)$	**GDF5** (93%)

±24.69)	$GNA11$(2%)	$GRM7$(2%)	$IGF2$(2%)	$KIF21B$(2%)
	$LPIN1$(1%)	$NDUFA8$(2%)	$NKD1$(2%)	$NUP153$(2%)
	$PHTF2$(2%)	$PROS1$(2%)	$SNCA$(2%)	$SPAG5$(2%)
	$STK40$(2%)	$TMEM17$(2%)	$TUSC1$(2%)	$ZGPAT$(1%)
PSIS	**ABTB2**(70%)	$ACER2$(2%)	$ACSBG1$(1%)	**ADAM17**(97%)
(AIC =	**ADCK1**(85%)	$ADCY7$(1%)	**AFAP1L2**(96%)	$AGPAT5$(29%)
1931.38	$AHNAK$(2%)	$ALG12$(85%)	$ALKBH6$(10%)	$ALPK2$(22%)
±12.55)	AMH(24%)	$AMIGO2$(29%)	$ANKH$(29%)	$ANKRD12$(13%)
	ANKRD22(77%)	**AP2A1**(99%)	$AP3B2$(15%)	$APH1B$(19%)
	APOL4(78%)	$ARHGEF6$(14%)	$ASB12$(16%)	$ASF1B$(29%)
	ASNA1(84%)	$ATG9A$(29%)	$ATP10B$(43%)	**ATP2C2**(97%)
	$ATP8B3$(8%)	AUH(43%)	**AXIN2**(96%)	$BAGE2$(19%)
	$BCL2L10$(1%)	$BCL9$(2%)	BCR(1%)	$BIN1$(2%)
	$BMP8A$(22%)	**BTN3A1**(78%)	$C10orf11$(22%)	**C11orf41**(82%)
	C12orf43(85%)	$C12orf52$(20%)	$C12orf62$(15%)	$C14orf129$(56%)
	$C14orf37$(2%)	$C14orf79$(19%)	$C16orf93$(8%)	$C17orf44$(56%)
	$C17orf46$(1%)	$C18orf8$(1%)	$C19orf36$(1%)	**C19orf55**(99%)
	C22orf26(84%)	**C2orf62**(97%)	$C3orf47$(76%)	**C3orf50**(95%)
	$C4orf44$(1%)	**C6orf163**(98%)	$C6orf59$(1%)	$C7orf55$(70%)
	$C7orf59$(1%)	$C8orf84$(15%)	$C9orf167$(13%)	**C9orf93**(99%)
	$CAPRIN2$(13%)	$CCDC102A$(1%)	$CCDC160$(8%)	$CCDC3$(70%)
	$CCL13$(29%)	$LILRA6$(1%)	$LMAN2$(1%)	$LOC100130691$(1%)
	$LOC284232$(1%)	$LOC441177$(1%)	$LOC613037$(1%)	$LOC81691$(1%)
	MAK(1%)	$MAPKAPK5$(1%)	$MCM3AP$(1%)	$MRPS10$(1%)
	$MRPS18C$(1%)	$MYT1$(1%)	$NDUFAF3$(1%)	$NEK2$(1%)
	$NFKBIE$(1%)	$NFKBIL1$(1%)	NNT(1%)	$NTN1$(1%)
	$RAB34$(1%)	$RAP1GAP2$(1%)	RFK(1%)	$RGS2$(1%)
	$RHEBL1$(1%)	$RNF121$(1%)	$RPL36$(1%)	$RSPO4$(1%)
	$SALL4$(1%)	$SCGN$(1%)	$SLA2$(1%)	$SLC41A1$(1%)
	$SNF8$(1%)	$SNRPA1$(1%)	$SOAT2$(1%)	$SOBP$(1%)
	$SPHK2$(1%)	$STAG3$(1%)		
coxCS	**A1CF**(91%)	**A2BP1**(91%)	**A2ML1**(91%)	**A2M**(91%)
(AIC =	**A4GALT**(91%)	**A4GNT**(91%)	**AAAS**(91%)	**AACSL**(89%)
1937.71	**AACS**(91%)	**AADAC**(89%)	**AADAT**(89%)	**AAGAB**(89%)
	AAK1(90%)	$AAMP$(73%)	**AARS2**(91%)	**AARSD1**(89%)
±13.69)	$AASDH$(28%)	**AASS**(91%)	**AATF**(91%)	**AATK**(86%)
	ABCA10(91%)	**ABCA11P**(89%)	$ABCA12$(27%)	$ABCA13$(23%)
	$ABCA1$(50%)	**ABCA2**(91%)	**ABCA3**(89%)	**ABCA4**(91%)
	ABCA5(91%)	$CCDC109B$(1%)	$CCDC110$(1%)	$CCDC112$(1%)
	$CCDC113$(1%)	$CCDC114$(1%)	$CCDC115$(1%)	$CCDC116$(1%)
	$CCDC117$(1%)	$CCDC11$(1%)	$CCDC120$(1%)	$CCDC121$(1%)
	$CCDC122$(1%)	$CCDC123$(1%)	$CCDC125$(1%)	$CCDC126$(1%)
	$CCDC12$(1%)	$CCDC130$(1%)	$CCDC132$(1%)	$CCDC134$(1%)
	$CCDC135$(1%)	$CCDC137$(1%)	$CCDC138$(1%)	$CCDC13$(1%)
	$CCDC144A$(1%)	$CCDC144B$(1%)	$CCDC144C$(1%)	$CCDC146$(1%)
	$CCDC147$(1%)	$CYP4A22$(2%)	$CYP4B1$(2%)	$CYP4F12$(2%)

	$CYP4F22(2\%)$	$CYP4F2(2\%)$	$CYP4F3(2\%)$	$CYP4V2(2\%)$
	$CYP4X1(2\%)$	$CYP4Z2P(2\%)$	$CYP51A1(2\%)$	$CYP7A1(2\%)$
	$CYP7B1(2\%)$	$CYP8B1(1\%)$	$CYR61(1\%)$	$CYS1(2\%)$
	$CYSLTR1(2\%)$	$CYTH2(2\%)$	$CYTH3(2\%)$	$CYTH4(1\%)$
	$CYTIP(1\%)$	$CYTSA(1\%)$	$CYTSB(1\%)$	$CYYR1(1\%)$
	$D2HGDH(1\%)$	$D4S234E(1\%)$	$DAAM1(1\%)$	$DAAM2(1\%)$
	$DAB1(1\%)$	$HAVCR1(1\%)$	$HAVCR2(1\%)$	$HBA1(1\%)$
	$HBA2(1\%)$	$HBB(1\%)$	$HBD(1\%)$	$HBE1(1\%)$
	$HBEGF(1\%)$	$HBG1(1\%)$	$HBG2(1\%)$	$HBP1(1\%)$
	$HBS1L(1\%)$	$HBXIP(1\%)$	$HCFC1R1(1\%)$	$HCFC1(1\%)$
	$HCG18(1\%)$	$HCG22(1\%)$	$HCG26(1\%)$	$HCG27(1\%)$
	$HCG4P6(1\%)$	$HCG4(1\%)$	$HCG9(1\%)$	$HCN2(1\%)$
	$HCN3(1\%)$	$HCN4(1\%)$	$HCP5(1\%)$	$HCST(1\%)$
	$KNTC1(1\%)$	$KPNA1(1\%)$	$KPNA3(1\%)$	$KPNA4(1\%)$
	$KPNA5(1\%)$	$KPNA6(1\%)$	$KPNB1(1\%)$	$KPTN(1\%)$
	$KRAS(1\%)$	$KRBA1(1\%)$	$KRBA2(1\%)$	$KRCC1(1\%)$
	$KREMEN1(1\%)$	$KRI1(1\%)$	$KRIT1(1\%)$	$KRT13(1\%)$
	$KRT14(1\%)$	$KRT15(1\%)$	$KRT16(1\%)$	$KRT18(2\%)$
	$KRT19(2\%)$	$KRT1(2\%)$	$KRT23(2\%)$	$KRT24(2\%)$
	$KRT25(2\%)$	$KRT2(2\%)$	$KRT34(2\%)$	$PSMD8(2\%)$
	$PSMD9(2\%)$	$PSME2(2\%)$	$PSME3(2\%)$	$PSME4(2\%)$
	$PSMF1(1\%)$	$PSMG1(2\%)$	$PSMG2(2\%)$	$PSMG3(1\%)$
	$PSMG4(2\%)$	$PSORS1C1(2\%)$	$PSORS1C2(2\%)$	$PSORS1C3(2\%)$
	$PSPC1(2\%)$	$PSPH(1\%)$	$PSPN(1\%)$	$PSTPIP1(2\%)$
	$PSTPIP2(2\%)$	$PTAFR(2\%)$	$PTAR1(2\%)$	$PTBP2(2\%)$
	$PTCD1(2\%)$	$PTCD2(2\%)$	$PTCH2(2\%)$	$PTCHD1(2\%)$
	$PTCHD2(2\%)$	$PTCRA(2\%)$	$PTDSS1(2\%)$	$UBR5(2\%)$
	$UBR7(2\%)$	$UBTD2(2\%)$	$UBTFL1(2\%)$	$UBTF(2\%)$
	$UBXN10(2\%)$	$UBXN11(2\%)$	$UBXN1(2\%)$	$UBXN2A(2\%)$
	$UBXN2B(2\%)$	$UBXN4(2\%)$	$UBXN6(2\%)$	$UBXN7(2\%)$
	$UBXN8(2\%)$	$UCA1(2\%)$	$UCHL1(2\%)$	$UCK1(2\%)$
	$UCK2(2\%)$	$UCKL1AS(2\%)$	$UCKL1(2\%)$	$UCN3(2\%)$
	$UCN(2\%)$	$UCP2(2\%)$	$UCP3(2\%)$	$UFC1(2\%)$
	$UFD1L(2\%)$	$UFM1(2\%)$	$UFSP1(2\%)$	$UFSP2(2\%)$

Table C.2 – Résultats des méthodes de *Screening* sur l'ensemble de gènes

Annexe D

Présentation de l'outil Gimli

Gimli (Campos et al., 2013) est un outil *open-source* pour la reconnaissance automatique des termes biomédicaux dans un texte. Cet outil utilise de nombreuses ressources et outils disponibles publiquement. Pour le pré-traitement, GDep (Tsuruoka et al., 2005) est utilisé et notamment pour le traitement linguistique et le traitement des termes clés. Pour obtenir les ressources lexicales, deux dictionnaires sont utilisées comme BioThesaurus (Liu et al., 2006) pour le nom des gènes et des protéines et BioLexicon pour les termes des domaines biomédicaux. L'architecture de cet outils est composé de trois grandes parties. La première partie consiste à la collecte des *corpus*. Cette collecte est nécessaire pour effectuer l'apprentissage et l'évaluation de l'outil. De nombreux *corpus* publiques sont disponibles, mais Gimli utilise seulement deux d'entre eux : GENETAG (Tanabe et al., 2005) et JNLPBA (Kim et al., 2004). GENETAG (Tanabe et al., 2005) est composé de 20 000 phrases extraites d'*abstracts* provenant de MEDLINE (base de données publiques de la littérature biomédicale accessible sur `https://www.nlm.nih.gov/bsd/pmresources.html`) avec des données hétérogènes (protéines, DNA, RNA) et annotées par des experts en biochimie, en génétique et en biologie moléculaire. JNLPBA (Kim et al., 2003) est constitué de 2 404 *abstracts* extraits de MEDLINE en utilisant les termes MeSH (pour *Medical Subject Headings*) suivants `"human"`, `"blood cell"`, `"transcription factor"`. MeSH est un vocabulaire hierarchierement organisé et contrôlé, il est produit par la *National Library of Medicine*. L'annotation manuelle de ces *abstracts* a été basée sur les 5 classes de l'ontologie GENIA (Kim et al., 2003). La deuxième partie de cet outil concerne l'ensemble des variables. L'outil fournit un large ensemble de variables incluant des variables orthographiques (capturant la connaissance sur la formation du mot), des variables morphologiques (permettant d'identifier des similarités entre différents termes), des variables linguistiques (permettant d'ajouter des informations sur les relations entre les termes d'une phrase) et des variables de domaine biologique. Les variables de domaines biologiques sont récupérées à partir des dictionnaires (GENETAG et JNLPBA) cités ci-dessus. La troisième partie de l'outil concerne l'apprentissage du modèle et celui-ci est effectué à partir de l'outil MALLET (McCallum, 2002). Cette apprentissage s'effectue sur les variables extraites des deux corpus en utilisant une méthode de *machine learning* supervisé. La méthode appliquée est le champ aléatoire conditionnel introduit par Lafferty et al. (2001), elle permet de labeliser des données comme par exemple du texte.

Soit X une variable aléatoire à labeliser. Dans le cadre de notre jeu de données, cela correspond aux *abstracts*. Soit Y une variable aléatoire des labels et appartient à un alphabet fini. Y peut être vu comme un ensemble de n labels associés aux observations X, *i.e.* $X = \{X_1, \ldots, X_n\}$ et $Y = \{Y_1, \ldots, Y_n\}$. Par exemple, X_1 = "le traitement CTLA4/PD1 est utilisé chez les patients atteints du cancer du rein à cellules claires" et $Y_1 = CTLA4/PD1, CTLA4, PD1, cancer, rein, ccRCC$. L est l'ensemble des labels possibles et O est l'ensemble des observations (lexique discret car nos observations correspondent à du texte).

Un champ aléatoire conditionnel est défini par la probabilité conditionnelle :

$$p(Y|X) = \frac{1}{Z(X)} \exp\left[\sum_{i=1}^{n}\sum_{k}^{K} \lambda_k f_k(y_{i-1}, y_i, x, i)\right], \tag{D.1}$$

avec λ_k un paramètre à estimer à partir des données d'entraînement indiquant le degré d'importance de la variable. Ce modèle est donc défini à partir de la probabilité qu'une séquence de labels particulière soit associée à la séquence d'observations. $Z(X)$ de l'équation (D.1) est un facteur de normalisation et chaque fonction $f_k(y_{i-1}, y_i, x, t)$ est soit une fonction de transition $t(y_{i-1}, y_i, x, t)$ soit une fonction d'état $S(y_{i-1}, y_i, x, t)$. L'entraînement d'un champ aléatoire conditionnel, qui signifie estimer le paramètre λ_k, consiste à maximiser la log-vraisemblance sur un ensemble de n couples (séquence d'observations, séquence de labels) :

$$\arg\max_{\lambda_k} \left\{\sum_{i=1}^{n} \log(p(y_i|x_i, \lambda_k)\right\}. \tag{D.2}$$

Estimer la valeur de λ_k est un problème d'optimisation convexe. Cependant, quand des modèles de plus grand ordre sont considérés la complexité d'entraînement augmente de manière exponentielle.

Dans le but de corriger certaines erreurs générées par les modèles de champs aléatoires conditionnels, Gimli possède une étape de post-traitement dans cette troisième partie de l'outil. L'implémentation de cette étape permet la correction des parenthèses et la résolution d'abréviations dans le but de compléter des noms mal séparés. Une fois le modèle entraîné, le jeu de test peut être annoté. Notre jeu de test correspond à l'ensemble des *abstracts* fournis à l'outil. L'outil nous permet d'obtenir finalement des annotations à partir de l'ensemble des *abstracts* fournis.

Annexe E

Vérifier l'hypothèse des risques proportionnels à partir des résidus de Schoenfeld

E.1 Introduction

J'ai présenté en Section 1 du chapitre 3 l'idée des résidus de Schoenfeld ainsi que l'utilisation du test des risques proportionnels proposé par Grambsch et al. (1994). Dans cette annexe, nous commençons par détailler la définition des résidus introduite par Schoenfeld (1980) et Schoenfeld (1982) et l'histoire autour du test des risques proportionnels. De nombreux travaux ont été réalisés pour le test des risques proportionnels qui se différencient par le choix de la fonction de transformation du temps dans les différents tests (Wei, 1984 ; Gill et al., 1987 ; O'Quigley et al., 1989). Le choix de la transformation du temps peut amener à une mauvaise spécification de modèle. Par exemple, Harrell et al. (1985) a d'abord suggéré de calculer la corrélation entre les résidus de Schoenfeld pour chaque covariable et le rang du temps de survie pour diagnostiquer la présence des risques non proportionnels. Grambsch et al. (1994) a évoqué le problème de la transformation du temps et a proposé un test global et un test pour chaque variables des résidus standardisés de Schoenfeld. Nous détaillons le test proposé par Grambsch et al. (1994). Nous présentons aussi les limites de ces tfests évoqués par Therneau et al. (2000). Enfin, nous présentons les résultats du test des risques proportionnels sur chacune de nos simulations.

E.2 Définition des résidus (Schoenfeld, 1982 ; Schoenfeld, 1980)

Calcul des résidus de Schoenfeld

Supposons que nous avons n individus et pour chaque individu, nous avons un vecteur de p variables $X_{i.} = (X_{i1}, \ldots, X_{ip})^T$. Le modèle à risques proportionnels est défini par la fonction de risque :

$$\lambda_i(t) = \lambda_0(t) \exp\left(\beta^T X_{i.}\right), \tag{E.1}$$

où β est un vecteur de p paramètres et $\lambda_0(t)$ est la fonction de risque de base. Soit D les indices des individus décédés et R_i les individus à risque au temps t_i (quand le i-ème individu est décédé). L'estimation des β est effectuée à partir de la vraisemblance partielle de Cox (Cox, 1975) définie de la manière suivante :

$$\frac{\exp(\beta^T X_{m.})}{\sum_{k \in R_i} \exp\left(\beta^T X_{k.}\right)}. \tag{E.2}$$

Dans ce modèle, X_{ij} est une variable aléatoire avec :

$$\mathbb{E}[X_{ij}|R_i] = \frac{\sum_{k\in R_i} X_{kj}\exp(\beta^T X_{k.})}{\sum_{k\in R_i}\exp(\beta^T X_{k.})} \tag{E.3}$$

et l'estimateur de β par maximum de vraisemblance est solution de :

$$\sum_{i\in D}[X_{ij} - \mathbb{E}(X_{ij}|R_i)] = 0 \tag{E.4}$$

Notons cette solution par $\hat{\beta}$ et $\hat{\mathbb{E}}(X_{ij}|R_i)$ est $\mathbb{E}[X_{ij}|R_i]$ avec β remplacé par $\hat{\beta}$. Le résidu partiel au temps d'observation de l'individu i t_i est le vecteur $\hat{r}_i = (\hat{r}_{i1}, \ldots, \hat{r}_{ip})^T$:

$$\hat{r}_{ik} = X_{ik} - \hat{\mathbb{E}}[X_{ik}|R_i].$$

Le résidu est la différence entre la valeur observée de $X_{i.}$ et son espérance conditionnelle sachant R_i.

Examen de l'hypothèse des risques proportionnels

Si les risques proportionnels sont vérifiés, nous avons $\mathbb{E}(\hat{r}_i) \simeq 0$ et obtenons un graphe de $\hat{r}_{ik}$ en fonction de t_i centré autour de 0. Supposons que :

$$\lambda_i(t) = \lambda_0(t)\exp((\beta + \theta g(t_i))X_{ik}), \tag{E.5}$$

avec $g(t_i)$ variant autour de 0. En développant $\mathbb{E}(X_{ij}|R_i)$ autour de $g(t_i)$, on obtient :

$$\mathbb{E}(\hat{r}_{ik}) = g(t_i)\left[\mathbb{E}(X_{ij}^2|R_i) - \mathbb{E}(X_{ij}|R_i)^2\right] \tag{E.6}$$

Ainsi, à partir de (E.6) un changement de $g(t_i)$ sera observé sur le graphe des résidus en fonction des temps de décès.

E.3 Définition et amélioration par Grambsch et al. (1994)

Définition

Utilisant l'approche des processus ponctuels (Fleming et al., 2005), Grambsch et al. (1994) considèrent chaque individu comme étant un processus de comptage indépendant $N_i(t), t \geq 0, i = 1, \ldots, n$ avec la fonction d'intensité donnée par :

$$Y_i(t)\exp(\beta^T X_{i.})d\Lambda_0(t), \tag{E.7}$$

où $Y_i(t)$ indique si le i-ème individu est à risque au temps t, β est le vecteur des paramètres de régression et $X_{i.}$ est un p-vecteur des variables pour chaque individu et $d\Lambda_0(t)$ est la fonction de risque de base. β peut être estimé en maximisant la log-vraisemblance partielle de Cox (Cox, 1975) :

$$\sum_{i=1}^{n}\int_0^{\infty}\left[Y_i(t)\exp(\beta^T X_{i.}) - \log\left\{\sum_{j=1}^{n} Y_j(t)\exp(\beta^T X_j(t))\right\}\right]dN_i(t) \tag{E.8}$$

Grambsch et al. notent :

$$S^{(r)}(\beta, t) = \sum_{i=1}^{n} Y_i(t) \exp(\beta^T X_{i.}) X_{i.}^{\otimes r}, \tag{E.9}$$

pour $r = 0, 1, 2$ où pour un vecteur colonne a, $a^{\otimes 2}$ correspond au produit aa^T, $a^{\otimes 1}$ est le vecteur a et $a^{\otimes 0}$ est le scalaire 1. La moyenne et la variance conditionnelle au temps t est :

$$M(\beta, t) = \frac{S^{(1)}(\beta, t)}{S^{(0)}(\beta, t)} \tag{E.10}$$

$$V(\beta, t) = \frac{S^{(2)}(\beta, t)}{S^{(0)}(\beta, t)} - \left\{ \frac{S^{(1)}(\beta, t)}{S^{(0)}(\beta, t)} \right\}^{\otimes 2}. \tag{E.11}$$

Les résidus de Schoenfeld sont alors définis :

$$r_i(\beta) = X_{(i)} - M(\beta, t_i), \tag{E.12}$$

avec $X_{(i)}$ le vecteur des covariables pour l'individu dont le temps d'observation est t_i.

Approche pour le test de non-proportionnalité

Supposons qu'on souhaite tester l'hypothèse nulle des risques proportionnels comme dans (E.7) *versus* l'alternative des coefficients dépendants du temps avec la fonction d'intensité donnée par :

$$\lambda_i(t)dt = Y_i(t) \exp(\beta^T(t) X_{i.}) d\Lambda_0(t) \tag{E.13}$$

$$= Y_i(t) \exp(\{\beta + G(t)\theta\}^T X_{i.}) d\Lambda_0(t) \tag{E.14}$$

L'approche de Grambsch et al. généralise celle de Schoenfeld (1982) qui considérait la non-proportionnalité pour une seule variable. Supposons (E.14) et β connu :

$$r_i(\beta) = X_{(i)} - M(\beta, t_i) \tag{E.15}$$

$$= \left[X_{(i)} - M(\beta(t_i), t_i)\right] + \left[M(\beta(t_i), t_i) - M(\beta, t_i)\right] \tag{E.16}$$

En développant $M(\beta(t_i), t_i)$ dans un développement de Taylor autour de $\beta(t_i) = \beta$ dans la seconde somme de l'équation (E.16), Grambsch et al. obtiennent :

$$\mathbb{E}[r_i(\beta)|F_{t_i}] \simeq V(\beta, t_i) G_i \theta, \tag{E.17}$$

où $G_i = G(t_i)$ est une matrice diagonale dont les coordonnées (j, j) sont égales à $g_j(t_i) - \bar{g}_j$. Soit $r_i^\star = r_i^\star(\beta) = V^{-1}(\beta, t_i) r_i(\beta)$ les résidus standardisés de Schoenfeld, Grambsch et al. ont montré :

$$\mathbb{E}(r_i^\star | F_{t_i}) \simeq G_i \theta, \tag{E.18}$$

$$\mathbb{V}(r_i^\star | F_{t_i}) = V^{-1}(\beta, t_i) V(\beta(t_i), t_i) V^{-1}(\beta, t_i) \simeq V^{-1}(\beta, t_i) \tag{E.19}$$

De plus, Grambsch et al. remarquent que les équations (E.18) et (E.19) suggèrent un modèle linéaire pour $r_i^\star$. La méthode des moindres carrés généralisés avec $V_i \equiv V(\beta, t_i)$ donne alors :

$$\hat{\theta} = (\sum G_i V_i G_i)^{-1} \sum G_i r_i. \tag{E.20}$$

Cela mène à un test du χ^2 à p degrés de liberté avec une statistique de test définie par :

$$(\sum G_i ri)^T (\sum G_i V_i G_i)^{-1} (\sum G_i r_i) \tag{E.21}$$

dont l'hypothèse nulle est $H_0 : \theta = 0$. Supposons que β est inconnu et $\hat{\beta}$ est l'estimateur du maximum de la vraisemblance partielle sous H_0, $\hat{V}_i = V(\hat{\beta}, t_i)$ et $\hat{r}_i = r_i(\hat{\beta})$. Grambsch et al. montrent qu'on a comme ci-dessus : $\mathbb{E}(\hat{V}_i^{-1}\hat{r}_i) \simeq G_i\theta$. Cependant, la somme des résidus est égale à zéro. Cela implique que les résidus sont corrélés et $cov(\hat{r}_l, \hat{r}_m)$ est estimé par $\delta_{lm}V_l - V_l(\sum V_i)^{-1}V_m$ sous H_0 (Schoenfeld, 1982). Ainsi, $\mathbb{V}(\hat{V}_i^{-1}\hat{r}_i) \simeq \hat{V}_i^{-1} - (\sum_l \hat{V}_l)^{-1}$. La méthode des moindres carrés donne alors :

$$\hat{\theta} = F^{-1} \sum G_i r_i, \tag{E.22}$$

avec

$$F = \sum G_i \hat{V}_i G_i^T - (\sum G_i \hat{V}_i)(\sum \hat{V}_i)^{-1}(\sum G_i \hat{V}_i)^T. \tag{E.23}$$

Sous H_0 la variance de $n^{-1/2} \sum G_i \hat{r}_i$ peut être estimé par $n^{-1}F$ menant à un test du χ^2 à p degrés de liberté dont la statistique de test est :

$$T(G) = (\sum G_i \hat{r}_i)^T F^{-1} \sum G_i \hat{r}_i. \tag{E.24}$$

E.4 Limites

Différents tests

Différents choix pour $G(t)$ peuvent donner dans les différents tests une mauvaise définition du modèle. Plusieurs tests ont été développés et chacun de ces tests diffèrent par le choix de $g(t)$. Pour le premier test, $g(t)$ est une fonction du temps ce qui implique que $T(G)$ est un score de test pour l'addition de la variable dépendante du temps au modèle. Ce test est initialement proposé par Cox (1972) et le test proposé par Gill et al. (1987) est en lien avec celui-ci. Pour le deuxième test, $g(t)$ est une constante par morceaux sur des intervalles de temps. Les intervalles et les constantes sont choisis préalablement. Dans cette situation, le score du test est proposé par O'Quigley et al. (1989). L'inconvénient de ce test est le choix des constantes et des intervalles de temps, mais une ligne de conduite est suggérée. Dans le troisième test, il est supposé que $G(t)$ est $g(t) = N(t-)$ alors le score de test est basé sur un graphe des résidus *versus* le rang des temps d'évènement. Cela est équivalent au test de Breslow (Breslow et al., 1984) qui utilise un score de rang pour une variable dépendante du temps dans le modèle de Cox, et également similaire à celui proposé par Harrell (1982) utilisant la corrélation entre les résidus non standardisés et le rang de l'événement. Pour le quatrième test, Lin (1991) suggère de comparer $\hat{\beta}$ à la solution de $\hat{\beta}_g$ estimée à partir de l'équation de Cox pondérée :

$$\sum_i G_i r_i(\beta) = 0,$$

avec $g(t)$ est une des fonctions pondérées communément choisie pour les tests du log-rank pondéré. Si les estimateurs de $\hat{\beta}_g$ étaient basés sur un algorithme de Newton-Raphson commençant $\hat{\beta}$, le test sera identique à T. Chacun de ces tests peut être directement visualisé comme une simple test de tendance appliqué au graphe des résidus standardisés *versus* $g(t)$.

Diagnostiques

Les statistiques de test impliquent une forme pré-définie de la proportionnalité au départ donné par la fonction $g(t)$. Mais, il est possible de n'avoir aucune hypothèse sur la nature de la non-proportionnalité. Dans ce cas, le mieux est de laisser parler les données. Schoenfeld (1982) et Lin (1991) recommandaient de tracer les résidus en fonction des temps de décès et Wei (1984) et Therneau et al. (1990) recommandaient de tracer les sommes cumulées. À partir des équations (E.14) et (E.18), Grambsch et al. suggèrent de tracer les points de $\hat{V}_k^{-1}\hat{r}_k + \hat{\beta}$ *versus* t_k qui révéleront la forme de $\beta(t)$. En réalité, $\hat{V}_k$ peut être instable, et surtout près de la fin de l'étude quand le nombre de sujets à risque est plus faible que le nombre de lignes $\hat{V}_k$. Pour la plupart des jeux de données, $V(\hat{\beta}, t)$ change légèrement et est plutôt stable jusqu'aux derniers temps de décès. Elle peut être substituée par la valeur moyenne $\bar{V} = \mathcal{I}/d$, où $\mathcal{I}^{-1}$ est la matrice de covariance de $\hat{\beta}$. Enfin, Keele (2010) met en garde sur la bonne utilisation du test de non-proportionnalité. En effet, ce test permet de détecter un certain nombre d'erreurs de spécification en plus de la non-proportionnalité. C'est-à-dire qu'un résultat significatif pour le test de non-proportionnalité peut indiquer plusieurs défaillances du modèle. Une interprétation correcte pour un résultat significatif ne signifie pas forcément que les risques ne sont pas proportionnels, mais que si la spécification du risque est correcte alors des preuves existent pour dire que les risques ne sont pas proportionnels. La question importante est de savoir quels erreurs de spécification peuvent amener à une mauvaise interprétation du résultat significatif de non-proportionnalité. La première erreur évoquée par Keele (2010) est l'omission d'une variable importante, cela peut consister à l'oubli d'une variable ou l'oubli d'une interaction importante entre variables. La deuxième erreur provoquant un résultat positif du test de non-proportionnalité est la considération d'une variable comme linéaire alors que son effet ne l'est pas. Enfin, utiliser un modèle à risques proportionnels quand un modèle de survie différent est plus approprié peut également conduire à un résultat de test de non-proportionnalité significatif. Keele (2010) suggère de prendre un certain soin avant de tester l'hypothèse de non-proportionnalité. Pour corriger ces erreurs de spécification, la première tâche consiste à ajouter la variable omise dans le terme de droite avant d'utiliser le test de Therneau et al. (2000). Ensuite, il est nécessaire de vérifier si des interactions existent entre les variables et les ajouter quand nécessaire. Pour cela, il existe des algorithmes recherchant automatiquement des interactions parmi les variables du terme à droite (Harrell, 2015). Enfin, le test de formes non-linéaires est un moyen pour corriger une mauvaise spécification du modèle. La réalisation de ce test peut se faire à partir de fonctions polynomiales de variables ou à partir d'une méthode non paramétrique comme des splines. Une fois que la spécification du modèle est claire, le test de non-proportionnalité de Therneau et al. (2000) peut être appliqué au modèle. Des résultats significatifs devront mener à inclure une interaction entre la variable concernée et une certaine fonction de temps. Le test de Therneau et al. (2000) devra être répété. Enfin, si des risques non-proportionnels apparaissent toujours, il est souhaitable de considérer un modèle paramétrique dont l'hypothèse de risques proportionnels ne tient pas. Pour illustrer cela, Keele (2010) a réalisé une étude de simulations intéressante. La conclusion qu'il retient de cette simulation est qu'une fois la spécification est correcte, le test détecte bien les variables dépendantes du temps. Dans le cas d'interaction, cela semble plus compliqué (semble avoir une faible puissance). Il évoque que la puissance faible peut être due à la petite taille de l'échantillon. Le seuil de 0.05 pour le test serait peut-être trop rigide. Il conclut qu'une spécification correcte doit être précédée du test de non-proportionnalité, bien qu'il y ait une certaine difficulté de trouver des interactions ou des variables omises et toutes les variables doivent être testées pour la non-linéarité.

E.5 Résultats du test des risques proportionnels pour les différentes simulations

E.5.1 Résumé des résultats

Pour 10 variables

	p-valeur du test global	Risques proportionnels vérifiés
Cox/weibull	0.81	OUI
AFT/Log-normale	7.3×10^{-174}	NON
AH/Log-normale	5.8×10^{-28}	NON
AFT/Log-normale modifié	2.0×10^{-15}	NON
AFT/Log-normale sparse	2.9×10^{-13}	NON
AFT/Log-normale censuré	5.8×10^{-09}	NON

Table E.1 – Résultats du test global de Grambsch et al. (1994) sur les données simulées avec 10 variables

Pour 1000 variables

	p-valeur du test global	Risques proportionnels vérifiés
Cox/weibull	9.8×10^{-07}	NON
AFT/Log-normale	2.3×10^{-23}	NON
AH/Log-normale	8.9×10^{-05}	NON
AFT/Log-normale modifié	6.5×10^{-13}	NON
AFT/Log-normale sparse	3.4×10^{-06}	NON
AFT/Log-normale censuré	0	NON

Table E.2 – Résultats du test global de Grambsch et al. (1994) sur les données simulées avec 1000 variables

E.5.2 Détails des résultats pour la simulation Cox/Weibull

Pour 10 variables

res	chisq	df	p
X1	0.0250299890877384	1	0.874292357620342
X2	0.0500098433430021	1	0.823046146498587
X3	0.9907085958000666	1	0.319569249146088
X4	0.9506390544894420	1	0.329556684830964
X5	0.0124900427958349	1	0.911014608040370
X6	0.7644472924801999	1	0.381940327335452
X7	1.5590178359528959	1	0.211809085922943
X8	0.0025660946401527	1	0.959599132742552
X9	1.6601814050650128	1	0.197578832693500
X10	0.2742398403999111	1	0.600501727618854
GLOBAL	6.0412786170742310	10	0.811783346045548

Table E.3 – Résultats du test des risques proportionnels pour la simulation Cox/Weibull avec 10 variables

Pour 100 variables

res	chisq	df	p
X1	$2.02208704875884e-01$	1	0.6529439187825584
X2	$2.93077347863549e+00$	1	0.0869058467876710
X3	$3.91470027083715e+00$	1	0.0478655499602125
X4	$1.00430181743311e+00$	1	0.3162718280889716
X5	$4.98884200574901e-01$	1	0.4799908054524468
X6	$2.77754347244024e+00$	1	0.0955946904745301
X7	$2.89667835495336e+00$	1	0.0887622878712089
X8	$6.03060254267081e-02$	1	0.8060129070744179
X9	$5.06547780371575e-01$	1	0.4766371264921028
X10	$1.79712679946625e-03$	1	0.9661857813169563
X11	$1.12686855794058e-01$	1	0.7371056813228217
X12	$5.90198483286216e-04$	1	0.9806180982502680
X13	$1.51335771461772e+00$	1	0.2186274429798280
X14	$4.04026800934592e-03$	1	0.9493181121802690
X15	$3.08354247870767e+00$	1	0.0790880935754012
X16	$5.33036179697568e-01$	1	0.4653331761458446
X17	$2.41576302361961e-01$	1	0.6230699371224490
X18	$2.61257773213853e-04$	1	0.9871039781055435
X19	$6.07315204282622e-01$	1	0.4358004776410241
X20	$1.74049538761780e+00$	1	0.1870760700992575
X21	$3.42668430437973e+00$	1	0.0641507743623270
X22	$1.31556624378323e+00$	1	0.2513893241584232
X23	$2.24677818189404e-02$	1	0.8808493756721575
X24	$6.44508790617135e-02$	1	0.7995946194470189
X25	$2.78434878876361e+00$	1	0.0951893800662687
X26	$1.28739530750515e+00$	1	0.2565285177182633
X27	$1.67491555953831e-01$	1	0.6823508314837247
X28	$3.29704928951982e-01$	1	0.5658328882690663
X29	$4.78660806748380e-01$	1	0.4890295431038779
X30	$3.73401655455156e-01$	1	0.5411558880301778
X31	$2.57841780891258e-01$	1	0.6116067722367116
X32	$1.69398227961370e+00$	1	0.1930768478850450
X33	$6.48684152878363e-01$	1	0.4205835223957048
X34	$2.96790165118577e-02$	1	0.8632205598361373
X35	$3.13506478151894e-02$	1	0.8594603642311545
X36	$2.31334244618392e-01$	1	0.6305362662427013
X37	$2.00289661280144e-01$	1	0.6544871413126345
X38	$3.50958388404579e+00$	1	0.0610147785274326
X39	$1.28724239323306e-01$	1	0.7197584756108671
X40	$4.03729869003965e-01$	1	0.5251692566726640
X41	$1.71934976168426e-01$	1	0.6783978276550270
X42	$2.86681533993542e+00$	1	0.0904236362803588
X43	$3.76023784050820e-01$	1	0.5397389597668247
X44	$1.78641638188432e-02$	1	0.8936738972612571

X45	$3.90435655602076e-01$	1	0.5320704915602070
X46	$1.12713141844573e-01$	1	0.7370761555716617
X47	$5.64155211782552e-01$	1	0.4525908726432583
X48	$4.40870693457746e-03$	1	0.9470608873375739
X49	$1.53256890293694e+00$	1	0.2157273073018615
X50	$6.74193613038748e-01$	1	0.4115933321992640
X51	$1.67091102228076e-01$	1	0.6827100832609612
X52	$1.97469628069228e+00$	1	0.1599502593592873
X53	$1.09524823661632e-01$	1	0.7406857670637896
X54	$1.60859733393767e-02$	1	0.8990745626177318
X55	$1.09727927338030e+00$	1	0.2948639663064120
X56	$2.75534386378083e+00$	1	0.0969299742567672
X57	$4.16533546423806e-02$	1	0.8382819692432167
X58	$1.42425420636546e+00$	1	0.2327044488179756
X59	$3.33269563963328e+00$	1	0.0679154786686682
X60	$5.39009814034982e+00$	1	0.0202513318004501
X61	$8.26859890756484e-01$	1	0.3631820987348279
X62	$3.97752757111834e-01$	1	0.5282521145965331
X63	$2.26685447672189e+00$	1	0.1321679165756373
X64	$3.13808531726730e-01$	1	0.5753524674451509
X65	$2.64223222144916e+00$	1	0.1040572445081788
X66	$2.82018833821794e-01$	1	0.5953810443827758
X67	$8.66532438051935e-01$	1	0.3519170684486799
X68	$2.95667209491394e-03$	1	0.9566361659415131
X69	$5.27112914681848e-02$	1	0.8184108463500308
X70	$1.30461822403857e-01$	1	0.7179536854420904
X71	$3.15555815934360e-01$	1	0.5742907579512461
X72	$1.22275908324998e-02$	1	0.9119506470160662
X73	$1.09705687325323e-01$	1	0.7404794556386579
X74	$3.06234922723430e-04$	1	0.9860380765107543
X75	$2.02224179771473e-01$	1	0.6529315102958978
X76	$1.05374478417916e-01$	1	0.7454729626389622
X77	$2.68139848293585e+00$	1	0.1015265340794241
X78	$2.02636955238002e-02$	1	0.8868030742977959
X79	$1.49768839373680e+00$	1	0.2210273831444312
X80	$3.79734712031752e-03$	1	0.9508633641509573
X81	$2.34019369260339e+00$	1	0.1260738770372394
X82	$2.52221133512779e-01$	1	0.6155154332619744
X83	$1.12564152286040e+00$	1	0.2887069229913914
X84	$1.08096372205189e+00$	1	0.2984820648195846
X85	$1.03882612914518e-01$	1	0.7472191780034272
X86	$4.07680700011525e+00$	1	0.0434757411983932
X87	$4.57571561774626e-02$	1	0.8306178468874774
X88	$6.54871589156537e-01$	1	0.4183763250721695
X89	$6.23471874408916e-01$	1	0.4297600341930825
X90	$2.60155687488404e-01$	1	0.6100132156001730

X91	$7.89602967083999e-01$	1	0.3742202065272117
X92	$1.89537597185656e-02$	1	0.8904991323419873
X93	$8.36959975489795e-02$	1	0.7723498186297102
X94	$4.22809612163602e-02$	1	0.8370851382155716
X95	$1.82221186065716e-01$	1	0.6694712965461378
X96	$5.80514203967822e-01$	1	0.4461108399567369
X97	$3.74514039799180e-01$	1	0.5405539526805316
X98	$3.86551111512033e-01$	1	0.5341178779776516
X99	$2.06156283491995e+00$	1	0.1510551766434660
X100	$1.73352408736349e+00$	1	0.1879614640432501
GLOBAL	$1.04811048018200e+02$	100	0.3512818122105701

Table E.4 – Résultats du test des risques proportionnels pour la simulation Cox/Weibull avec 100 variables

Pour 1000 variables

res	chisq	df	p
X1	$7.85011662066931e+00$	1	$5.08174290935302e-03$
X2	$2.38428797643235e+00$	1	$1.22560723220045e-01$
X3	$5.05338457720756e-01$	1	$4.77163795016606e-01$
X4	$1.23269159518504e+00$	1	$2.66884224803214e-01$
X5	$3.10337516071910e-01$	1	$5.77473145832305e-01$
X6	$5.79750821168084e-02$	1	$8.09725512691040e-01$
X7	$3.26786123638758e+00$	1	$7.06496751090467e-02$
X8	$1.21287963098182e-01$	1	$7.27641766287524e-01$
X9	$1.31717583847393e+01$	1	$2.84199930717465e-04$
X10	$2.96937262788352e-01$	1	$5.85808883316480e-01$
X11	$6.45173673900510e-01$	1	$4.21843520700315e-01$
X12	$1.00833148072115e+00$	1	$3.15302896647615e-01$
X13	$1.02379552094121e+00$	1	$3.11620389655010e-01$
X14	$5.01654140739402e-02$	1	$8.22775690594024e-01$
X15	$2.39861365370689e+00$	1	$1.21442831477986e-01$
X16	$7.49290222002644e-01$	1	$3.86701043757581e-01$
X17	$4.87031750295744e-03$	1	$9.44362669039849e-01$
X18	$4.38809343429576e-01$	1	$5.07697695080655e-01$
X19	$7.38762615200044e-01$	1	$3.90057545105749e-01$
X20	$2.61880729288931e-01$	1	$6.08831002922091e-01$
X21	$9.11717401924501e-01$	1	$3.39659151748569e-01$
X22	$5.04924143029980e-01$	1	$4.77344450190136e-01$
X23	$5.14210166199711e-02$	1	$8.20608793057697e-01$
X24	$7.11862254208182e-01$	1	$3.98826281006828e-01$
X25	$2.20968010038451e-01$	1	$6.38303331128083e-01$
X26	$2.52503460721382e-01$	1	$6.15317804200697e-01$
X27	$3.28870628009396e+00$	1	$6.97579663239122e-02$
X28	$1.80734289886253e+00$	1	$1.78827303129501e-01$
X29	$3.18386465822310e-01$	1	$5.72578937149953e-01$

X30	$1.13604267629580e+00$	1	$2.86490085559636e-01$
X31	$5.96931390957069e-02$	1	$8.06981605555448e-01$
X32	$1.93058099464228e+00$	1	$1.64694895179341e-01$
X33	$7.42397044147034e-02$	1	$7.85260932685172e-01$
X34	$1.93796356056510e+00$	1	$1.63889818203303e-01$
X35	$6.25406915569986e-02$	1	$8.02524423004003e-01$
X36	$1.78922883697255e-01$	1	$6.72300535896008e-01$
X37	$2.69708664994475e+00$	1	$1.00531797516863e-01$
X38	$3.35518208210718e-03$	1	$9.53809218880525e-01$
X39	$2.04262386136865e+00$	1	$1.52945629704187e-01$
X40	$3.74058930611928e+00$	1	$5.31057123621301e-02$
X41	$2.36320060876456e-02$	1	$8.77824955004810e-01$
X42	$1.86892018903770e-01$	1	$6.65516050527873e-01$
X43	$2.05698027216807e+00$	1	$1.51510161345584e-01$
X44	$9.03423296288761e-02$	1	$7.63742404708404e-01$
X45	$2.23644486172553e-01$	1	$6.36276923977596e-01$
X46	$1.12056332195933e-01$	1	$7.37815067006911e-01$
X47	$6.25525824517951e+00$	1	$1.23825195701727e-02$
X48	$2.02664351304816e-01$	1	$6.52578799186717e-01$
X49	$9.00255089018902e-01$	1	$3.42713321622065e-01$
X50	$9.09520993435309e-02$	1	$7.62970230049041e-01$
X51	$4.90633304954233e-01$	1	$4.83644937567410e-01$
X52	$1.61095790123538e+00$	1	$2.04357202979981e-01$
X53	$2.13321451927865e+00$	1	$1.44138203963432e-01$
X54	$4.22078626872447e+00$	1	$3.99316528055894e-02$
X55	$2.97466488280427e-01$	1	$5.85475081624368e-01$
X56	$1.59157287950398e-01$	1	$6.89933569068406e-01$
X57	$3.28718702993306e-02$	1	$8.56127386665274e-01$
X58	$4.00004929662938e+00$	1	$4.54989331305264e-02$
X59	$6.97312505870252e-02$	1	$7.91728481296741e-01$
X60	$1.29766708422386e-01$	1	$7.18674046292476e-01$
X61	$2.31605365266145e+00$	1	$1.28044510053713e-01$
X62	$2.64111153843798e+00$	1	$1.04130665740537e-01$
X63	$8.01868609014868e-01$	1	$3.70535272748035e-01$
X64	$4.97666044570291e+00$	1	$2.56915311721795e-02$
X65	$6.98090386264409e+00$	1	$8.23839904961816e-03$
X66	$1.17088899200288e-01$	1	$7.32213514712484e-01$
X67	$3.16991262945594e+00$	1	$7.50064587039567e-02$
X68	$7.01985324359968e-01$	1	$4.02117399550124e-01$
X69	$1.78353595229398e-01$	1	$6.72791969208124e-01$
X70	$1.17056852263850e+00$	1	$2.79284455215985e-01$
X71	$5.70051428342792e-01$	1	$4.50238480854067e-01$
X72	$9.40963969953713e-02$	1	$7.59032633311465e-01$
X73	$1.07381669514326e-02$	1	$9.17466846524916e-01$
X74	$5.72356485486733e-01$	1	$4.49324028411157e-01$
X75	$2.60158481876489e+00$	1	$1.06756912361410e-01$

X76	$1.32241457277705e+00$	1	$2.50159169729417e-01$
X77	$9.50244865659731e-01$	1	$3.29656976702041e-01$
X78	$6.64096315413624e-01$	1	$4.15117503935931e-01$
X79	$1.90067947379604e+00$	1	$1.68002284021183e-01$
X80	$4.71105027066545e-01$	1	$4.92479243232702e-01$
X81	$1.72240237012163e+00$	1	$1.89384100680416e-01$
X82	$2.83294876141121e-01$	1	$5.94549725658857e-01$
X83	$3.02750523688700e+00$	1	$8.18637996192076e-02$
...	...	...	...
X999	$3.89766206503544e+00$	1	$4.83533518912721e-02$
X1000	$1.32791593347415e+00$	1	$2.49176314018975e-01$
GLOBAL	$3.26996054451182e+03$	1000	$1.77980058138885e-238$

Table E.5 – Résultats du test des risques proportionnels pour la simulation Cox/Weibull avec 1000 variables

Annexe F

Résultats détaillés de la prédiction de la survie en grande dimension

F.1 Comportement des données simulées

F.1.1 Distribution des données simulées

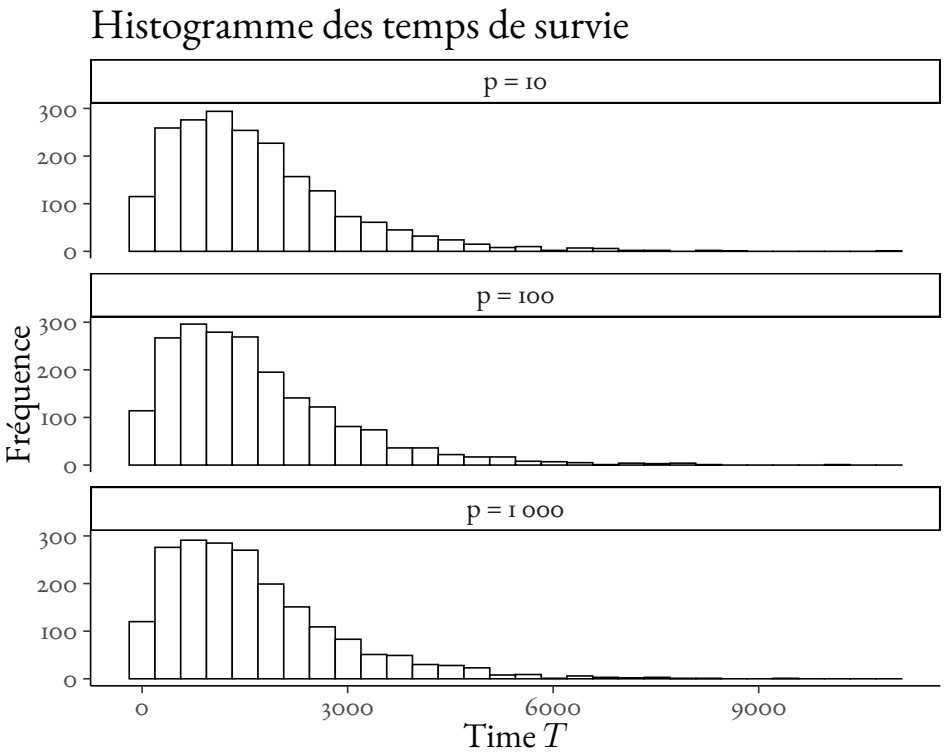


Figure F.1 – Distribution des temps de survie simulés par un modèle AFT/Log-normale censuré

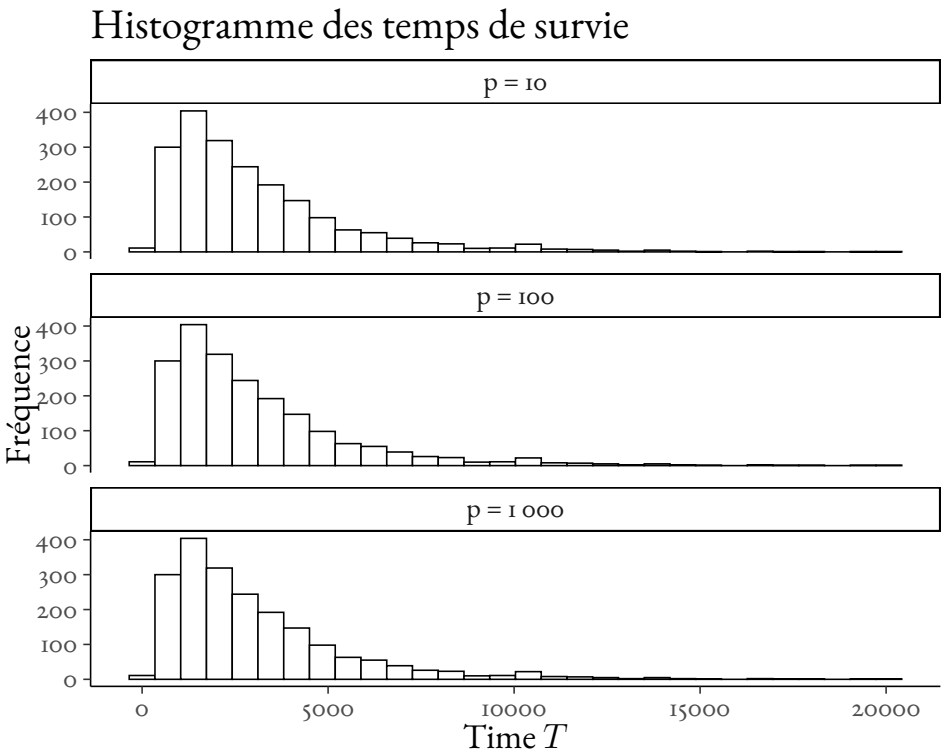


Figure F.2 – Distribution des temps de survie simulés par un modèle AFT/Log-normale sparse

F.1.2 Courbes de survie des données simulées

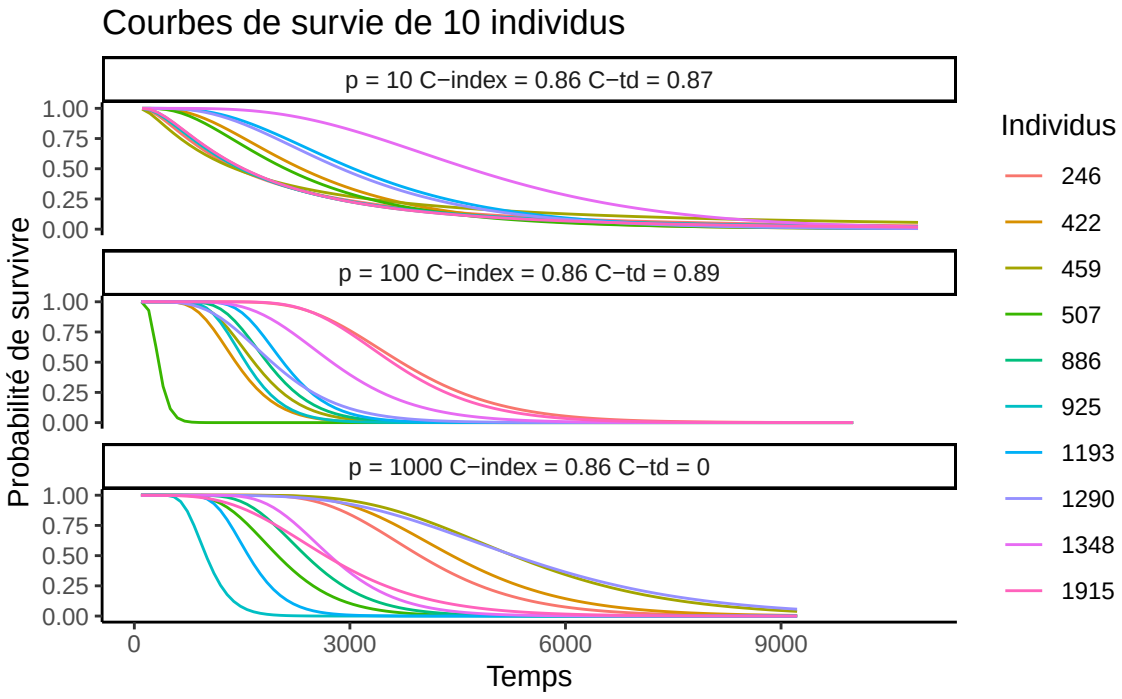


Figure F.3 – Courbes de survie des individus simulés par un modèle AFT/Log-normale censuré

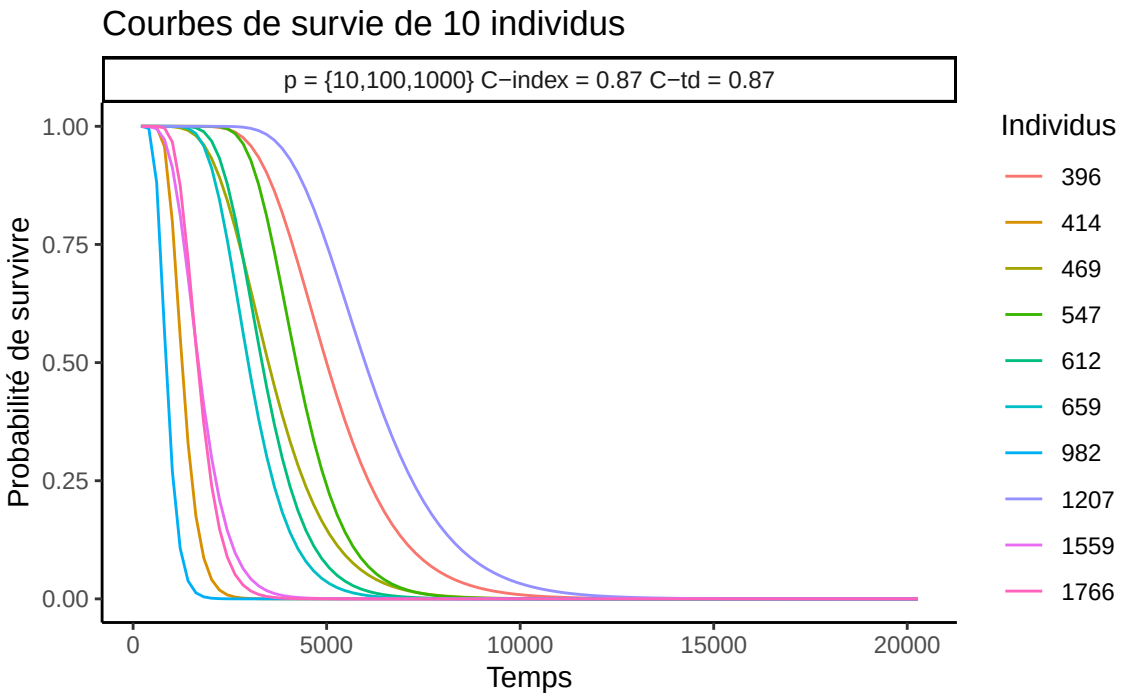


Figure F.4 – Courbes de survie des individus simulés par un modèle AFT/Log-normale sparse

Annexe G

Illustration de la performance des réseaux de neurones pour la prédiction de la survie sur des données du cancer du sein

Nous présentons dans cette annexe les premiers résultats des méthodes étudiées dans le manuscrit obtenus sur le jeu de données du cancer du sein. Ces résultats sont préliminaires car il serait plus pertinent de faire une sélection de variables avant d'appliquer les méthodes sur ce jeu de données. Le nombre de variables de ce jeu de données est très important et notre étude de comparaison a montré que la performance des méthodes était meilleure quand le nombre de variables étaient de l'ordre de 100 variables.

Description des données :
Les données Breast proviennent du projet (GSE6532) de Loi et al. (2007) dont le but est de caractériser les sous-types des récepteurs oestrogènes positifs à partir de profils d'expression des gènes. Les récepteurs d'oestrogènes sont un groupe de protéines présent à l'intérieur des cellules, qui est activé par l'hormone oestrogène. Différentes formes de récepteurs existent se reportant à des sous-types de récepteurs d'oestrogènes. Si une sur-expression est montrée pour ces récepteurs, alors ils sont considérées comme *ER-positve*. Ces données sont accessibles à l'adresse suivante : `https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE6532`. De plus, ces données ont été étudiées dans le cadre de la survie par Tian et al. (2012). La durée de survie d'intérêt considéré par Tian et al. (2012) est la survie sans rechute. Nous l'avons également considéré comme variable d'intérêt. Considérant seulement les patients avec les données complètes, nous avons eu accès à l'expression de 44 928 gènes pour 246 patients. Parmi l'ensemble des patients, 64% sont censurés.

Résultats :
Sur la Table G.1, nous pouvons voir que Cox-nnet est la méthode la plus performante concernant le C-index et l'IBS sur le jeu de données du cancer du sein. Ce jeu de données possède un nombre important de variables une petite taille d'échantillon et nous pouvons observer que cela impacte directement les résultats. L'IBS est très élevé quel que soit la méthode. Quand sa valeur est supérieure à 0.25 cela signifie que la prédiction est aléatoire. Cox-nnet a le meilleur IBS, mais celui est très élevé. Cela est confirmé par la Figure G.1, nous observons que la courbe de Cox-nnet est la plus proche en comparant aux autres méthodes de la courbe de l'estimateur Kaplan-Meier mais elle est éloignée de la courbe de l'estimateur de Kaplan-Meier. De plus, le pourcentage de données censurées est très élevé (64%) sur ce jeu de données. Cela peut expliquer que Cox-nnet ait la meillleure valeur de C-index car nous avons pu voir sur leur étude à partir des données simulées que la modélisation de CoxL1 gérait mieux la censure que la modélisation à un temps discret.

		CoxL1	Cox-nnet	NNsurv Deep	NNsurv
Breast	C_{td}	0.5358	**0.6939**	0.6725	0.6681
	IBS	0.2404	**0.2348**	0.2854	0.2898

Table G.1 – Résultats des différentes méthodes sur des jeux de données du cancer du sein

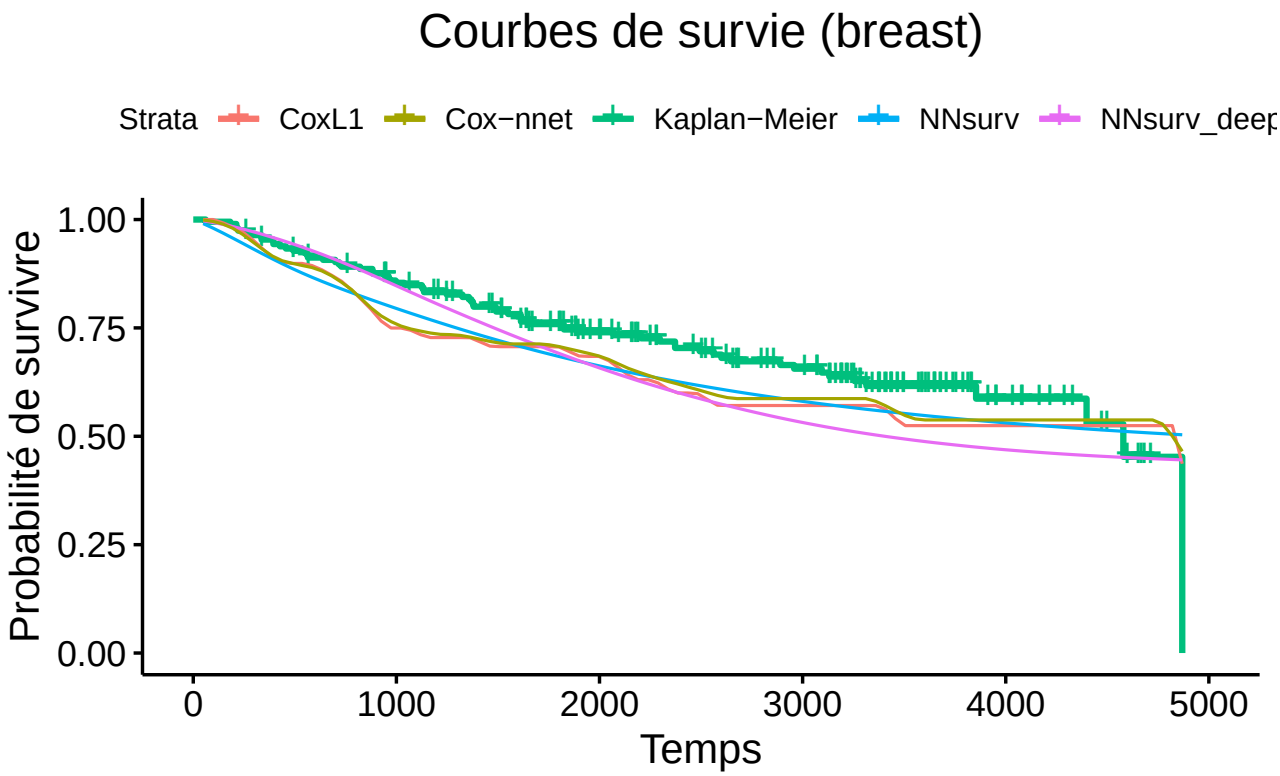


Figure G.1 – Courbes de survie obtenues à partir des méthodes sur le jeu de données du cancer du sein (Breast)

Références

Aalen, Odd (1978). « Nonparametric Inference for a Family of Counting Processes ». In : *The Annals of Statistics* 6.4, p. 701-726.

Akaike, Hirotogu (1998). « Information Theory and an Extension of the Maximum Likelihood Principle ». In : New York : Springer Science+Business Media, p. 199-213. doi : 10.1007/978-1-4612-1694-0_15.

Anders, Simon et Wolfgang Huber (2010). « Differential expression analysis for sequence count data ». en. In : *Genome Biology* 11.10, R106. doi : 10.1186/gb-2010-11-10-r106.

Antolini, Laura, Patrizia Boracchi et Elia Biganzoli (2005). « A time-dependent discrimination index for survival data ». en. In : *Statistics in Medicine* 24.24, p. 3927-3944. doi : 10.1002/sim.2427.

Antoniou, A. C., P. D. P. Pharoah, G. McMullan, N. E. Day, M. R. Stratton, J. Peto, B. J. Ponder et D. F. Easton (2002). « A comprehensive model for familial breast cancer incorporating BRCA1, BRCA2 and other genes ». en. In : *British Journal of Cancer* 86.1, p. 76-83. doi : 10.1038/sj.bjc.6600008.

Arita, Héctor T. (2017). « Multisite and multispecies measures of overlap, co-occurrence, and co-diversity ». en. In : *Ecography* 40.6, p. 709-718. doi : 10.1111/ecog.01942.

Arita, Héctor T., J. Andrés Christen, Pilar Rodríguez et Jorge Soberón (2008). « Species diversity and distribution in presence-absence matrices : mathematical relationships and biological implications ». eng. In : *The American Naturalist* 172.4, p. 519-532. doi : 10.1086/590954.

Arteaga, Carlos L., Mark X. Sliwkowski, C. Kent Osborne, Edith A. Perez, Fabio Puglisi et Luca Gianni (2012). « Treatment of HER2-positive breast cancer : current status and future perspectives ». en. In : *Nature Reviews Clinical Oncology* 9.1, p. 16-32. doi : 10.1038/nrclinonc.2011.177.

Audoux, Jérôme, Nicolas Philippe, Rayan Chikhi, Mikaël Salson, Mélina Gallopin, Marc Gabriel, Jérémy Le Coz, Emilie Drouineau, Thérèse Commes et Daniel Gautheret (2017). « DE-kupl : exhaustive capture of biological variation in RNA-seq data through k-mer decomposition ». In : *Genome Biology* 18.1, p. 243. doi : 10.1186/s13059-017-1372-2.

Barut, Emre, Jianqing Fan et Anneleen Verhasselt (2016). « Conditional Sure Independence Screening ». en. In : *Journal of the American Statistical Association* 111.515, p. 1266-1277. doi : 10.1080/01621459.2015.1092974.

Baselga, Andrés (2010). « Partitioning the turnover and nestedness components of beta diversity ». en. In : *Global Ecology and Biogeography* 19.1, p. 134-143. doi : 10.1111/j.1466-8238.2009.00490.x.

– (2013). « Multiple site dissimilarity quantifies compositional heterogeneity among several sites, while average pairwise dissimilarity may be misleading ». en. In : *Ecography* 36.2, p. 124-128. doi : 10.1111/j.1600-0587.2012.00124.x.

Baselga, Andrés et C. David L. Orme (2012). « betapart : an R package for the study of beta diversity ». en. In : *Methods in Ecology and Evolution* 3.5, p. 808-812. doi : 10.1111/j.2041-210X.2012.00224.x.

Bender, Ralf, Thomas Augustin et Maria Blettner (2005). « Generating survival times to simulate Cox proportional hazards models ». en. In : *Statistics in Medicine* 24.11, p. 1713-1723. doi : 10.1002/sim.2059.

Bengio, Yoshua, Aaron Courville et Pascal Vincent (2013). « Representation learning : A review and new perspectives ». In : *IEEE transactions on pattern analysis and machine intelligence* 35.8, p. 1798-1828.

Benjamini, Yoav et Yosef Hochberg (1995). « Controlling The False Discovery Rate - A Practical And Powerful Approach To Multiple Testing ». In : *J. Royal Statist. Soc., Series B* 57, p. 289-300. doi : 10.2307/2346101.

Bickel, Peter J., Bo Li, Alexandre B. Tsybakov, Sara A. van de Geer, Bin Yu, Teófilo Valdés, Carlos Rivero, Jianqing Fan et Aad van der Vaart (2006). « Regularization in statistics ». en. In : *Test* 15.2, p. 271-344. doi : 10.1007/BF02607055.

Biganzoli, Elia, Patrizia Boracchi, Luigi Mariani et Ettore Marubini (1998). « Feed forward neural networks for the analysis of censored survival data : a partial logistic regression approach ». en. In : *Statistics in Medicine* 17.10, p. 1169-1186. doi : 10.1002/(SICI)1097-0258(19980530)17:10<1169::AID-SIM796>3.0.CO;2-D.

Bottou, Léon (jan. 1999). « On-line Learning and Stochastic Approximations ». en. In : *On-Line Learning in Neural Networks*. Sous la dir. de David Saad. 1re éd. Cambridge University Press, p. 9-42. doi : 10.1017/CBO9780511569920.003.

Boulesteix, Anne-Laure, Silke Janitza, Jochen Kruppa et Inke R. König (2012). « Overview of random forest methodology and practical guidance with emphasis on computational biology and bioinformatics ». In : *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* doi : 10.1002/widm.1072.

Breiman, Leo (2001). « Random Forests ». en. In : *Machine Learning* 45.1, p. 5-32. doi : 10.1023/A:1010933404324.

Breslow, N. E., L. Edler et J. Berger (1984). « A Two-Sample Censored-Data Rank Test for Acceleration ». en. In : *Biometrics* 40.4, p. 1049. doi : 10.2307/2531155.

Brier, Glenn W. (1950). « Verification of forecasts expressed in terms of probability ». en. In : *Monthly Weather Review* 78.1, p. 1-3. doi : 10.1175/1520-0493(1950)078<0001:VOFEIT>2.0.CO;2.

Brown, S. F., A. J. Branford et W. Moran (1997). « On the use of artificial neural networks for the analysis of survival data ». In : *IEEE Transactions on Neural Networks* 8.5, p. 1071-1077. doi : 10.1109/72.623209.

Bussy, Simon, Agathe Guilloux, Stéphane Gaïffas et Anne-Sophie Jannot (2019). « C-mix : A high-dimensional mixture model for censored durations, with applications to genetic data ». In : *Statistical Methods in Medical Research* 28.5. PMID : 29658407, p. 1523-1539. doi : 10.1177/0962280218766389. eprint : https://doi.org/10.1177/0962280218766389.

Campos, David, Sérgio Matos et José Luís Oliveira (2013). « Gimli : open source and high-performance biomedical name recognition ». In : *BMC Bioinformatics* 14.1, p. 54. doi : 10.1186/1471-2105-14-54.

Chen, Ying Qing et Mei-Cheng Wang (2000). « Analysis of Accelerated Hazards Models ». In : *Journal of the American Statistical Association* 95.450, p. 608-618. doi : 10.1080/01621459.2000.10474236.

Ching, Travers, Xun Zhu et Lana X. Garmire (2018). « Cox-nnet : An artificial neural network method for prognosis prediction of high-throughput omics data ». en. In : *PLOS Computational Biology* 14.4. Sous la dir. de Florian Markowetz, e1006076. doi : 10.1371/journal.pcbi.1006076.

Claeskens, Gerda et Fabrizio Consentino (2008). « Variable Selection with Incomplete Covariate Data ». en. In : *Biometrics* 64.4, p. 1062-1069. doi : 10.1111/j.1541-0420.2008.01003.x.

Cottrell, Marie, Madalina Olteanu, Fabrice Rossi, Joseph Rynkiewicz et Nathalie Villa-Vialaneix (2012). « Neural networks for complex data ». In : *KI-Künstliche Intelligenz* 26.4, p. 373-380.

Cox, D. R. (1972). « Regression Models and Life-Tables ». In : *Journal of the Royal Statistical Society. Series B (Methodological)* 34.2, p. 187-220.

– (1975). « Partial likelihood ». en. In : *Biometrika* 62.2, p. 269-276. doi : 10.1093/biomet/62.2.269.

Curtis, Christina, Sohrab P Shah, Suet-Feung Chin, Gulisa Turashvili, Oscar M Rueda, Mark J Dunning, Doug Speed, Andy G Lynch, Shamith Samarajiwa, Yinyin Yuan et al. (2012). « The genomic and transcriptomic architecture of 2,000 breast tumours reveals novel subgroups ». In : *Nature* 486.7403, p. 346-352.

Fan, Jianqing, Yang Feng et Yichao Wu (2010a). *High-dimensional variable selection for Cox's proportional hazards model*. EN. Institute of Mathematical Statistics. doi : 10.1214/10-IMSCOLL606.

Fan, Jianqing et Runze Li (2001). « Variable Selection via Nonconcave Penalized Likelihood and Its Oracle Properties ». In : *Journal of the American Statistical Association* 96.456, p. 1348-1360.

Fan, Jianqing et Jinchi Lv (2008). « Sure independence screening for ultrahigh dimensional feature space ». en. In : *Journal of the Royal Statistical Society : Series B (Statistical Methodology)* 70.5, p. 849-911. doi : 10.1111/j.1467-9868.2008.00674.x.

Fan, Jianqing, Richard Samworth et Yichao Wu (2009). « Ultrahigh Dimensional Feature Selection : Beyond The Linear Model ». In : *J. Mach. Learn. Res.* 10, p. 2013-2038.

Fan, Jianqing et Rui Song (2010b). « Sure independence screening in generalized linear models with NP-dimensionality ». In : *The Annals of Statistics* 38.6, p. 3567-3604.

Faraggi, David et Richard Simon (1995). « A neural network model for survival data ». en. In : *Statistics in Medicine* 14.1, p. 73-82. doi : 10.1002/sim.4780140108.

Fine, J. P., Z. Ying et L. J. Wei (1998). « On the Linear Transformation Model for Censored Data ». In : *Biometrika* 85.4, p. 980-986.

Fleming, Thomas R et David P. Harrington (2005). *Counting Processes and Survival Analysis*. 1re éd. John Wiley & Sons, Ltd. doi : 10.1002/9781118150672.

Fotso, Stephane (2018). *Deep Neural Networks for Survival Analysis Based on a Multi-Task Framework*. arXiv : 1801.05512 [stat.ML].

Gallopin, Mélina (2015). « Classification et inférence de réseaux pour les données RNA-seq ». fr. Thèse de doct. Université Paris-Saclay.

Gamelin, E., M. Boisdron-Celle et A. Morel (2014). « La dihydropyrimidine déshydrogénase (DPD) ». fr. In : *Oncologie* 16.2-3, p. 96-102. doi : 10.1007/s10269-014-2373-3.

Gensheimer, Michael F. et Balasubramanian Narasimhan (2018). « A Scalable Discrete-Time Survival Model for Neural Networks ». In : *arXiv :1805.00917 [cs, stat]*. arXiv : 1805.00917.

Gerds, Thomas A. et Martin Schumacher (2006). « Consistent Estimation of the Expected Brier Score in General Survival Models with Right-Censored Event Times ». fr. In : *Biometrical Journal* 48.6, p. 1029-1040. doi : 10.1002/bimj.200610301.

Gill, Richard et Martin Schumacher (1987). « A simple test of the proportional hazards assumption ». en. In : *Biometrika* 74.2, p. 289-300. doi : 10.1093/biomet/74.2.289.

Goldenshluger, Alexander et Oleg Lepski (2011). « Bandwidth selection in kernel density estimation : Oracle inequalities and adaptive minimax optimality ». EN. In : *Annals of Statistics* 39.3, p. 1608-1632. doi : 10.1214/11-AOS883.

Goncalves, Andre, Braden Soper, Mari Nygård, Jan F Nygård, Priyadip Ray, David Widemann et Ana Paula Sales (2020). « Improving five-year survival prediction via multitask learning across HPV-related cancers ». In : *PloS one* 15.11.

Graf, Erika, Claudia Schmoor, Willi Sauerbrei et Martin Schumacher (1999). « Assessment and comparison of prognostic classification schemes for survival data ». en. In : *Statistics in Medicine* 18.17-18, p. 2529-2545. doi : 10.1002/(SICI)1097-0258(19990915/30)18:17/18<2529::AID-SIM274>3.0.CO;2-5.

Grambsch, Patricia M. et Terry M. Therneau (1994). « Proportional Hazards Tests and Diagnostics Based on Weighted Residuals ». In : *Biometrika* 81.3, p. 515-526. doi : 10.2307/2337123.

Gremel, Gabriela et al. (2017). « A systematic search strategy identifies cubilin as independent prognostic marker for renal cell carcinoma ». eng. In : *BMC cancer* 17.1, p. 9. doi : 10.1186/s12885-016-3030-6.

Guyon, Isabelle et André Elisseeff (2003). « An Introduction to Variable and Feature Selection ». In : *J. Mach. Learn. Res.* 3, p. 1157-1182.

Guyon, Isabelle, Jason Weston, Stephen Barnhill et Vladimir Vapnik (2002). « Gene Selection for Cancer Classification using Support Vector Machines ». en. In : *Machine Learning* 46.1, p. 389-422. doi : 10.1023/A:1012487302797.

Hao, Ning et Hao Helen Zhang (2014). « Interaction Screening for Ultrahigh-Dimensional Data ». In : *Journal of the American Statistical Association* 109.507. PMID : 25386043, p. 1285-1301. doi : 10.1080/01621459.2014.881741. eprint : https://doi.org/10.1080/01621459.2014.881741.

Harrell, FE, KL Lee, DB Matchar et TA Reichert (1985). « Regression models for prognostic prediction : advantages, problems, and suggested solutions ». In : *Cancer treatment reports* 69.10, 1071—1077.

Harrell, Frank E. (1982). « Evaluating the Yield of Medical Tests ». en. In : *JAMA : The Journal of the American Medical Association* 247.18, p. 2543. doi : 10.1001/jama.1982.03320430047030.

– (2015). *Regression Modeling Strategies : With Applications to Linear Models, Logistic and Ordinal Regression, and Survival Analysis*. en. Springer Series in Statistics. Cham : Springer International Publishing. doi : 10.1007/978-3-319-19425-7.

Harrow, Jennifer et al. (2012). « GENCODE : The reference human genome annotation for The ENCODE Project ». en. In : *Genome Research* 22.9, p. 1760-1774. doi : 10.1101/gr.135350.111.

Hastie, Trevor, Robert Tibshirani et Martin Wainwright (2015). *Statistical Learning with Sparsity : The Lasso and Generalizations*. en. CRC Press.

Hoerl, Arthur E. et Robert W. Kennard (1970). « Ridge Regression : Biased Estimation for Nonorthogonal Problems ». In : *Technometrics* 12.1, p. 55-67. doi : 10.2307/1267351.

Hong, Hyokyoung G., Jian Kang et Yi Li (2018). « Conditional screening for ultra-high dimensional covariates with survival outcomes ». en. In : *Lifetime Data Analysis* 24.1, p. 45-71. doi : 10.1007/s10985-016-9387-7.

Huang, Shujun, Nianguang Cai, Pedro Penzuti Pacheco, Shavira Narrandes, Yang Wang et Wayne Xu (2018). « Applications of Support Vector Machine (SVM) Learning in Cancer Genomics ». eng. In : *Cancer Genomics & Proteomics* 15.1, p. 41-51. doi : 10.21873/cgp.20063.

Imbert, Alyssa, Armand Valsesia, Caroline Le Gall, Claudia Armenise, Gregory Lefebvre, Pierre-Antoine Gourraud, Nathalie Viguerie et Nathalie Villa-Vialaneix (déc. 2017). « Multiple hot-deck imputation for network inference from RNA sequencing data ». In : *Bioinformatics* 34.10, p. 1726-1732. doi : 10.1093/bioinformatics/btx819. eprint : https://academic.oup.com/bioinformatics/article-pdf/34/10/1726/25118161/btx819.pdf.

Ishwaran, Hemant, Udaya B. Kogalur, Eugene H. Blackstone et Michael S. Lauer (sept. 2008). « Random survival forests ». In : *Ann. Appl. Stat.* 2.3, p. 841-860. doi : 10.1214/08-AOAS169.

Jardillier, Rémy, Florent Chatelain et Laurent Guyon (2020). *Benchmark of lasso-like penalties in the Cox model for TCGA datasets reveal improved performance with pre-filtering and wide differences between cancers*. en. preprint. Bioinformatics. doi : 10.1101/2020.03.09.984070.

Johnson, W Evan, Cheng Li et Ariel Rabinovic (2007). « Adjusting batch effects in microarray expression data using empirical Bayes methods ». In : *Biostatistics* 8.1, p. 118-127.

Kalbfleisch, John D. et Ross L. Prentice (2002). *The Statistical Analysis of Failure Time Data : Kalbfleisch/The Statistical*. Wiley Series in Probability and Statistics. Hoboken, NJ, USA : John Wiley & Sons, Inc. doi : 10.1002/9781118032985.

Kaplan, E. L. et Paul Meier (1958). « Nonparametric Estimation from Incomplete Observations ». In : *Journal of the American Statistical Association* 53.282, p. 457-481. doi : 10.1080/01621459.1958.10501452.

Katzman, Jared L., Uri Shaham, Alexander Cloninger, Jonathan Bates, Tingting Jiang et Yuval Kluger (2018). « DeepSurv : personalized treatment recommender system using a Cox proportional hazards deep neural network ». In : *BMC Medical Research Methodology* 18.1, p. 24. doi : 10.1186/s12874-018-0482-1.

Keele, Luke (2010). « Proportionally Difficult : Testing for Nonproportional Hazards in Cox Models ». en. In : *Political Analysis* 18.2, p. 189-205. doi : 10.1093/pan/mpp044.

Kim, J.-D., T. Ohta, Y. Tateisi et J. Tsujii (2003). « GENIA corpus—a semantically annotated corpus for bio-textmining ». en. In : *Bioinformatics* 19.suppl_1, p. i180-i182. doi : 10.1093/bioinformatics/btg1023.

Kim, Jin-Dong, Tomoko Ohta, Yoshimasa Tsuruoka, Yuka Tateisi et Nigel Collier (2004). « Introduction to the bio-entity recognition task at JNLPBA ». In : *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications*. JNLPBA '04. USA : Association for Computational Linguistics, p. 70-75.

Kingma, Diederik P et Max Welling (2013). « Auto-encoding variational bayes ». In : *arXiv preprint arXiv :1312.6114*.

Klein, John P. et Melvin L. Moeschberger (1997). « Basic Quantities and Models ». en. In : *Survival Analysis : Techniques for Censored and Truncated Data*. Sous la dir. de John P. Klein et Melvin L. Moeschberger. Statistics for Biology and Health. New York, NY : Springer, p. 21-53. doi : 10.1007/978-1-4757-2728-9_2.

Kohonen, Teuvo (1990). « The self-organizing map ». In : *Proceedings of the IEEE* 78.9, p. 1464-1480.

Kvamme, Håvard et Ørnulf Borgan (2019a). « Continuous and Discrete-Time Survival Prediction with Neural Networks ». In : *arXiv :1910.06724 [cs, stat]*. arXiv : 1910.06724.

Kvamme, Håvard, Ørnulf Borgan et Ida Scheel (2019b). « Time-to-Event Prediction with Neural Networks and Cox Regression ». In : *arXiv :1907.00825 [cs, stat]*. arXiv : 1907.00825.

Lafferty, John, Andrew McCallum et Fernando Pereira (2001). « Conditional Random Fields : Probabilistic Models for Segmenting and Labeling Sequence Data ». In : *Departmental Papers (CIS)*.

Lander, Eric S. et al. (2001). « Initial sequencing and analysis of the human genome ». en. In : *Nature* 409.6822, p. 860-921. doi : 10.1038/35057062.

Law, Charity W., Yunshun Chen, Wei Shi et Gordon K. Smyth (2014). « voom : Precision weights unlock linear model analysis tools for RNA-seq read counts ». eng. In : *Genome Biology* 15.2, R29. doi : 10.1186/gb-2014-15-2-r29.

Lecun, Y. (1985). « Une procedure d'apprentissage ponr reseau a seuil asymetrique ». In : *Proceedings of Cognitiva 85*, p. 599-604.

Lee, Changhee, William R Zame, Jinsung Yoon et Mihaela van der Schaar (2018). « DeepHit : A Deep Learning Approach to Survival Analysis With Competing Risks. » In : *AAAI*, p. 2314-2321.

Leemis, Lawrence M., Li-Hsing Shih et Kurt Reynertson (1990). « Variate generation for accelerated life and proportional hazards models with time dependent covariates ». en. In : *Statistics & Probability Letters* 10.4, p. 335-339. doi : 10.1016/0167-7152(90)90052-9.

Lemler, Sarah (2014). « Estimation for counting processes with high-dimensional covariates ». These de doctorat. Evry-Val d'Essonne.

– (2016). « Oracle inequalities for the Lasso in the high-dimensional Aalen multiplicative intensity model ». EN. In : *Annales de l'Institut Henri Poincaré, Probabilités et Statistiques* 52.2, p. 981-1008. doi : 10.1214/14-AIHP662.

Liestbl, Knut, Per Kragh Andersen et Ulrich Andersen (1994). « Survival analysis and neural nets ». en. In : *Statistics in Medicine* 13.12, p. 1189-1200. doi : 10.1002/sim.4780131202.

Lin, Aifen et Wei-Hua Yan (2018). « Heterogeneity of HLA-G Expression in Cancers : Facing the Challenges ». In : *Frontiers in Immunology* 9. doi : 10.3389/fimmu.2018.02164.

Lin, D. Y. (1991). « Goodness-of-Fit Analysis for the Cox Regression Model Based on a Class of Parameter Estimators ». In : *Journal of the American Statistical Association* 86.415, p. 725-728. doi : 10.2307/2290404.

Liu, Hongfang, Zhang-Zhi Hu, Jian Zhang et Cathy Wu (2006). « BioThesaurus : a web-based thesaurus of protein and gene names ». en. In : *Bioinformatics* 22.1, p. 103-105. doi : 10.1093/bioinformatics/bti749.

Loi, Sherene et al. (2007). « Definition of clinically distinct molecular subtypes in estrogen receptor-positive breast carcinomas through genomic grade ». eng. In : *Journal of Clinical Oncology : Official Journal of the American Society of Clinical Oncology* 25.10, p. 1239-1246. doi : 10.1200/JCO.2006.07.1522.

Love, Michael I., Wolfgang Huber et Simon Anders (2014). « Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2 ». In : *Genome Biology* 15.12, p. 550. doi : 10.1186/s13059-014-0550-8.

Mani, D. R., James Drew, Andrew Betz et Piew Datta (1999). « Statistics and data mining techniques for lifetime value modeling ». In : *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*. KDD '99. San Diego, California, USA : Association for Computing Machinery, p. 94-103. doi : 10.1145/312129.312205.

Marchet, Camille, Zamin Iqbal, Daniel Gautheret, Mikaël Salson et Rayan Chikhi (2020). « REINDEER : efficient indexing of k-mer presence and abundance in sequencing datasets ». en. In : *Bioinformatics* 36.Supplement_1, p. i177-i185. doi : 10.1093/bioinformatics/btaa487.

Margulies, Marcel et al. (2005). « Genome sequencing in microfabricated high-density picolitre reactors ». en. In : *Nature* 437.7057, p. 376-380. doi : 10.1038/nature03959.

Marin-Acevedo, Julian A., Aixa E. Soyano, Bhagirathbhai Dholaria, Keith L. Knutson et Yanyan Lou (2018). « Cancer immunotherapy beyond immune checkpoint inhibitors ». en. In : *Journal of Hematology & Oncology* 11.1, p. 8. doi : 10.1186/s13045-017-0552-6.

McCallum, Andrew Kachites (2002). « MALLET : A Machine Learning for Language Toolkit ». http://mallet.cs.umass.edu.

McCarthy, Davis J., Yunshun Chen et Gordon K. Smyth (jan. 2012). « Differential expression analysis of multifactor RNA-Seq experiments with respect to biological variation ». In : *Nucleic Acids Research* 40.10, p. 4288-4297. doi : 10.1093/nar/gks042. eprint : https://academic.oup.com/nar/article-pdf/40/10/4288/25335174/gks042.pdf.

McCulloch, Warren S. et Walter Pitts (1943). « A logical calculus of the ideas immanent in nervous activity ». en. In : *The bulletin of mathematical biophysics* 5.4, p. 115-133. doi : 10.1007/BF02478259.

Michiels, Stefan, Serge Koscielny et Catherine Hill (2005). « Prediction of cancer outcome with microarrays : a multiple random validation strategy ». eng. In : *Lancet (London, England)* 365.9458, p. 488-492. doi : 10.1016/S0140-6736(05)17866-0.

Minsky, Marvin et Seymour Papert (1969). « Perceptrons; an introduction to computational geometry. 1969 ». In : *ISBN : 9780262534772 (citado en página 200)*, p. 292.

Mogensen, Ulla B, Hemant Ishwaran et Thomas A Gerds (2012). « Evaluating Random Forests for Survival Analysis using Prediction Error Curves ». In : *Journal of statistical software* 50.11, p. 1-23.

Nelson, Wayne (1972). « Theory and Applications of Hazard Plotting for Censored Failure Data ». In : *Technometrics* 14.4, p. 945-966. doi : 10.1080/00401706.1972.10488991.

Nunes, Tiago, David Campos, Sérgio Matos et José Luís Oliveira (2013). « BeCAS : biomedical concept recognition services and visualization ». en. In : *Bioinformatics* 29.15, p. 1915-1916. doi : 10.1093/bioinformatics/btt317.

O'Quigley, John et Fabienne Pessione (1989). « Score Tests for Homogeneity of Regression Effect in the Proportional Hazards Model ». en. In : *Biometrics* 45.1, p. 135. doi : 10.2307/2532040.

Pardoll, Drew M. (2012). « The blockade of immune checkpoints in cancer immunotherapy ». en. In : *Nature Reviews Cancer* 12.4, p. 252-264. doi : 10.1038/nrc3239.

Ramlau-Hansen, Henrik (1983). « Smoothing Counting Process Intensities by Means of Kernel Functions ». In : *The Annals of Statistics* 11.2, p. 453-466.

Rapaport, Franck, Raya Khanin, Yupu Liang, Mono Pirun, Azra Krek, Paul Zumbo, Christopher E. Mason, Nicholas D. Socci et Doron Betel (2013). « Comprehensive evaluation of differential gene expression analysis methods for RNA-seq data ». In : *Genome Biology* 14.9, p. 3158. doi : 10.1186/gb-2013-14-9-r95.

Rau, Andrea, Cathy Maugis-Rabusseau, Marie-Laure Martin-Magniette et Gilles Celeux (2015). « Co-expression analysis of high-throughput transcriptome sequencing data with Poisson mixture models ». eng. In : *Bioinformatics (Oxford, England)* 31.9, p. 1420-1427. doi : 10.1093/bioinformatics/btu845.

Robles, José A., Sumaira E. Qureshi, Stuart J. Stephen, Susan R. Wilson, Conrad J. Burden et Jennifer M. Taylor (2012). « Efficient experimental design and analysis strategies for the detection of differential expression using RNA-Sequencing ». eng. In : *BMC genomics* 13, p. 484. doi : 10.1186/1471-2164-13-484.

Roblin, E., P.-H. Cournède et S. Michiels (2020). « On the Use of Neural Networks with Censored Time-to-Event Data ». In : *Proceedings of IMSCO 2020*. Sous la dir. de G. Bebis. LNBI.

Rodrigo, Hansapani et Chris P. Tsokos (2017). « Artificial Neural Network Model for Predicting Lung Cancer Survival ». en. In : *Journal of Data Analysis and Information Processing* 05, p. 33. doi : 10.4236/jdaip.2017.51003.

Rosenblatt, F. (1958). « The perceptron : A probabilistic model for information storage and organization in the brain ». In : *Psychological Review* 65.6, p. 386-408. doi : 10.1037/h0042519.

Rumelhart, D, P Smolensky, J McCelland et G Hinton (1986). « Parallel Distributed Processing : An Exploration in the Microstructure of Cognition, Volume 2 : Psychological and Biological Models ». In : *MIT Press*.

Saldana, Diego Franco et Yang Feng (2018). « SIS : An R Package for Sure Independence Screening in Ultrahigh-Dimensional Statistical Models ». en. In : *Journal of Statistical Software* 83.2. doi : 10.18637/jss.v083.i02.

Sanger, F., S. Nicklen et A. R. Coulson (1977). « DNA sequencing with chain-terminating inhibitors ». en. In : *Proceedings of the National Academy of Sciences* 74.12, p. 5463-5467. doi : 10.1073/pnas.74.12.5463.

Schena, Mark, Dari Shalon, Ronald W. Davis et Patrick O. Brown (1995). « Quantitative Monitoring of Gene Expression Patterns with a Complementary DNA Microarray ». en. In : *Science* 270.5235, p. 467-470. doi : 10.1126/science.270.5235.467.

Schoenfeld, David (1980). « Chi-squared goodness-of-fit tests for the proportional hazards regression model ». en. In : *Biometrika* 67.1, p. 145-153. doi : 10.1093/biomet/67.1.145.

– (1982). « Partial residuals for the proportional hazards regression model ». en. In : *Biometrika* 69.1, p. 239-241. doi : 10.1093/biomet/69.1.239.

Searls, David B. (2000). « Bioinformatics Tools for Whole Genomes ». In : *Annual Review of Genomics and Human Genetics* 1.1, p. 251-279. doi : 10.1146/annurev.genom.1.1.251.

Seyednasrollah, Fatemeh, Asta Laiho et Laura L. Elo (2015). « Comparison of software packages for detecting differential expression in RNA-seq studies ». en. In : *Briefings in Bioinformatics* 16.1, p. 59-70. doi : 10.1093/bib/bbt086.

Sharma, Padmanee et James P. Allison (2015). « The future of immune checkpoint therapy ». en. In : *Science* 348.6230, p. 56-61. doi : 10.1126/science.aaa8172.

Si, Yaqing, Peng Liu, Pinghua Li et Thomas P. Brutnell (2014). « Model-based clustering for RNA-seq data ». eng. In : *Bioinformatics (Oxford, England)* 30.2, p. 197-205. doi : 10.1093/bioinformatics/btt632.

Smyth, G. K. (2005). « limma : Linear Models for Microarray Data ». en. In : *Bioinformatics and Computational Biology Solutions Using R and Bioconductor*. Sous la dir. de Robert Gentleman, Vincent J. Carey, Wolfgang Huber, Rafael A. Irizarry et Sandrine Dudoit. Statistics for Biology and Health. Springer New York, p. 397-420.

Sondka, Zbyslaw, Sally Bamford, Charlotte G. Cole, Sari A. Ward, Ian Dunham et Simon A. Forbes (2018). « The COSMIC Cancer Gene Census : describing genetic dysfunction across all human cancers ». en. In : *Nature Reviews Cancer* 18.11, p. 696-705. doi : 10.1038/s41568-018-0060-1.

Soneson, Charlotte et Mauro Delorenzi (2013). « A comparison of methods for differential expression analysis of RNA-seq data ». In : *BMC Bioinformatics* 14, p. 91. doi : 10.1186/1471-2105-14-91.

Southern, Edwin, Kalim Mir et Mikhail Shchepinov (1999). « Molecular interactions on microarrays ». en. In : *Nature Genetics* 21.1, p. 5-9. doi : 10.1038/4429.

Stelzer, Gil et al. (2016). « The GeneCards Suite : From Gene Data Mining to Disease Genome Sequence Analyses ». en. In : *Current Protocols in Bioinformatics* 54.1, p. 1.30.1-1.30.33. doi : https://doi.org/10.1002/cpbi.5.

Street, W. Nick (1998). « A Neural Network Model for Prognostic Prediction ». In : *Proceedings of the Fifteenth International Conference on Machine Learning*. ICML '98. San Francisco, CA, USA : Morgan Kaufmann Publishers Inc., p. 540-546.

Tanabe, Lorraine, Natalie Xie, Lynne H. Thom, Wayne Matten et W. John Wilbur (2005). « GENETAG : a tagged corpus for gene/protein named entity recognition ». en. In : *BMC Bioinformatics* 6.1, S3. doi : 10.1186/1471-2105-6-S1-S3.

Therneau, Terry M. et Patricia M. Grambsch (2000). *Modeling Survival Data : Extending the Cox Model*. en. Sous la dir. de K. Dietz, M. Gail, K. Krickeberg, J. Samet et A. Tsiatis. Statistics for Biology and Health. New York, NY : Springer New York. doi : 10.1007/978-1-4757-3294-8.

Therneau, Terry M, Patricia M Grambsch et Thomas R Fleming (1990). « Martingale-Based Residuals for Survival Models ». en. In : p. 15.

Tian, Lu, Ash Alizadeh, Andrew Gentles et Robert Tibshirani (2012). « A Simple Method for Detecting Interactions between a Treatment and a Large Number of Covariates ». In : *arXiv :1212.2995 [stat]*. arXiv : 1212.2995.

Tibshirani, Robert (1996). « Regression Shrinkage and Selection via the Lasso ». In : *Journal of the Royal Statistical Society. Series B (Methodological)* 58.1, p. 267-288.

– (1997). « The Lasso Method for Variable Selection in the Cox Model ». en. In : *Statistics in Medicine* 16.4, p. 385-395. doi : 10.1002/(SICI)1097-0258(19970228)16:4<385::AID-SIM380>3.0.CO;2-3.

Tibshirani, Robert, Michael Saunders, Saharon Rosset, Ji Zhu et Keith Knight (2005). « Sparsity and smoothness via the fused lasso ». en. In : *Journal of the Royal Statistical Society : Series B (Statistical Methodology)* 67.1, p. 91-108. doi : 10.1111/j.1467-9868.2005.00490.x.

Tronik-Le Roux, Diana, Mathilde Sautreuil, Mahmoud Bentriou, Jérôme Vérine, Maria Belén Palma, Marina Daouya, Fatiha Bouhidel, Sarah Lemler, Joel LeMaoult, François Desgrandchamps, Paul-Henry Cournède et Edgardo D. Carosella (2020). « Comprehensive landscape of immune-checkpoints uncovered in clear cell renal cell carcinoma reveals new and emerging therapeutic targets ». en. In : *Cancer Immunology, Immunotherapy*. doi : 10.1007/s00262-020-02530-x.

Tsuruoka, Yoshimasa, Yuka Tateishi, Jin-Dong Kim, Tomoko Ohta, John McNaught, Sophia Ananiadou et Jun'ichi Tsujii (2005). « Developing a Robust Part-of-Speech Tagger for Biomedical Text ». en. In : *Advances in Informatics*. Sous la dir. de Panayiotis Bozanis et Elias N. Houstis. Lecture Notes in Computer Science. Berlin, Heidelberg : Springer, p. 382-392. doi : 10.1007/11573036_36.

Vano, Y A, S Oudard et N Giraldo (2015). « Cancer du rein à cellules claires : pourquoi les inhibiteurs de checkpoints sont-ils efficaces ? La biologie ». fr. In : p. 5.

Vapnik, Vladimir Naumovich (1998). *Statistical learning theory*. Adaptive and learning systems for signal processing, communications, and control. New York : Wiley.

Venter, J. Craig et al. (2001). « The Sequence of the Human Genome ». en. In : *Science* 291.5507, p. 1304-1351. doi : 10.1126/science.1058040.

Verweij, Pierre J. M. et Hans C. Van Houwelingen (1994). « Penalized likelihood in Cox regression ». en. In : *Statistics in Medicine* 13.23-24, p. 2427-2436. doi : 10.1002/sim.4780132307.

Wei, L. J. (1984). « Testing Goodness of Fit for Proportional Hazards Model with Censored Observations ». In : *Journal of the American Statistical Association* 79.387, p. 649-652. doi : 10.2307/2288412.

Whittaker, R. H. (1960). « Vegetation of the Siskiyou Mountains, Oregon and California ». en. In : *Ecological Monographs* 30.3, p. 279-338. doi : 10.2307/1943563.

– (1972). « Evolution and Measurement of Species Diversity ». en. In : *TAXON* 21.2-3, p. 213-251. doi : 10.2307/1218190.

Wu, Yichao (2012). « Elastic net for Cox's proportional hazards model with a solution path algorithm ». In : *Statistica Sinica* 22, p. 27.

Yuan, Ming et Yi Lin (2006). « Model selection and estimation in regression with grouped variables ». en. In : *Journal of the Royal Statistical Society : Series B (Statistical Methodology)* 68.1, p. 49-67. doi : 10.1111/j.1467-9868.2005.00532.x.

Zhang, H. H. et W. Lu (2007). « Adaptive Lasso for Cox's proportional hazards model ». en. In : *Biometrika* 94.3, p. 691-703. doi : 10.1093/biomet/asm037.

Zhao, Lili et Dai Feng (mar. 2020). « DNNSurv : Deep Neural Networks for Survival Analysis Using Pseudo Values ». In : *arXiv :1908.02337 [cs, stat]*. arXiv : 1908.02337.

Zhao, Sihai Dave et Yi Li (2012). « Principled sure independence screening for Cox models with ultra-high-dimensional covariates ». en. In : *Journal of Multivariate Analysis* 105.1, p. 397-411. doi : 10.1016/j.jmva.2011.08.002.

Zou, Hui (2006). « The Adaptive Lasso and Its Oracle Properties ». en. In : *Journal of the American Statistical Association* 101.476, p. 1418-1429. doi : 10.1198/016214506000000735.

Zou, Hui et Trevor Hastie (2005). « Regularization and variable selection via the elastic net ». en. In : *Journal of the Royal Statistical Society : Series B (Statistical Methodology)* 67.2, p. 301-320. doi : 10.1111/j.1467-9868.2005.00503.x.

université PARIS-SACLAY
ÉCOLE DOCTORALE
Interfaces:
matériaux, systèmes, usages

Titre : Contributions à la détection de marqueurs et à l'analyse de survie en oncologie

Mots clés : Analyse de survie, détection de marqueurs, réseaux de neurones, grande dimension, cancer

Résumé : La médecine personnalisée en oncologie permet d'adapter les traitements aux caractéristiques des patients. L'utilisation des données d'expression de gènes comme caractéristiques amène de nouvelles problématiques : la grande dimension. L'objectif de cette thèse est d'étudier et développer des méthodes adaptées à la grande dimension pour la détection de marqueurs et l'analyse de survie en oncologie. Dans une première partie de ce travail, nous nous intéressons à la détection de marqueurs en oncologie avec deux objectifs différents. Le premier objectif consiste à identifier les gènes signatures du cancer du rein à cellules claires (ccRCC). Nous réalisons, dans un premier temps, une analyse différentielle et nous couplons, par la suite, une sélection de variables issue de l'analyse différentielle avec une méthode d'apprentissage. Le second objectif de cette partie est d'étudier les méthodes de régularisation et de Screening pour mettre en évidence les gènes influençant la survie des patients. La stabilité de ces méthodes a également été étudiée à partir d'un indice de similarité. Dans la seconde partie de cette thèse, nous nous intéressons à la prédiction de la survie en grande dimension. Nous avons étudié l'apport des réseaux de neurones dans ce contexte. Ces méthodes ont été peu étudiées en analyse de survie en grande dimension. Nous comparons deux approches de réseaux de neurones : une basée sur le modèle de Cox et une autre basée sur un modèle à temps discret. Nous nous sommes concentrés sur celle-ci en l'adaptant à la grande dimension. Ensuite, une étude de comparaison est réalisée afin d'évaluer les performances de ces deux approches et la prédiction à partir du modèle de Cox avec une procédure d'estimation de type Lasso est prise comme référence. Un plan de simulation a été créé en prenant en compte différents modèles de survie pour générer des données avec différents niveaux de complexité. La sparsité et la censure sont également prises en compte. Les performances sont donc évaluées à partir de deux métriques différentes (C-index et IBS) sur ces données simulées et illustrées sur des jeux de données réelles.

Title: Contributions to marker detection and survival analysis in oncology

Keywords: Survival analysis, marker detection, neural network, high-dimension, cancer

Abstract: Personalized medicine plays an important role in oncology, it enables to adapt treatments to the characteristics of patients. The use of gene expression data to characterize the patients raises new issues related to high-dimensional statistics. The objective of this thesis consists in studying and developing methods adapted to the high-dimension for marker detection and survival analysis in oncology. In the first part of this work, we are interested in marker detection with two different objectives. The first one consists in identifying the genes responsible of Clear Cell Renal Cell Carcinoma (ccRCC). First, we realize a differential analysis. Secondly, we couple a variable selection obtained from a differential analysis with a learning method. The second objective of this part consists in studying regularization and screening methods to underline the genes impacting the survival of patients. The stability of these methods is also studied with a similarity index. In the second part of this thesis, we are interested in predicting survival in high-dimension. We study the potential of neural networks in this context. We distinguish two strategies for neural networks: one based on the Cox model and another one based on a discrete-time model. As this last one is less studied, we focus on neural networks based on this strategy and we have adapted it to the high-dimensional setting. We present a comparison study to observe the impact of different models in survival analysis in the context of high-dimension. We create a simulation plan to make this comparison more relevant and the data are simulated with different survival models to have data of different complexity levels. We also study the effect of censorship and sparsity. The performances of these methods are evaluated with the Concordance index and the Integrated Brier Score on this simulated data and on real datasets.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France