Forêts aléatoires et sélection de variables : analyse des données des enregistreurs de vol pour la sécurité aérienne

Baptiste Gregorutti

École Doctorale de Science Mathématiques de Paris Centre

Thèse de doctorat

Discipline : Mathématiques
Spécialité : Statistique

présentée par

Baptiste Gregorutti

Forêts aléatoires et sélection de variables : analyse des données des enregistreurs de vol pour la sécurité aérienne

---

Soutenue le 11 mars 2015 devant le jury composé de :

M. Gérard Biau	Univ. Pierre et Marie Curie	directeur
M. Michel Broniatowski	Univ. Pierre et Marie Curie	examinateur
M. Bertrand Iooss	EDF R&D	rapporteur
M. Pierre Jouniaux	Safety Line	encadrant
M. Bertrand Michel	Univ. Pierre et Marie Curie	co-directeur
M^{me} Florence Nicol	ENAC	examinateur
M. Jean-Michel Poggi	Univ. Paris-Sud Orsay	rapporteur
M. Fabrice Rossi	Univ. Paris 1 Panthéon-Sorbonne	examinateur
M. Philippe Saint Pierre	Univ. Pierre et Marie Curie	co-directeur

Laboratoire de statistique théorique et appliquée (LSTA)
Université Pierre et Marie Curie (Paris 6)
Boîte 158, Tours 15-25, 2ème étage
4 place Jussieu
75252 Paris Cedex 05

Safety Line
15 rue Jean-Baptiste Berlier
75013 Paris

École Doctorale de Sciences Mathématiques de Paris Centre
Université Pierre et Marie Curie (Paris 6)
Boîte 290, Tours 15-25, 1er étage
4 place Jussieu
75252 Paris Cedex 05

Remerciements

L'environnement dans lequel j'ai évolué durant ces trois années de thèse a été idéal. C'est sans aucun doute grâce à la disponibilité de mes directeurs de thèse ainsi qu'à l'intérêt qu'ils ont porté à mon travail et à Safety Line.

Je remercie tout d'abord Gérard qui a bien voulu m'accepter parmi ses thésards chanceux. Il a su être présent à chaque fois que j'en avais besoin, et ce malgré ses diverses responsabilités, notamment à la tête du LSTA. Ses conseils précieux et ses remarques franches m'ont permis de valoriser mon travail et ont été d'une grande aide lors de la rédaction de ce manuscrit.

Je suis sincèrement reconnaissant envers Bertrand et Philippe qui ont su me guider au quotidien, notamment dans la façon d'aborder le problème difficile qui m'était posé. Je les remercie pour les réunions constructives autour de la machine à café où ils m'ont appris la rédaction d'un article de recherche, chose qui ne m'était pas acquise au départ. Gérard, Bertrand et Philippe, je peux affirmer que, grâce à vous, je suis fier de ce manuscrit. J'espère avoir l'occasion de continuer à travailler avec vous dans le futur.

Une thèse CIFRE est le fruit d'un partenariat entre un laboratoire de recherche et une entreprise. La réussite de cette thèse est de ce fait également due aux conditions que m'a donné Pierre au sein de Safety Line. Merci, Pierre, pour m'avoir fait confiance et m'avoir permis de participer, à mon humble niveau, au développement de Safety Line. J'ai tout particulièrement apprécié travailler avec toi dans les locaux de 10 m^2 à l'incubateur de Telecom Paristech : une table, une chaise et un ordinateur portable, nous n'avions besoin de rien de plus... Durant ces trois années, j'ai pu voir de près ce qu'est le fonctionnement d'une startup et les nombreuses difficultés rencontrées. Malgré les embuches, tu as su garder le cap – trait de caractère tout à fait cohérent avec ta formation de pilote de ligne – et tu as su mené Safety Line là où elle est actuellement. Tu m'as beaucoup appris, tant humainement que professionnellement. Pour tout cela, je te remercie.

Je remercie également Jean-Michel Poggi et Bertrand Iooss pour m'avoir fait l'honneur de rapporter ma thèse. Merci pour votre lecture attentive, pour vos remarques constructive ainsi que pour l'intérêt que vous portez à mes travaux.

Merci à Michel Broniatowski, Florence Nicol et Fabrice Rossi pour avoir accepté de participer à mon jury. Je suis particulièrement reconnaissant envers Michel Broniatowski de m'avoir orienté vers Gérard au moment où j'étais à la recherche d'un sujet de thèse.

Je tiens à remercier toutes les personnes que j'ai côtoyé au LSTA pendant ces trois années et particulièrement les doctorants. Un grand merci aux "Biau-men", Benjamin, Clément et Erwan pour les excellents moments passés ensemble. Merci à Cécile pour son

humour sans failles. Merci aux actuels résidents du bureau 204, Assia, Roxane, Quyen, Matthieu et les anciens Sarah L., Sarah O., Aurélie, Manu, Virgile et Abdullah. Plus généralement, je remercie Mokhtar, Amadou, Alexis, Svetlana, Boris ainsi que Jean-Patrick, Agathe, Stéphane, Olivier L., Olivier W., Étienne, Fanny sans oublier Louise, Corinne et Pascal.

J'adresse mes sincères remerciements aux collègues, qui contribuent à la formidable ambiance qui règne chez Safety Line. Merci à Karim, Éric, Claude, Kristof (pour avoir eu l'idée d'acheter des mini paniers de basket), Cyrille, Saad, Marouane, Anthony, Ana et Bertrand. Une mention toute particulière à mes voisins de bureau, indéniablement les meilleurs, Begaiym, Cindie et Robin-Marie-Camille. Un énorme Big Up à Nora, chef com', pour sa bonne humeur et pour les parties de basket endiablées. Je pense aussi à toutes les personnes que j'ai rencontrées depuis trois ans, notamment Didier, Pascal, Sébastien, David, Anne-Claire, Alexis, Melissa et Julie.

Pour conclure, je souhaite remercier ma famille et mes amis pour leur soutien. Merci à mes parents et à mes frangines Anaëlle et Jeanne pour leurs encouragements. Enfin, je remercie Marie pour son amour. Merci d'avoir accepté de mettre de côté notre vie sociale pendant les longs mois de la rédaction de ma thèse. Merci de m'avoir toujours soutenu dans les moments difficiles. Merci pour tout ce que nous avons partagé ensemble ces huit dernières années et pour ce que nous partagerons ensemble dans le futur.

Table des matières

Avant-propos

Cette thèse CIFRE est le fruit d'un partenariat entre le Laboratoire de Statistique Théorique et Appliquée (LSTA) de l'Université Pierre et Marie Curie et la société Safety Line, éditrice de logiciels dans le domaine du transport aérien. Créée fin 2010 par d'anciens enquêteurs au Bureau d'Enquêtes et d'Analyses pour la Sécurité de l'Aviation Civile (BEA), Safety Line propose à ses clients des solutions logicielles innovantes dédiées à la gestion des risques opérationnels et à l'analyse des données aéronautiques.

Avertissement

Les données présentées dans ce rapport sont anonymisées afin de préserver la confidentialité des clients de Safety Line. C'est pourquoi, certaines informations (compagnie, numéros de vol, types d'avion, aéroports de destination, date, heure, etc.) ne seront pas spécifiées.

Chapitre 1

Introduction

Résumé. Ce chapitre introductif a pour objectif de présenter le contexte industriel ainsi que les motivations de ces travaux de thèse. Nous décrivons en premier lieu les données dont nous disposons, issues des enregistreurs de vol. Nous explicitons ensuite les problématiques statistiques qui découlent des objectifs opérationnels et détaillons pour finir nos contributions.

Sommaire

1.1 Contexte général et enjeux de la thèse

Dans un contexte de sécurité élevé, l'Organisation de l'Aviation Civile Internationale (OACI [1]) a récemment défini un ensemble de recommandations visant à réduire davantage le nombre d'accidents en vol. Chaque État doit désormais exiger des exploitants aériens et des organismes de maintenance qu'ils mettent en place des Systèmes de Gestion de la Sécurité (SGS) afin d'évaluer précisément les risques opérationnels. Ainsi, les compagnies aériennes sont depuis 2006 dans l'obligation de passer progressivement d'un modèle basé uniquement sur le respect des normes (la conformité) à un modèle de gestion des risques tel que l'a défini l'OACI.

1. Un glossaire des termes aéronautiques est donné en Annexe A.1.

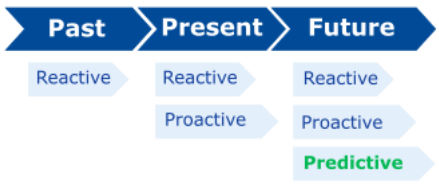


FIGURE 1.1 – Les différents modes d'un Système de Gestion de la Sécurité (source : IATA Safety Report 2013[2]).

Un SGS se décline en trois modes. Le mode réactif consiste à mener des enquêtes à la suite d'incidents survenus durant le vol afin d'en connaître les causes, puis à mettre en place des actions correctives. Le mode proactif se définit comme la connaissance des dangers et des défenses associées afin d'anticiper des situations à risque. Enfin, la démarche prédictive, complémentaire au réactif et au proactif, vise à développer une mesure objective des risques et de leurs origines pour être en permanence conscient du niveau de sécurité de la compagnie.

Pour illustrer cette démarche de la gestion des risques, considérons un aérodrome présentant des caractéristiques particulières (présence de montagnes à proximité, pistes courtes, ...). Le mode proactif d'un SGS doit alors identifier les dangers liés à ces caractéristiques pour permettre aux compagnies de former les pilotes et de mettre en place des procédures adaptées. Si un incident survient sur cet aérodrome, le mode réactif doit en identifier les causes pour éviter qu'il ne se reproduise dans le futur. Le volet prédictif se résume à mesurer le niveau de risque en tenant compte des caractéristiques de l'aérodrome et à suivre son évolution dans le temps, et ce afin d'identifier des causes d'incidents potentiels.

Si les volets réactifs et proactifs sont bien intégrés par les compagnies aériennes, le mode prédictif n'est à ce jour pas mis en œuvre. Le rapport IATA 2013 sur la sécurité [2] souligne, en effet, que l'analyse prédictive est la prochaine étape pour compléter le SGS (Figure 1.1).

Pour atteindre cet objectif, il est nécessaire d'exploiter les données des enregistreurs de vol, les boîtes noires. Elles sont recueillies par les compagnies après chaque vol et sont à ce jour uniquement exploitées pour s'assurer du respect de la conformité. Pour cela, les gestionnaires de la sécurité des compagnies aériennes utilisent l'*analyse des vols* (en anglais *Flight Data Monitoring*, FDM), outil permettant de détecter des événements survenus au cours des vols. Plus précisément, ces événements sont définis par des dépassements de seuils fixés par la compagnie sur certains paramètres préalablement choisis. Par exemple, l'événement "Speed High during Approach at 500 ft" s'intéresse à une vitesse excessive à l'altitude de 500 pieds lorsque l'avion est en phase d'approche (soit une minute environ avant l'atterrissage).

L'information à laquelle donne accès l'*analyse des vols* est fortement réduite car elle correspond uniquement aux dépassements de seuils. De deux choses l'une : soit un événement est détecté, et dans ce cas, le vol est analysé, soit aucun dépassement de seuil n'est détecté et les données ne sont pas conservées. De plus, les analyses sont effectuées vol par vol et ne contiennent pas d'informations sur le niveau de risque à l'échelle de la flotte. L'accident du vol Air France AF447 survenu en 2009 entre Rio et Paris illustre bien les

2. http ://www.iata.org/publications/Documents/iata-safety-report-2013.pdf

limites de cette démarche. Le Bureau d'Enquêtes et d'Analyses (BEA) a montré que la catastrophe a en partie été causée par le givrage des sondes Pitot, instruments mesurant la pression totale de l'air pour, notamment, calculer la vitesse de l'avion. Or, plusieurs cas de givrage des sondes Pitot avaient été relevés avant l'accident mais le risque n'avait pas été correctement évalué. Le respect de la conformité n'est donc pas une démarche suffisante dans un objectif de gestion des risques dans le mesure où cette approche ne permet pas d'identifier les risques et de les anticiper.

Le niveau de sécurité atteint aujourd'hui par le transport aérien est avant tout le fruit du respect de normes très strictes. Ces normes définissent un ensemble de marges de sécurité qui permettent d'évoluer dans une "zone" sure d'exploitation. La sécurité se définit alors comme le respect des marges de sécurité. Il faut donc savoir comment les marges sont utilisées afin de s'assurer du niveau de sécurité et de le maintenir. Plus les marges sont réduites, plus le risque est important. Par exemple, lors de l'atterrissage, il est demandé aux compagnies de vérifier que les performances de l'avion permettent l'arrêt sur moins de 60 % de la piste. Les 40 % restants définissent une marge de sécurité pour éviter les sorties de piste. Plus on s'éloigne du seuil de 60 % de la longueur de piste, plus le risque de sortie de piste est élevé.

L'enjeu de cette thèse est de faire évoluer l'analyse des données des enregistreurs de vol par l'introduction de méthodes statistiques. Des situations à risque doivent être identifiées en mesurant l'utilisation de marges de sécurité. L'analyse prédictive des risques aériens revient à anticiper des scénarios critiques par la mise en place d'actions correctives (formations, nouvelles procédures, ...). Dans cet objectif, il est crucial d'analyser tous les vols mis à disposition. De nouveaux facteurs de risque peuvent alors être identifiés. Ils permettent d'apporter une meilleure compréhension des risques et donc une meilleure vision du niveau de sécurité de la compagnie. Cette approche se différencie des solutions utilisées actuellement par les exploitants aériens car elle utilise l'ensemble de l'information contenue dans les enregistreurs de vol là où l'*analyse des vols* FDM se contente d'examiner les vols un par un. Le but final de Safety Line est de proposer, en collaboration avec les experts aéronautiques, des solutions automatiques de traitements statistiques des données à grande échelle, en complément de l'*analyse des vols*.

1.2 Données de vol

Les avions sont équipés de deux enregistreurs de vol, le *Cockpit Voice Recorder* (CVR) et le *Flight Data Recorder* (FDR, Figure 1.2). Le CVR enregistre les communications radio, les échanges entre les pilotes et les contrôleurs aériens ainsi que les bruits du poste de pilotage (alarmes, variations du régime moteur, ...). Ces enregistrements sont analysés à la suite d'événements majeurs. Ils permettent aux enquêteurs de comprendre les dysfonctionnements survenus à bord et de mettre parfois en évidence des erreurs humaines. Le FDR recueille les informations relatives aux systèmes de l'avion. Tout comme pour le CVR, ces données sont en général analysées à la suite d'incidents survenus en vol.

Le nombre de paramètres enregistrés à bord peut varier de quelques centaines à plusieurs milliers selon le type et l'âge de l'avion. Certains sont mesurés et d'autres calculés. La fréquence d'enregistrement est en général d'une mesure par seconde mais pour les avions les plus récents, l'enregistrement peut se faire tous les huitièmes de seconde. Afin de donner un aperçu des données dont nous disposons, nous présentons une liste sélectionnée des principaux paramètres de vol. Cette liste peut être consultée en Annexe A.3.

FIGURE 1.2 – Enregistreur de vol.

Vitesses et accélérations

Plusieurs types de vitesses sont considérées en aéronautique. On distingue la vitesse air (air speed) – vitesse de l'avion par rapport à la masse d'air – et la vitesse sol (ground speed, GSC) qui correspond à la vitesse de déplacement par rapport au référentiel terrestre. La différence entre la vitesse air et la vitesse sol est la composante de vent sur la trajectoire de l'avion (voir la Figure 1.3). La vitesse et l'orientation du vent sont cruciales en pilotage car elles peuvent altérer la stabilisation de l'avion notamment en phase d'approche.

La densité de l'air variant en fonction de l'altitude et de la température, différentes vitesses air sont employées. La vitesse indiquée (IAS) est donnée par la différence entre la pression totale – captée par les tubes Pitots – et la pression atmosphérique ambiante (dite pression statique). La vitesse conventionnelle (Calibrated Air Speed, CASC) correspond à la vitesse corrigée des erreurs d'instruments. En règle générale, les pilotes se réfèrent à la vitesse corrigée jusqu'à une altitude de transition où l'on contrôle la vitesse en nombre de Mach, rapport entre la vitesse vraie et la vitesse du son. La vitesse d'approche (VAPP) est une vitesse de référence que le pilote doit respecter au moment de la phase d'approche soit quelques minutes avant l'atterrissage pour conserver une marge suffisante par rapport au décrochage. Elle est calculée en fonction notamment de la masse de l'avion. Les vitesses sont exprimées en nœuds c'est-à-dire 0.514 $m.s^{-1}$. Enfin, la vitesse verticale (IVV) désigne le taux de chute et est exprimée en pieds par minute.

Les accélérations longitudinale (LONGG), latérale (LATG) et verticale (VRTG) définissent les accélérations sur les différents axes. Elles sont mesurées jusqu'à une fréquence de 8 Hz et l'unité de mesure est le g soit 9.81 $m.s^{-2}$.

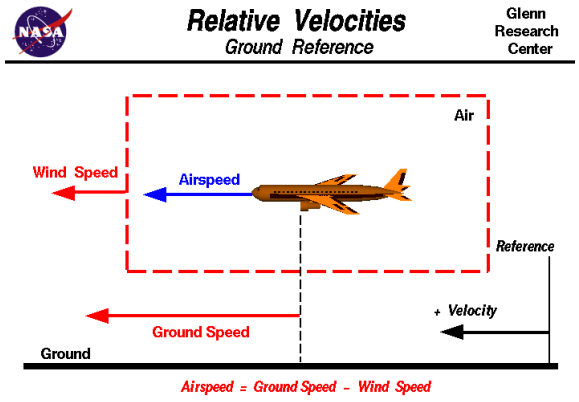


FIGURE 1.3 – Différence entre la vitesse air et la vitesse sol (source : NASA).

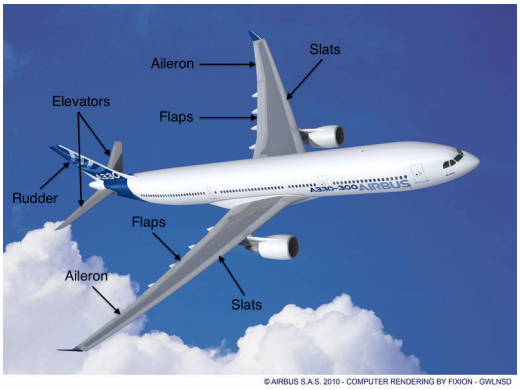


Figure 1.4 – Contrôles de direction de l'avion (source : Airbus).

Altitudes

Plusieurs enregistrements d'altitude sont disponibles. L'altitude barométrique (variable ALT_STDC) est la différence entre la pression atmosphérique autour de l'avion et celle du niveau de la mer. La radio altitude (RALTC) est la distance verticale entre l'avion et la surface qu'il survole. Elle est mesurée grâce à des capteurs positionnés sous l'avion, les radiosondes. Enfin, la hauteur (HEIGHT) est distance verticale entre l'avion et le seuil de piste. L'unité de mesure est le pied, c'est-à-dire 0.3048 mètres.

Paramètres de pilotage

La Figure 1.4 représente les différents systèmes liés aux contrôles de direction de l'avion. La position de la gouverne de direction (RUDDER) influe sur le mouvement de l'avion autour de l'axe vertical (le lacet). La position des ailerons (AILL, AILR) influe sur le roulis c'est-à-dire la rotation autour de l'axe longitudinal (ROLL). Les gouvernes de profondeur (ELEVL, ELEVR) permettent quant à elles de contrôler l'assiette, rotation de l'avion autour de l'axe latéral (PITCH).

La commande de poussée des moteurs (TLA) contrôle la puissance délivrée par les moteurs (N1 et N2). Les paramètres N1 et N2 sont exprimés en pourcentage de vitesse de rotation. Les volets (FLAPC) et les becs (SLATS) permettent au pilote d'augmenter la portance et de réduire la vitesse de décrochage au décollage et à l'atterrissage. Des paramètres binaires sont également disponibles tels que l'indicateur de sortie des trains d'atterrissage (LDGL, LDGR), l'enclenchement du pilote automatique (AP_OFF) ou de l'assistance au freinage lors de l'atterrissage (auto brake, AUTO_BRK_OFF).

Alarmes

Une grande partie des données enregistrées sont des variables binaires, parmi lesquelles se trouvent des alarmes. Elles indiquent aux pilotes des situations anormales. Citons en autres la proximité du sol (GPWS, TERRAIN, PULL_UP, SINKRATE, etc.), un angle de descente excessif (GLIDESLOPE) ou la présence de cisaillement de vent (windshear, WIN_SHR_WAR), changement brusque de la vitesse et de l'orientation du vent.

Données de géolocalisation

Parmi les données de géolocalisation, le cap magnétique (HEAD_MAG) est une mesure de l'angle entre le nord magnétique et l'axe longitudinal de l'avion. Ce paramètre est la conséquence directe de l'action du pilote. La latitude (LATP) et la longitude (LONP) sont aussi des paramètres enregistrés en vol. Ils indiquent la position en temps réel de l'aéronef et sont exprimés en degrés.

Autres paramètres

D'autres paramètres de vol relatifs à l'avion sont enregistrés. Par exemple, la masse en kilogrammes (Gross Weight, GW_KG), la quantité de carburant (Fuel Quantity, FQTYC), les aéroports d'origine, de destination ainsi que la date et l'heure du vol.

1.3 Atterrissage long et atterrissage dur

La grande majorité des incidents surviennent lors des phases de décollage et d'atterrissage. Ils peuvent engendrer des coûts matériels importants pour les compagnies aériennes. C'est pourquoi l'étude des risques liés à ces deux phases particulières est une priorité pour Safety Line. L'atterrissage long et l'atterrissage dur sont les objets d'étude principaux de ce manuscrit. L'étude de l'atterrissage long sera développée dans les Chapitres 2, 4, 5 et 6 tandis que l'atterrissage dur fera l'objet du Chapitre 7 dans un contexte beaucoup plus opérationnel. Pour chaque risque étudié, les données sont relevées durant la phase d'approche, soit dix minutes avant le toucher des trains (Figure 1.5).

Phase d'approche

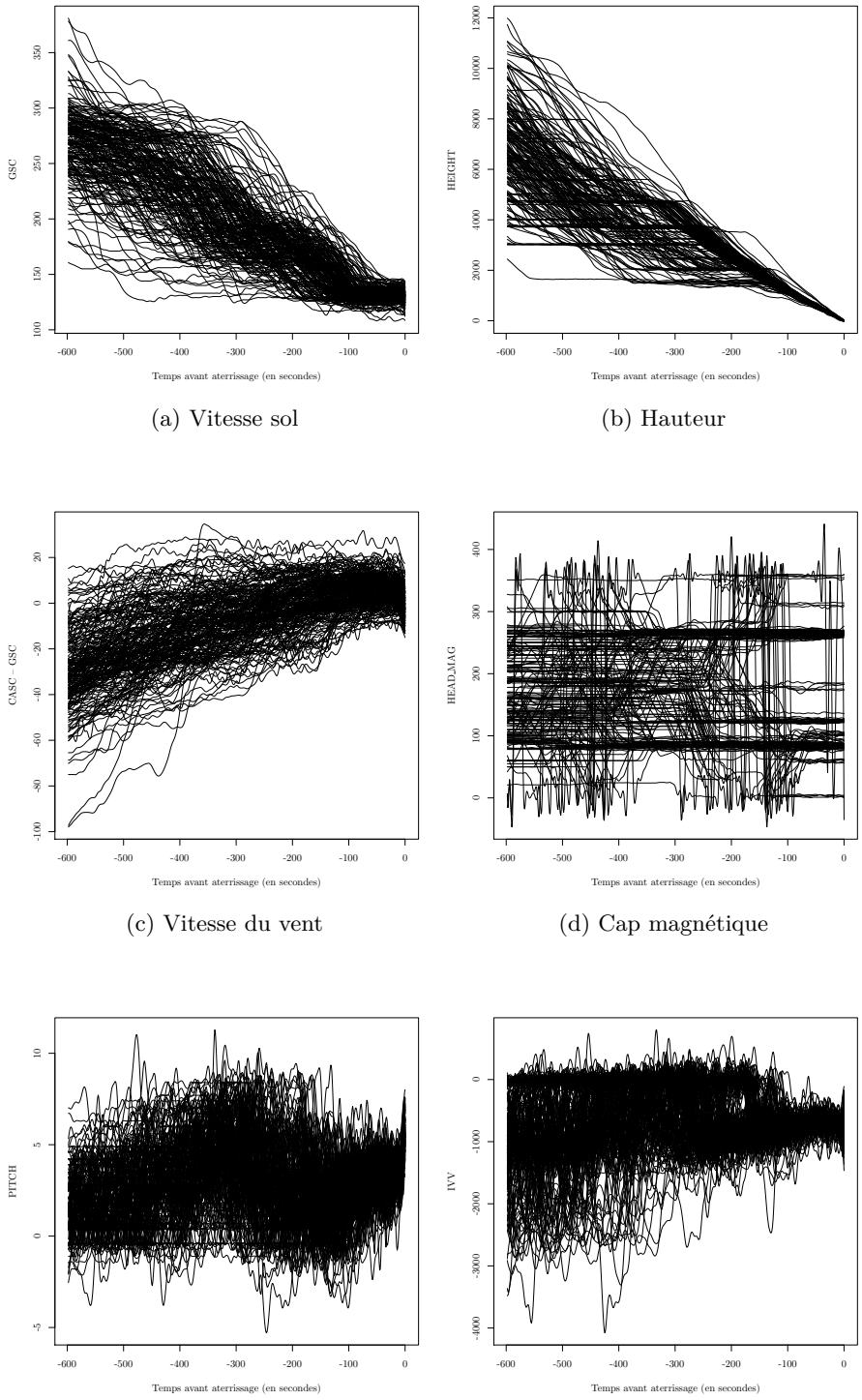
L'étude de la phase d'approche et de l'approche finale est très importante car elle peut contenir des signes précurseurs de risques potentiels à l'atterrissage. Par exemple, une vitesse excessive durant la phase d'approche peut conduire à une situation à risque à l'atterrissage.

La stabilisation de l'avion en approche est cruciale, en particulier lorsque l'altitude est entre 1000 pieds et 500 pieds. À cet instant, l'approche est dite stabilisée lorsqu'un ensemble de critères est satisfait :

- l'avion doit se trouver dans l'axe de la piste d'atterrissage et doit suivre un plan de descente à trois degrés ;
- la vitesse de l'avion doit être égale à la vitesse d'approche ;
- l'avion doit être dans la configuration de l'atterrissage (volets et trains sortis) et la puissance délivrée par les moteurs doit être supérieure au ralenti c'est-à-dire à une valeur supérieure à 50 % afin de conserver suffisamment de puissance en cas de remise de gaz ;
- la vitesse verticale de descente ne doit pas être supérieure à 1000 pieds par minute.

Si l'une de ces conditions n'est pas satisfaite, l'approche doit être interrompue et le pilote a l'obligation de remettre les gaz afin d'effectuer une nouvelle approche.

Le vent est un des facteurs connus de la non-stabilisation de l'avion. Par exemple, la présence de vent arrière ou le cisaillement de vent peuvent fortement altérer les conditions d'approche et engendrer des situations critiques à l'atterrissage.



(a) Vitesse sol

(b) Hauteur

(c) Vitesse du vent

(d) Cap magnétique

(e) Assiette

(f) Vitesse verticale

Figure 1.5 – Quelques exemples de courbes durant la phase d'approche, soit dix minutes avant l'atterrissage. Les données sont lissées pour faciliter la visualisation.

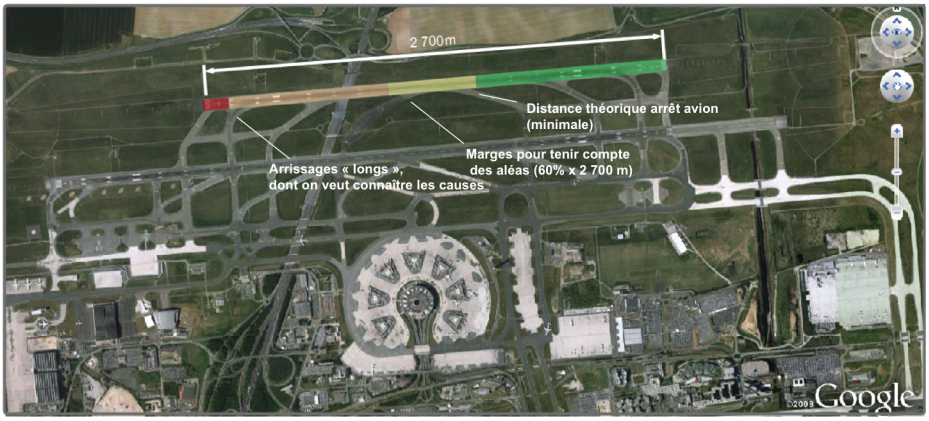


FIGURE 1.6 – Définition de l'atterrissage long (source : Google Earth).

Les paramètres de vol déterminants pendant l'approche se répartissent en trois catégories :

- Commandes de pilotage ;
- État de l'avion (vitesse, altitude, angle d'attaque, régimes moteurs et masse, etc.) ;
- Environnement (vitesse du vent, cap, positions GPS, etc.).

Atterrissage long

Les sorties de piste sont redoutées par les opérateurs aériens car elles engendrent des coûts matériels importants. Ces accidents peuvent notamment être causés par de mauvaises conditions météorologiques ou une contamination de la piste qui peut impliquer des difficultés à réduire la vitesse de l'avion. Afin d'éviter les sorties de piste, il est demandé aux compagnies de vérifier que les performances de l'avion permettent l'arrêt sur moins de 60 % de la piste en tenant compte des conditions météorologiques prévues (voir la Figure 1.6). À défaut de pouvoir observer une quantité suffisante de sorties de piste, nous nous intéressons aux atterrissages longs. Un atterrissage est considéré comme long si la distance d'arrêt de l'avion est supérieure au seuil réglementaire de 60 % de la longueur de piste. L'identification des facteurs expliquant ces atterrissages est essentielle pour anticiper sur des incidents plus graves comme les sorties de piste.

Atterrissage dur

L'atterrissage dur correspond à une vitesse verticale très importante au moment du toucher des roues et se traduit par une forte compression des trains. Comme les sorties de piste, l'atterrissage dur peut engendrer des dégâts matériels importants.

Plus précisément, un atterrissage est considéré comme dur si la vitesse verticale de l'avion au moment du toucher excède 600 pieds par minute ou si l'accélération verticale (VRTG) dépasse 2.6 g. Ces seuils permettent de détecter le moment où les trains d'atterrissage nécessitent des opérations de maintenance. En pratique, d'autres seuils sont utilisés par les compagnies aériennes pour identifier différents niveaux d'atterrissage :

- Niveau 1 si VRTG $>$ 1.4 g ;
- Niveau 2 si VRTG $>$ 1.8 g ;

- Niveau 3 si VRTG $>$ 2.1 g.

Ces seuils étant fixés a priori, ils ne permettent pas de décrire finement le phénomène de l'atterrissage dur. L'approche que nous retenons consiste à mesurer l'intensité de l'atterrissage par la valeur de l'accélération verticale au toucher des roues sans fixer de seuil *a priori*.

Jeux de données utilisés

Les données étudiées pendant la thèse proviennent de deux compagnies et de deux types d'avion très différents. Nous analysons en effet des moyen-courriers d'une part et des long-courriers d'autre part. Les différences sont notamment les suivantes :

- Les avions moyen-courriers ont une masse plus faible que les long-courriers et sont plus maniables. À l'atterrissage, la masse est déterminante car elle influe sur la décélération de l'avion.
- La façon de piloter et les procédures de vol ne sont pas les mêmes.
- Le rapport poids/puissance est différent.

Quatre jeux de données sont extraits des enregistreurs de vol. Ils correspondent chacun à une compagnie et à un type d'avion (informations non précisées ici). Tous les paramètres de vol ne sont pas pris en compte. En effet, l'expertise métier est en mesure d'identifier ceux qui sont les moins informatifs pour le risque considéré. Les paramètres relatifs au décollage sont donc ici automatiquement supprimés.

Nous décrivons dans les paragraphes qui suivent les différents jeux de données exploités pendant la thèse. Les trois premières bases de données sont relatives au risque d'atterrissage long et la quatrième correspond à l'étude de l'atterrissage dur.

Données `Class1`. La base de données `Class1` est employée pour la détection des atterrissages longs. Les avions sont des moyen-courriers et les vols sont répartis sur plusieurs aéroports de destination. Nous observons uniquement les paramètres à **l'altitude de 500 pieds** (soit environ une minute avant l'atterrissage) pour la prédiction des classes d'atterrissage : un vol est labellisé 1 si sa distance d'atterrissage est supérieure au seuil de 60 % de la longueur de la piste et -1 sinon. La base de données est composée de :

- 10 819 vols dont 10 % de classe 1 ;
- 89 paramètres de vol sur 250 dont 49 variables binaires.

Données `Class2`. Une seconde base de données est employée pour la détection des atterrissages longs. Contrairement au jeu de données précédent, les enregistrements sont **temporels** et sont échantillonnés chaque seconde durant les dix dernières minutes avant l'atterrissage (Figure 1.5). D'autre part, les données sont issues d'avions long-courriers et correspondent à un aéroport de destination (non communiqué ici). La base de données est composée de :

- 254 vols dont 45 % de classe 1 ;
- 22 paramètres quantitatifs sur 184.

Les signaux sont recalées sur le point de toucher des trains, information donnée par les paramètres binaires LDGL et LDGR.

Données `Reg1`. Le jeu de données `Reg1` est employé pour la prédiction des distances d'atterrissage. Comme le jeu de données précédent, les enregistrements sont **temporels** et sont échantillonnés chaque seconde durant la phase d'approche. D'autre part, les données sont issues d'avions moyen-courriers et correspondent à un aéroport de destination. La base de données est composée de :

- 1868 vols ;
- 23 paramètres quantitatifs sur 250.

Les signaux sont recalées sur le point de toucher des trains, information donnée par les paramètres binaires LDGL et LDGR.

Données `Reg2`. Le jeu de données `Reg2` est employé pour l'analyse des atterrissages durs où l'accélération verticale au point de toucher est à prédire. Contrairement aux bases de données précédentes, les enregistrements sont échantillonnées chaque seconde durant **la minute qui précède l'atterrissage et quelques secondes ensuite**. D'autre part, les données sont issues d'avions long-courriers et correspondent à un aéroport de destination. La base de données est composée de :

- 699 vols ;
- 29 paramètres quantitatifs sur 184 dont 4 observés uniquement au point de toucher.

Les signaux sont recalées sur le point de toucher des trains. Il a été choisi d'extraire quelques secondes après l'atterrissage afin de tenir compte d'éventuels rebonds et de la décélération de l'avion.

1.4 Contexte statistique

Ce travail de thèse vise à proposer des outils méthodologiques d'analyse statistique des enregistreurs de vol. Nous souhaitons *in fine* présenter aux compagnies aériennes une approche nouvelle de la gestion des risques opérationnels. En particulier, l'identification des facteurs de risque est nécessaire car elle permet de mieux comprendre l'utilisation des marges de sécurité. Ces facteurs peuvent alors être surveillés en temps réel par les gestionnaires de la sécurité.

Pour répondre à cet enjeu, nous utilisons l'expertise métier qui est en mesure d'identifier les situations à risque. Nous choisissons donc des techniques d'apprentissage supervisé pour la prédiction des risques à l'atterrissage. Des techniques de sélection de variables sont également employées pour l'identification des paramètres de vol influents.

Cependant, l'analyse des données de vol est une tâche difficile du fait de leur nature complexe. En effet, à l'exception du jeu de données `Class1` où les paramètres de vol sont uniquement observés à 500 pieds, nous devons analyser des données échantillonnées dans le temps. Nous aurons donc naturellement recours à des outils provenant de l'*analyse des données fonctionnelles* (Ramsay and Silverman; 2005).

Les deux principaux thèmes statistiques étudiés dans ce manuscrit sont, d'une part, la sélection de variables en régression et en classification et, d'autre part, l'analyse des données fonctionnelles. Nous introduisons ces deux sujets dans les prochains paragraphes.

Apprentissage supervisé et sélection de variables

Considérons un vecteur aléatoire $(\mathbf{X}, Y)$ à valeurs dans $\mathbb{R}^p \times \mathcal{Y}$ dont la distribution de probabilité $P_{(\mathbf{X},Y)}$ est inconnue. Les méthodes d'apprentissage supervisé visent à estimer le lien entre le vecteur des covariables $\mathbf{X} = (X_1, \ldots, X_u, \ldots, X_p)$ (les paramètres de vol dans notre cas) et une variable à prédire Y (l'atterrissage long, par exemple), c'est-à-dire une fonction mesurable f définie sur $\mathbb{R}^p$ et à valeurs dans $\mathcal{Y}$. L'estimation d'une telle fonction se fait au moyen d'un échantillon $\mathcal{D}_n = \{(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_i, Y_i), \ldots, (\mathbf{X}_n, Y_n)\}$ de n couples indépendants et identiquement distribués (i.i.d.) de loi $P_{(\mathbf{X},Y)}$. Un estimateur $\hat{f}$ de f permet de prédire la valeur de sortie Y_{n+1} sachant une nouvelle observation $\mathbf{X}_{n+1}$. Dans la suite, nous étudierons à la fois la régression où $\mathcal{Y} = \mathbb{R}$ et la classification binaire lorsque $\mathcal{Y} = \{-1, 1\}$.

Du fait de la complexité des données de vol, les problèmes que l'on étudie sont difficiles à décrire par des modèles simples. C'est pourquoi, nous avons décidé de privilégier des approches non paramétriques et non linéaires plutôt que de décrire la distribution $P_{(\mathbf{X},Y)}$ de façon paramétrique.

Introduit par Breiman (2001) comme une variante du bagging, l'algorithme des forêts aléatoires agrège une collection d'arbres CART (Breiman et al.; 1984). Les arbres sont construits à partir d'échantillons *bootstrap* de l'ensemble d'apprentissage et leur agrégation améliore substantiellement les performances des arbres individuels. Les forêts aléatoires sont utilisées à la fois en régression et en classification. Elles ont montré de très bonnes performances en pratique pour des problèmes complexes (relations non linéaires, interactions, grande dimension, etc.).

Par ailleurs, dans un contexte de grande dimension, considérer des modèles de prédiction présentant un grand nombre de paramètres à estimer peut provoquer un phénomène de sur-apprentissage. La sélection de variables vise à réduire le nombre de variables utilisées dans le modèle pour améliorer les performances prédictives. De plus, notre objectif n'est pas de trouver toutes les variables liées à Y mais d'exhiber celles dont l'erreur de prédiction est minimale. Nous recherchons ainsi de la parcimonie dans les données pour, *in fine*, interpréter plus facilement les situations de vol observées en nous appuyant sur une connaissance experte du domaine aérien. Une solution serait d'évaluer les performances de tous les sous-ensembles de variables (sélection exhaustive) mais cette approche n'est pas envisageable dans un contexte de grande dimension. C'est pourquoi nous considérons un algorithme inspiré de la méthode *Recursive Feature Elimination* (RFE, Guyon et al.; 2002). Dans le cas des forêts aléatoires, nous utilisons la mesure d'importance par permutation (Breiman; 2001) qui permet d'évaluer la capacité de chaque variable à prédire Y. Nous employons ainsi une stratégie descendante (ou *backward*) où l'on retire pas-à-pas les variables les moins importantes au sens du critère d'importance. À chaque étape de l'algorithme, nous calculons l'erreur de prédiction. Le sous-ensemble finalement choisi est celui qui minimise l'erreur de prédiction (Algorithme 1). Cette procédure permet d'approcher les performances d'une sélection exhaustive en guidant la recherche du meilleur sous-ensemble par la mesure d'importance des forêts aléatoires.

Algorithm 1 *Recursive Feature Elimination* avec les forêts aléatoires

1: Construire une forêt avec les variables courantes et calculer l'erreur de prédiction
2: Calculer la mesure d'importance pour chaque variable
3: Éliminer la variable la moins importante
4: Répéter les étapes 1 à 3 tant qu'il reste des variables

Apprentissage de données fonctionnelles

L'analyse des données fonctionnelles désigne la modélisation et le traitement statistique de variables aléatoires à valeurs dans un espace de fonctions (Ramsay and Silverman; 2005). Ainsi, lorsque qu'un paramètre de vol X_u est observé dans le temps, nous ne pouvons plus le considérer à valeurs dans $\mathbb{R}$ mais dans un espace de fonctions $\mathcal{X}$, typiquement l'espace de Hilbert $L^2([0,1])$. Une façon de faire est de projeter X_u sur une base $\mathcal{B} = \{\varphi_1, \varphi_2, \ldots\}$ de fonctions orthogonales de $L^2([0,1])$ puis de considérer les d_u premiers coefficients de base comme nouvelles variables explicatives. Autrement dit, nous considérons une représentation fonctionnelle de la forme :

$$X_u(t) \simeq \sum_{k=1}^{d_u} \langle X_u, \varphi_k \rangle_{L^2} \varphi_k(t), \tag{1.4.1}$$

où $\langle \cdot, \cdot \rangle_{L^2}$ est le produit scalaire usuel de $L^2([0,1])$. Ainsi, on approche X_u dans $L^2([0,1])$ par sa projection sur le sous-espace de dimension d_u engendré par les fonctions $\varphi_1, \ldots, \varphi_{d_u}$. En pratique, la fonction X_u est observée sur une grille discrète de points $(t_1, \ldots, t_N)$. Les produits scalaires $\langle X_u, \varphi_k \rangle_{\mathcal{X}}$ sont alors approchés par les coefficients empiriques $Z_{uk} := \frac{1}{N} \sum_{\ell=1}^{N} X_u(t_\ell)\varphi_k(t_\ell)$. La décomposition (1.4.1) est donc estimée à partir du processus stochastique $\{X_u(t_\ell), \ell \in \{1, \ldots, N\}\}$.

Finalement, trouver une représentation fonctionnelle des données permet de se ramener à un cadre d'apprentissage en dimension finie en concaténant l'ensemble des paramètres de vol en un nouveau vecteur aléatoire $\mathbf{Z} = (Z_{11}, \ldots, Z_{1d_1}, Z_{21}, \ldots, Z_{2d_2}, \ldots)$. On observe à présent un couple $(\mathbf{Z}, Y)$ à valeurs dans $\mathbb{R}^D \times \mathcal{Y}$ où $D = \sum_u d_u$. Les algorithmes de classification et de régression tels que les forêts aléatoires peuvent ainsi être adaptés au traitement de données fonctionnelles.

1.5 Organisation du manuscrit et contributions

Dans les sections qui suivent, nous présentons brièvement le contenu des différents chapitres ainsi que les résultats obtenus.

1.5.1 Chapitre 2 : Analyse du risque d'atterrissage long

Ce premier chapitre vise dans un premier temps à introduire les concepts généraux de l'apprentissage statistique et de la sélection de variables dans un contexte non fonctionnel. Nous y présentons notamment les forêts aléatoires ainsi que les méthodes de sélection de variables associées. Nous montrons que la mesure d'importance par permutation (Breiman; 2001) est liée aux indices de Sobol (Sobol; 1993) utilisés couramment en analyse de sensibilité.

Les méthodes présentées sont appliquées au problème de la prédiction des atterrissages longs lorsque les paramètres de vol sont observés à l'altitude de 500 pieds (jeu de données `Class1`). Nous comparons trois méthodes de classification, la régression logistique, les machines à vecteurs de support (SVM) et les forêts aléatoires.

Six paramètres de vol discriminants sont sélectionnés par les trois méthodes : la vitesse sol, le cap magnétique, l'accélération longitudinale, la masse de l'avion, l'assiette et la position des volets. Nous identifions alors, en collaboration avec les experts aéronautiques, plusieurs facteurs de risque. Tout d'abord, la présence du cap montre un risque d'atterrissage long plus élevé sur certains aéroports. La masse et la vitesse traduisent une plus grande difficulté de décélération à l'atterrissage. Enfin, la vitesse du vent à 500 pieds

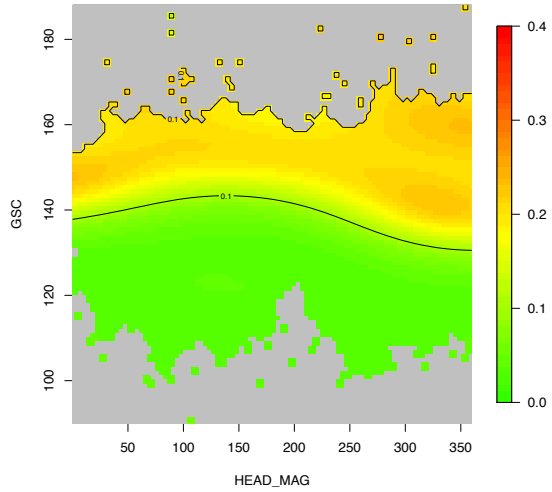


Figure 1.7 – Cartographie du risque d'atterrissage long pour la vitesse sol (GSC) et le cap magnétique (HEAD_MAG).

est également un facteur de risque. Elle a notamment un impact sur la stabilisation de l'avion à cette altitude. Nous montrons également que les forêts aléatoires ont de meilleures performances prédictives que les SVM et la régression logistique.

Pour conclure ce chapitre, nous proposons une cartographie représentant le risque d'atterrissage long à partir des probabilités *a posteriori* estimées (Figure 1.7). Une telle représentation des risques fournit un premier outil d'aide à la décision proposé aux opérationnels.

Cette première analyse des enregistreurs de vol apporte une première réponse satisfaisante au problème de la prédiction des atterrissages longs. Les résultats suggèrent d'aller plus loin dans l'analyse en traitant une séquence de vol plutôt que de se restreindre à une altitude donnée. Il suggèrent également de filtrer les données selon l'aéroport de destination.

Acquis du chapitre : première réponse satisfaisante au problème de la prédiction des atterrissages longs.
Objectifs du chapitre suivant : présentation d'outils d'analyse des données fonctionnelles pour traiter les paramètres de vol observés dans le temps.

1.5.2 Chapitre 3 : Apprentissage de données fonctionnelles

Dans ce chapitre, nous présentons les outils dont nous avons besoin pour traiter les données de vol échantillonnées dans le temps. Nous dressons un état de l'art non exhaustif de l'analyse des données fonctionnelles et donnons un bref aperçu de la littérature. Les méthodes présentées seront appliquées dans les chapitres suivants.

Nous introduisons tout d'abord l'approche par projection qui consiste à représenter chaque variable fonctionnelle sur une base de fonctions orthogonales (Équation (1.4.1)). L'objectif est de se ramener à un contexte d'apprentissage dans un espace de dimension finie. Ainsi, pour un paramètre X_u, nous recherchons une représentation fonctionnelle

commune aux n vols $X_{u1}, \ldots, X_{un}$. Autrement dit, les vols sont projetés sur une même base de fonctions $\{\varphi_1, \ldots, \varphi_{d_u}\}$, si possible de faible dimension.

Les méthodes de seuillages par ondelettes proposées par Donoho and Johnstone (1994) sont très utilisées pour débruiter des signaux. En particulier, la règle du seuillage "dur" réduit à zéro les coefficients d'ondelettes dont la valeur absolue est inférieure à un seuil donné. Cette méthode ne permet cependant pas de trouver une représentation fonctionnelle commune à n signaux indépendants. En effet, les coefficients retenus en seuillant indépendamment les n courbes n'ont aucune raison d'être rigoureusement les mêmes pour chaque signal. C'est pourquoi, nous adaptons la règle du seuillage dur pour la réduction de dimension simultanée de n processus indépendants.

Nous rappelons ensuite les principes fondamentaux de l'Analyse en Composantes Principales Fonctionnelle. L'intérêt de cette méthode de réduction de dimension est double. D'une part, elle permet de représenter n processus sur une base commune, la base de Karhunen-Loève. D'autre part, elle fournit une représentation compacte des signaux, c'est-à-dire basée sur un faible nombre de composantes principales.

Pour conclure ce chapitre, nous présentons un ensemble de références bibliographiques traitant de différents problèmes fonctionnels. Nous détaillons les modèles de régression linéaire et logistique fonctionnels ainsi que des méthodes basées sur des algorithmes non paramétriques ou non linéaires.

Acquis du chapitre : présentation d'outils d'analyse des données fonctionnelles et proposition d'une méthode de seuillage par ondelettes pour n processus indépendants.
Objectifs du chapitre suivant : étude de la sélection de variables avec les forêts aléatoires et application au contexte fonctionnel pour la prédiction des atterrissages longs.

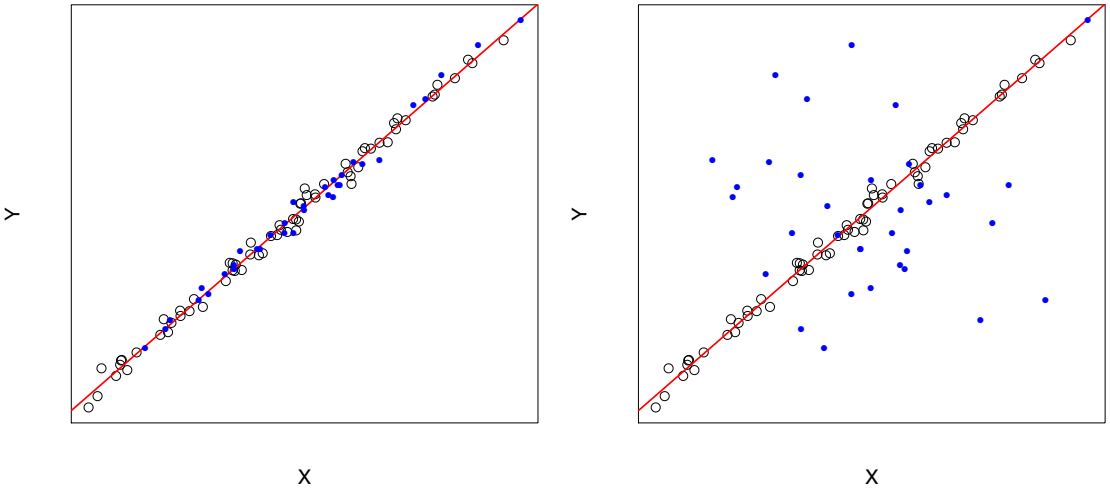
1.5.3 Chapitre 4 : Corrélation et importance des variables dans les forêts aléatoires

Ce chapitre traite de la problématique de la sélection de variables avec les forêts aléatoires dans un contexte non fonctionnel. Les méthodes présentées sont ensuite appliquées à l'étude du risque d'atterrissage long lorsque les données sont temporellement observées durant la phase d'approche. Ce chapitre fait l'objet d'un article disponible en version pré-publiée (Gregorutti et al.; 2014a).

La mesure d'importance par permutation a été introduite par Breiman (2001) pour pallier le manque d'interprétation des forêts aléatoires. Rappelons que les arbres qui constituent une forêt aléatoire sont construits à partir d'échantillons *bootstrap* des données. Pour chaque arbre, l'ensemble des observations qui ne sont pas retenues dans les *bootstrap* est appelé échantillon Out-Of-Bag (OOB). Ces échantillons sont utilisés pour mesurer l'importance des variables pour la prédiction de Y. Plus précisément, une variable X_u (ici à valeurs dans $\mathbb{R}$) est considérée comme importante si en cassant le lien entre X_u et Y, l'erreur de prédiction augmente. Pour casser ce lien, Breiman propose de permuter aléatoirement les réalisations de X_u dans les échantillons OOB. L'importance de X_u est alors l'augmentation moyenne de l'erreur de prédiction sur l'ensemble des arbres (Figure 1.8). Ce critère peut être vu comme un estimateur empirique de la quantité

$$\mathcal{I}(X_u) = \mathbb{E}\ell\left(Y, f(\mathbf{X}_{(u)})\right) - \mathbb{E}\ell\left(Y, f(\mathbf{X})\right), \tag{1.5.1}$$

où ℓ est une fonction de perte – typiquement la perte quadratique pour la régression ou la perte binaire pour la classification – et $\mathbf{X}_{(u)} = (X_1, \ldots, X'_u, \ldots, X_p)$ est un vecteur aléatoire



(a) Avant permutation (b) Après permutation

Figure 1.8 – Augmentation de l'erreur après permutation des valeurs d'une variable X pour l'échantillon OOB (points bleus). Les points noirs représentent l'ensemble *bootstrap*.

tel que X'_u est une réplication indépendante de X_u et indépendante des autres variables. Le remplacement de X_u par X'_u modélise la permutation aléatoire des observations contenues dans les échantillons OOB. Cette définition a été formalisée récemment par Zhu et al. (2012).

Dans un premier temps, nous étudions les aspects théoriques du critère (1.5.1) pour un modèle de régression additive, c'est-à-dire lorsque $\mathbb{E}[Y|\mathbf{X} = \mathbf{x}] = \sum_{u=1}^{p} f_u(x_u)$. Nous montrons que le critère $\mathcal{I}$ s'écrit de manière très simple comme

$$\mathcal{I}(X_u) = 2\,\mathrm{Var}(f_u(X_u)).$$

Si, de plus, la fonction de régression est linéaire, alors

$$\mathcal{I}(X_u) = 2\alpha_u^2\,\mathrm{Var}(X_u).$$

Par ailleurs, cette mesure d'importance est connue pour être sensible à la corrélation entre les variables explicatives. Nous montrons, en effet, que les valeurs d'importance décroissent lorsque le nombre de variables corrélées et le niveau de corrélation augmentent. Les résultats théoriques obtenus sont validés par des simulations. De plus, nous montrons numériquement que l'algorithme de sélection de variables RFE (Algorithme 1) se comporte bien en cas de corrélation. En effet, cette procédure limite les effets de la corrélation sur la mesure d'importance et permet de sélectionner un faible nombre de variables. Ces résultats sont obtenus pour plusieurs protocoles de simulation rencontrés dans la littérature.

Enfin, nous revenons à notre objectif initial de l'analyse des enregistreurs de vol dans un contexte fonctionnel. L'algorithme RFE est utilisé pour la classification des atterrissages longs lorsque les données sont échantillonnées chaque seconde durant la phase d'approche (jeu de données `Class2`). La procédure s'effectue en trois temps. Tout d'abord, nous projetons les signaux sur une base d'ondelettes. Nous sélectionnons ensuite les coefficients

d'ondelettes au moyen de l'algorithme RFE. Les paramètres de vol finalement choisis sont ceux dont les coefficients sont les plus fréquemment sélectionnés. Cette procédure permet d'identifier quatre paramètres de vol comme étant les plus influents pour le risque d'atterrissage long. Il s'agit de la température de l'air, de l'altitude, de la masse de l'avion et de la vitesse du vent. La présence de vent arrière durant le palier d'interception (à 4000 pieds) est notamment identifiée comme facteur de risque. Ces résultats ont été validés d'un point de vue opérationnel. Ils ont notamment confirmé des intuitions des experts sur le lien entre la présence de vent arrière lors du palier d'interception et les atterrissages longs.

Acquis du chapitre : étude des effets de la corrélation sur la sélection de variables avec les forêts aléatoires et application à l'étude du risque d'atterrissage long.
Objectifs du chapitre suivant : extension de la mesure d'importance par permutation pour des groupes de variables et application à la sélection des paramètres de vol.

1.5.4 Chapitre 5 : Mesure d'importance groupée et application à l'apprentissage de données fonctionnelles multivariées

Dans ce chapitre, nous étendons la mesure d'importance par permutation pour des groupes de variables avec l'objectif d'analyser plus finement les données de vol dans le cadre fonctionnel. Ce chapitre fait l'objet d'un article disponible en version pré-publiée (Gregorutti et al.; 2014b).

En accord avec l'idée de Breiman, nous mesurons l'augmentation de l'erreur de prédiction après permutation aléatoire de groupes de variables pour les observations OOB. Soit U un sous-ensemble de $\{1,\dots,p\}$ définissant un groupe de variables $\mathbf{X}_U := \{X_u\}_{u\in U}$ et soit $\bar{U}$ son complémentaire dans $\{1,\dots,p\}$ [3]. Nous définissons la mesure d'importance groupée par

$$\mathcal{I}(\mathbf{X}_U) = \mathbb{E}\ell\left(Y, f(\mathbf{X}_{(U)})\right) - \mathbb{E}\ell\left(Y, f(\mathbf{X})\right),$$

où $\mathbf{X}_{(U)} = (\mathbf{X}'_U, \mathbf{X}_{\bar{U}})$ et $\mathbf{X}'_U$, un vecteur aléatoire indépendant et de même loi que $\mathbf{X}_U$.

Les résultats du chapitre précédant peuvent être étendus à l'importance groupée. En effet, si la fonction de régression s'écrit $\mathbb{E}[Y|\mathbf{X}=\mathbf{x}] = f_U(\mathbf{x}_U) + f_{\bar{U}}(\mathbf{x}_{\bar{U}})$, alors

$$\mathcal{I}(\mathbf{X}_U) = 2\,\mathrm{Var}(f_U(\mathbf{X}_U)).$$

La conséquence directe de ce résultat est la suivante : si la fonction f_U est additive et sous l'hypothèse d'indépendance des variables du groupe, alors l'importance groupée est exactement la somme des importances individuelles. Cette propriété n'est cependant plus valable lorsque le groupe est composé de variables corrélées. Une étude de simulations suggère par ailleurs qu'en toute généralité, l'importance groupée n'est pas simplement une somme d'importances individuelles. Ces résultats montrent également que le niveau d'importance du groupe est d'autant plus grand que le nombre de variables le composant croît. C'est pourquoi nous proposons une version normalisée du critère $\mathcal{I}$ permettant de ne pas favoriser les groupes de taille plus importante :

$$\mathcal{I}_{\mathrm{nor}}(\mathbf{X}_U) := \frac{1}{|U|}\mathcal{I}(\mathbf{X}_U).$$

Dans une deuxième partie, nous proposons une nouvelle méthode de sélection de variables fonctionnelles basée sur la mesure d'importance groupée. Chaque variable fonc-

3. Les notations sont simplifiées dans cette section afin de faciliter la lecture.

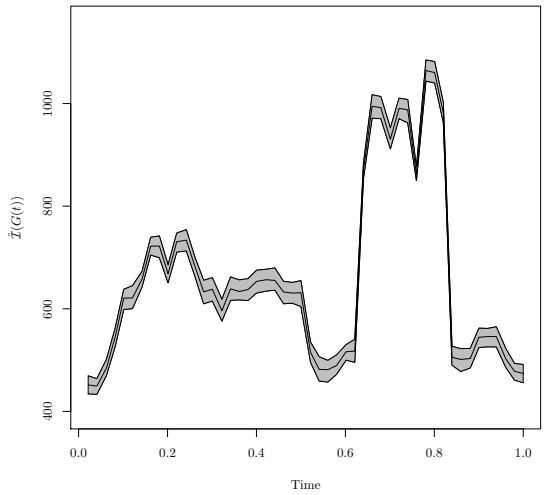


Figure 1.9 – Importance temporelle pour l'angle d'attaque.

tionnelle X_u est projetée sur une base d'ondelettes de dimension d_u. Nous pouvons donc mesurer l'importance de X_u en calculant l'importance du groupe composé de ses coefficients de base. L'algorithme RFE (Algorithme 1) est adapté à ce contexte pour sélectionner les groupes de variables qui minimisent l'erreur de prédiction. De plus, afin d'analyser de façon fine la décomposition des signaux sur la base d'ondelettes, nous proposons de regrouper les coefficients par

- niveaux d'ondelettes c'est-à-dire en groupes de taille croissante représentant les degrés d'approximation de plus en plus fins ;
- intervalles de temps.

Dans ce second cas, pour un temps t fixé, les coefficients d'ondelettes dont les fonctions de base correspondantes sont non nulles en t sont regroupés. Nous mesurons ainsi l'importance de chaque variable fonctionnelle en fonction du temps (Figure 1.9). Un ensemble de simulations est effectué dans chaque cas à partir de protocoles utilisant les ondelettes. Nous montrons que l'algorithme RFE basé sur la mesure d'importance groupée se comporte bien, même en cas de fortes corrélations entre les variables fonctionnelles.

Pour finir ce chapitre, la sélection groupée est appliquée à l'étude du risque d'atterrissage long. À la différence des chapitres précédents, nous étudions le problème de la régression des distances d'atterrissage pour les données `Reg1`. Nous identifions des paramètres de vol relatifs à la trajectoire de l'avion durant l'approche (angle d'attaque, écart au plan de descente), des paramètres de pilotage (position des gouvernes) et la masse de l'avion. Ces résultats sont cohérents avec ceux obtenus dans le Chapitre 4.

Acquis du chapitre : extension de la mesure d'importance par permutation pour des groupes de variables, proposition d'une procédure de sélection de variables fonctionnelles et application à la régression des distances d'atterrissage.
Objectifs du chapitre suivant : comparaison de méthodes de sélection des paramètres de vol.

1.5.5 Chapitre 6 : Comparaison de méthodes fonctionnelles appliquées aux données de vol

Dans ce chapitre, nous comparons différentes méthodes de sélection de variables pour étudier la prédiction des distances d'atterrissage. Nous comparons tout d'abord les trois approches employées dans ce mémoire, la sélection à 500 pieds (Chapitre 2), la sélection des coefficients d'ondelettes (Chapitre 4) et la sélection groupée (Chapitre 5). Dans les trois cas, nous utilisons l'algorithme RFE avec les forêts aléatoires (Algorithme 1).

Nous observons que la sélection à 500 pieds est moins performante que les approches fonctionnelles. Ce résultat confirme qu'une séquence entière de vol contient plus d'informations qu'un instant donné pour prédire les distances d'atterrissages. Nous confirmons cette observation au regard des résultats donnés par les deux procédures fonctionnelles (sélection des coefficients d'ondelettes et sélection groupée). En effet, les paramètres de vol identifiés sont relatifs à la trajectoire de l'avion ainsi qu'au pilotage en phase d'approche : la position des gouvernes, l'angle d'attaque, l'écart au plan de descente et l'altitude. La sélection à 500 pieds a identifié la masse, la vitesse du vent ainsi que la vitesse air et la vitesse d'approche. Ces deux derniers paramètres sont liés à l'altitude de 500 pieds car les pilotes doivent respecter la vitesse d'approche à cet instant pour assurer une bonne stabilisation de l'avion.

Dans un second temps, nous comparons l'algorithme de sélection groupée des forêts aléatoires avec le *Group Lasso* (Yuan and Lin; 2006a), méthode bien connue en régression linéaire. Nous observons de meilleures performances prédictives avec les forêts aléatoires. Les deux méthodes donnent, par ailleurs, des résultats cohérents pour la sélection des paramètres de vol : la plupart des paramètres de vol sélectionnés par le *Group Lasso* sont parmi les plus fréquemment sélectionnés par les forêts aléatoires.

Acquis du chapitre : comparaison de différentes approches de sélection des paramètres de vol.
Objectifs du chapitre suivant : présentation du produit *FlightScanner* dans lequel s'intègrent les outils proposés dans ce manuscrit et illustration de l'utilisation du logiciel pour l'étude du risque d'atterrissage dur.

1.5.6 Chapitre 7 : Valorisation industrielle – Produit *FlightScanner*

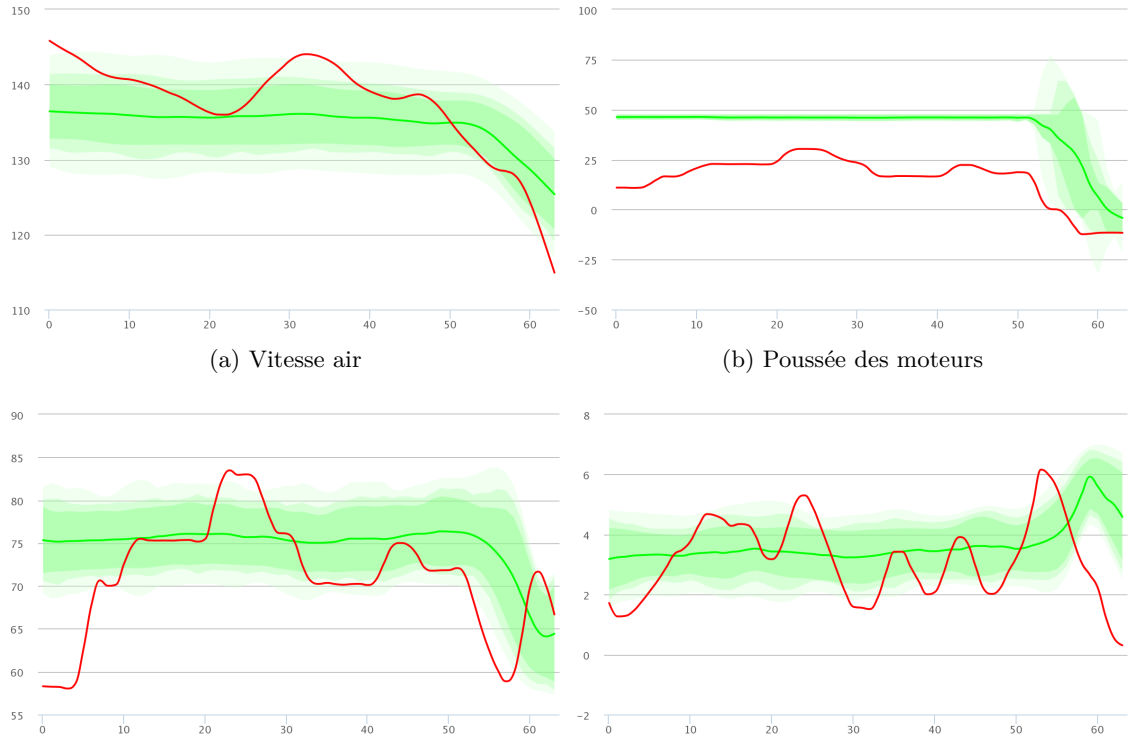
Ce chapitre à pour objectif d'illustrer la façon dont les outils proposés dans les chapitres précédents sont valorisés au sein de la société Safety Line. Nous présentons le produit *FlightScanner* dans lequel s'intègrent nos travaux. Au sein de la plateforme *Operation Health Monitoring* dédiée à l'analyse des données aéronautiques, cette solution innovante dans l'exploitation aérienne permet à la fois le monitoring des risques et le suivi des facteurs qui les influencent.

Le processus de traitement des données de vol implémenté dans le logiciel s'effectue en trois étapes : une phase de collecte des données, une phase d'apprentissage et une phase de prédiction. L'étape de collecte se résume à récupérer et à décoder les données directement issues des enregistreurs de vol pour les rendre utilisables par les algorithmes statistiques. L'étape d'apprentissage consiste à rechercher dans un historique d'exploitation les paramètres de vol ayant le plus d'influence sur un risque donné (atterrissage dur par exemple). L'analyse en composantes principales fonctionnelle est implémentée pour réduire la dimension des données échantillonnées dans le temps. Les paramètres de vol sont ensuite sélectionnés au moyen de l'algorithme de sélection groupée que l'on propose dans le Chapitre 5. De plus, cette étape d'apprentissage nous permet de construire des

profils-types de vol pour certains paramètres à surveiller. Il définissent des situations de vol sures et permettent aux utilisateurs d'observer des déviations de certains vols par rapport à la situation "normale". Plus la déviation est importante, plus le risque d'atterrissage dur est grand. La phase de prédiction consiste à évaluer le niveau de risque de chaque nouveau vol reçu à partir des résultats de la phase de sélection. Elle permet aux utilisateurs d'avoir des mesures objectives sur le niveau de risque à l'échelle d'un vol, d'une flotte ou d'un aéroport et surtout de pouvoir suivre son évolution dans le temps. Les nouveaux vols sont comparés aux profils-types pour identifier ceux qui dévient de la situation "normale" et donc de préciser les facteurs de risque.

Dans un second temps, nous présentons une utilisation opérationnelle des outils implémentés dans *FlightScanner* pour l'étude du risque d'atterrissage dur. Nous décrivons en particulier une interprétation des résultats fournis par les interfaces utilisateur. Les résultats montrent que la réduction prématurée de la vitesse de l'avion et un arrondi effectué trop tôt sont des facteurs de risque pour l'atterrissage dur (Figure 1.10).

Acquis du chapitre : présentation du produit *FlightScanner* dans lequel s'intègrent les outils proposés dans ce manuscrit et illustration de l'utilisation du logiciel pour l'étude du risque d'atterrissage dur.



(a) Vitesse air

(b) Poussée des moteurs

(c) Régime moteur

(d) Assiette

Figure 1.10 – Comparaison d'un vol à risque par rapport aux profils-types pour quatre paramètres de vol. Les graphiques sont issus de l'interface utilisateur de *FlightScanner*.

Chapitre 2

Analyse des données de vol à 500 pieds pour le risque d'atterrissage long

Résumé. Ce chapitre est une première analyse des enregistreurs de vol pour le risque d'atterrissage long. Dans un premier temps, nous introduisons l'apprentissage statistique supervisé et la sélection de variables afin de donner un aperçu des méthodes employées dans la littérature. Nous développons ensuite trois méthodes de classification : la régression logistique, les machines à vecteurs de support (SVM) et les forêts aléatoires. Ces méthodes sont comparées pour la prédiction des atterrissages longs lorsque les données de vol sont observées à 500 pieds.

Sommaire

2.1 Apprentissage statistique supervisé

L'apprentissage statistique supervisé, présenté dans cette section, se situe au carrefour de l'informatique, de l'optimisation et des statistiques. Un ensemble d'ouvrages de références traite de cette problématique, voir par exemple Devroye et al. (1996), Vapnik (1995, 1998), Cristianini and Shawe-Taylor (2000) et Hastie et al. (2001).

Généralités

Considérons un couple $(\mathbf{X}, Y)$ de variables aléatoires à valeurs dans $\mathbb{R}^p \times \mathcal{Y}$ dont la loi jointe $P_{(\mathbf{X},Y)}$ est inconnue. L'apprentissage supervisé consiste à estimer le lien entre $\mathbf{X} = (X_1, \ldots, X_p)$ (le vecteur des covariables) et Y (la variable de sortie), c'est-à-dire une fonction mesurable f définie sur $\mathbb{R}^p$ et à valeurs dans $\mathcal{Y}$.

L'erreur commise par une fonction f pour la prédiction de Y est donnée par

$$R(f) = \mathbb{E}\ell(Y, f(\mathbf{X})),$$

où ℓ est une fonction de perte fixée. L'application R, appelée risque, mesure l'écart moyen entre Y et sa prédiction $f(\mathbf{X})$. La meilleure fonction de prédiction $f^\star$ est alors celle qui minimise le risque sur la classe $\mathcal{F}$ des fonctions définies sur $\mathbb{R}^p$ et à valeurs dans $\mathcal{Y}$, c'est-à-dire

$$f^\star \in \arg\min_{f \in \mathcal{F}} R(f).$$

Cependant, à défaut de pouvoir calculer une telle fonction ne connaissant pas la loi $P_{(\mathbf{X},Y)}$, nous pouvons l'estimer à partir d'un échantillon $\mathcal{D}_n = \{(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)\}$ de n observations indépendantes et identiquement distribuées et de loi $P_{(\mathbf{X},Y)}$, où $\mathbf{X}_i = (X_{i1}, \ldots, X_{ip})$. Autrement dit, il s'agit de trouver une solution au problème de minimisation de l'estimateur empirique $\hat{R}(f)$ de $R(f)$ en se restreignant à une sous-classe $\mathcal{C}$ de $\mathcal{F}$:

$$\begin{aligned} \hat{f} &\in \arg\min_{f \in \mathcal{C}} \hat{R}(f) \\ &= \arg\min_{f \in \mathcal{C}} \frac{1}{n} \sum_{i=1}^{n} \ell(Y_i, f(\mathbf{X}_i)). \end{aligned}$$

L'estimateur ainsi optimisé fournit une prédiction $\hat{f}(\mathbf{X}_{n+1})$ pour une nouvelle observation $\mathbf{X}_{n+1}$, prédiction que l'on espère proche de la vraie valeur Y_{n+1}. Autrement dit, il s'agit de trouver $\hat{f}$ de sorte que $R(\hat{f})$ est proche de $R(f^\star)$. Ce principe, appelé *Minimisation du Risque Empirique*, a été formalisé par Vapnik (1995, 1998).

Le choix de l'espace $\mathcal{C}$ est conditionné par l'algorithme de prédiction ainsi que par l'*a priori* que l'on a sur les données. Par exemple, il est naturel de considérer des espaces paramétriques comme l'espace des fonctions linéaires, polynomiales, ou de façon plus générale un espace à noyau auto-reproduisant (RKHS). Un ensemble de méthodes dites non paramétriques basées sur le principe de la minimisation du risque empirique existent. Il s'agit des k plus proches voisins (Fix and Hodges; 1951), de la règle à noyau, des méthodes basées sur une partition des données (histogramme, arbres de décision). Ces méthodes sont adaptées à la régression et à la classification. Elles supposent une forme particulière à l'estimateur $\hat{f}$ (fonction constante par morceaux, moyenne pondérée etc.), voir le livre de Devroye et al. (1996).

Les deux problèmes que nous développerons dans ce manuscrit sont respectivement la régression où $\mathcal{Y} = \mathbb{R}$ et la classification binaire lorsque $\mathcal{Y} = \{-1, 1\}$. En régression, la fonction de perte généralement utilisée est la perte quadratique $\ell(y, f(\mathbf{x})) = (y - f(\mathbf{x}))^2$ et la fonction à estimer est $f^\star = \mathbb{E}[Y|\mathbf{X} = \mathbf{x}]$. Dans le cas de la classification binaire, la fonction de régression s'écrit

$$f^\star(\mathbf{x}) = 2\mathbb{P}[Y = 1|\mathbf{X} = \mathbf{x}] - 1.$$

L'objectif est alors d'estimer les probabilités a posteriori

$$\mathbb{P}[Y = 1|\mathbf{X} = \mathbf{x}] \text{ et } \mathbb{P}[Y = -1|\mathbf{X} = \mathbf{x}].$$

La fonction de décision est alors donnée par la règle de Bayes

$$g^\star(\mathbf{x}) = \begin{cases} 1 & \text{si } \mathbb{P}[Y = 1|\mathbf{X} = \mathbf{x}] \geq 0.5 \\ -1 & \text{sinon.} \end{cases}$$

Cette fonction est un minimiseur du risque $R(g)$ pour la perte $\ell(y, g(\mathbf{x})) = \mathbb{1}_{y \neq g(\mathbf{x})}$, couramment appelée perte "0/1" ou perte de classification.

Par ailleurs, il est souvent plus pratique d'exprimer la règle de Bayes en fonction de $f^\star$

$$g^\star(\mathbf{x}) = \begin{cases} 1 & \text{si } f^\star(\mathbf{x}) \geq 0 \\ -1 & \text{sinon,} \end{cases}$$

ce qui mène à considérer des estimateurs de la forme

$$\hat{g}(\mathbf{x}) = \begin{cases} 1 & \text{si } \hat{f}(\mathbf{x}) \geq 0 \\ -1 & \text{sinon,} \end{cases}$$

où $\hat{f}$ est une fonction à valeurs réelles qui définit un hyperplan séparateur. C'est le point de vue adopté par les Machines à vecteurs de support (SVM, Section 2.4) avec la fonction de perte *Hinge* $\ell(y, f(\mathbf{x})) = |1 - yf(\mathbf{x})|_+ = \max(0, 1 - yf(\mathbf{x}))$ et par la régression logistique utilisant la perte $\ell(y, f(\mathbf{x})) = \log[1 + \exp(-yf(\mathbf{x}))]$, voir la Section 2.3. Notons de plus, que ces deux fonctions de perte sont des relaxations convexes de la perte binaire comme le montre le graphe 2.1.

Compromis biais-variance

Les performances d'un estimateur $\hat{f}$ basé sur la minimisation du risque empirique sont très sensibles au choix de la classe $\mathcal{C}$. En effet, l'excès de risque $R(\hat{f}) - R(f^\star)$ se décompose comme suit :

$$R(\hat{f}) - R(f^\star) = \left(R(\hat{f}) - R(f^\star_{\mathcal{C}})\right) + \left(R(f^\star_{\mathcal{C}}) - R(f^\star)\right),$$

où $f^\star_{\mathcal{C}} \in \arg\min_{f \in \mathcal{C}} R(f)$. Le premier terme, dit de variance, mesure l'erreur d'estimation dans la classe $\mathcal{C}$. Le second terme mesure l'erreur commise par la restriction à la classe $\mathcal{C}$. Ce terme est appelé terme de biais, ou erreur d'approximation dans la communauté du Machine Learning. Plus la complexité de $\mathcal{C}$ augmente, plus la variance est grande et plus le biais est faible (Figure 2.2). Nous devons donc choisir avec attention l'espace $\mathcal{C}$ pour avoir le meilleur compromis entre biais et variance. Autrement dit, la classe $\mathcal{C}$ ne doit pas être trop vaste pour éviter le sur-apprentissage mais aussi pas trop restrictive afin d'espérer approcher la fonction $f^\star$ avec une bonne précision.

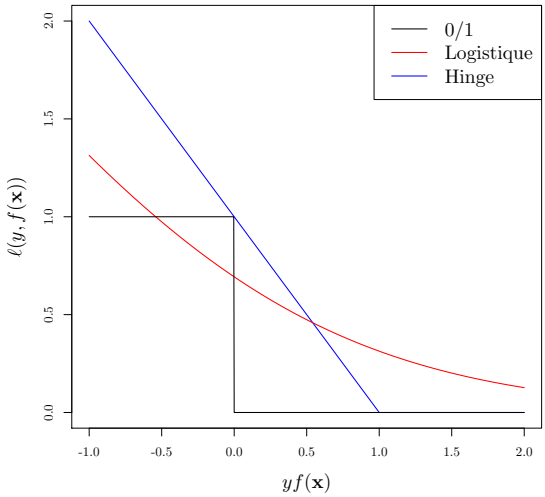


FIGURE 2.1 – Fonctions de perte de classification en fonction de $yf(\mathbf{x})$.

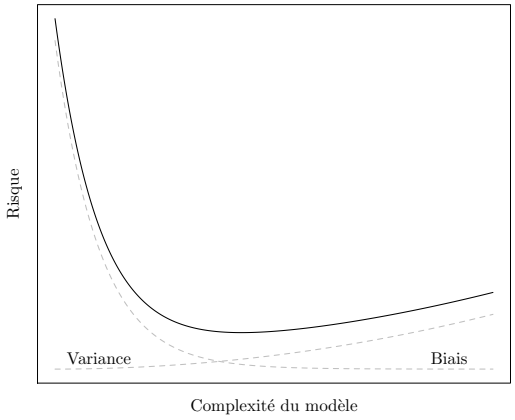


FIGURE 2.2 – Compromis biais-variance – Risque de prédiction en fonction de la complexité du modèle.

Prenons l'exemple du modèle de régression $Y = f^\star(\mathbf{X}) + \varepsilon$ avec ε une variable aléatoire d'espérance nulle et de variance σ^2 conditionnellement à $\mathbf{X}$. La décomposition biais-variance s'écrit :

$$\begin{aligned} R(\hat{f}) - R(f^\star) &= \mathbb{E}\left[\left(\hat{f}(\mathbf{X}) - f^\star(\mathbf{X})\right)^2\right] \\ &= \mathbb{E}\left[\left(\hat{f}(\mathbf{X}) - \mathbb{E}\hat{f}(\mathbf{X})\right)^2\right] + \left(\mathbb{E}\hat{f}(\mathbf{X}) - f^\star(\mathbf{X})\right)^2. \end{aligned}$$

En conséquence, l'erreur de prédiction s'écrit

$$R(\hat{f}) = \sigma^2 + \mathbb{E}\left[\left(\hat{f}(\mathbf{X}) - \mathbb{E}\hat{f}(\mathbf{X})\right)^2\right] + \left(\mathbb{E}\hat{f}(\mathbf{X}) - f^\star(\mathbf{X})\right)^2,$$

car $R(f^\star) = \sigma^2$. Cette dernière equation nous dit que l'erreur de prédiction $R(\hat{f})$ est la somme de l'erreur d'estimation de $f^\star$ par $\hat{f}$ et de σ^2, erreur incompressible du problème (Hastie et al.; 2001, Chap. 7).

Dans le cas de la grande dimension où n est comparable ou inférieur à p, le compromis biais-variance prend tout son sens. En effet, plus le nombre de variables est grand, plus la classe $\mathcal{C}$ est riche et plus le terme de variance augmente. Les méthodes d'estimation par critère pénalisé ainsi que l'approche par sélection de variables ont pour objectif de réaliser le compromis biais-variance.

Approches pénalisées

La recherche du meilleur estimateur $\hat{f}$ au sens du compromis biais-variance est un problème fondamental en statistique. Une solution possible consiste à pénaliser le risque empirique c'est-à-dire à considérer un problème du type

$$\arg\min_{f \in \mathcal{C}} \left\{ \hat{R}(f) + \mathrm{pen}_\lambda(f) \right\},$$

où le terme de pénalité $\mathrm{pen}_\lambda(f)$ permet de contraindre la complexité de le classe $\mathcal{C}$ à travers un paramètre de régularisation $\lambda > 0$. Un nombre important de travaux de la littérature suivent cette approche. Citons en particulier la *minimisation structurelle du risque* qui cherche à minimiser le risque empirique sur une collection de classes de taille croissante et qui choisit le meilleur estimateur en pénalisant la complexité (voir les ouvrages de Vapnik; 1995, 1998 et Devroye et al.; 1996, Chap. 18).

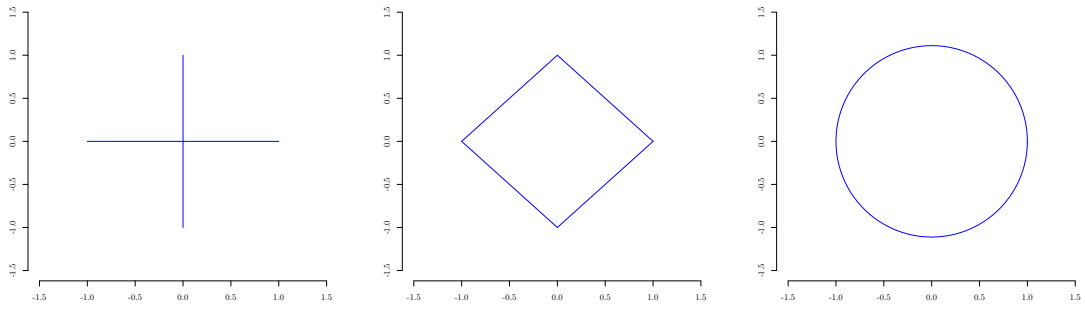
Dans le cas de la régression linéaire, c'est-à-dire telle que $f(\mathbf{x}) = \beta^\top \mathbf{x}$, un très grand nombre de méthodes pénalisées existent. Dans ce cas,

$$\hat{R}(f) = \hat{R}(\beta) = \frac{1}{n} \sum_{i=1}^{n} \left(Y_i - \beta^\top \mathbf{X}_i\right)^2.$$

Nous employons les notations suivantes dans la suite : pour un vecteur β de dimension p, $\|\beta\|_0 = \sharp\{j, \beta_j \neq 0\}$ désigne le nombre de composantes non nulles de β (norme ℓ_0), $\|\beta\|_1 = \sum_{j=1}^p |\beta_j|$ et $\|\beta\|_2 = (\sum_{j=1}^p \beta_j^2)^{1/2}$ sont respectivement les normes ℓ_1 et ℓ_2 de β. Dans la suite de la section, nous présentons les principaux problèmes d'estimation par pénalisation.

La Pénalisation ℓ_0 (ou sélection complète) est définie par

$$\hat{\beta}^{\ell_0} \in \arg\min_{\beta \in \mathbb{R}^p} \left\{ \hat{R}(\beta) + \lambda \|\beta\|_0 \right\}, \ \lambda > 0.$$



(a) ℓ_0 (b) ℓ_1 (c) ℓ_2

FIGURE 2.3 – Boules unité des normes ℓ_0, ℓ_1 et ℓ_2.

Le terme $\|\beta\|_0$ permet de pénaliser le nombre de composantes non nulles du vecteur β. Cette approche correspond par exemple aux critères d'information AIC (*Akaike Information Criterion*, Akaike; 1974), BIC (*Bayesian Information Criterion*, Schwarz; 1978) et le Cp de Mallows (Mallows; 1973), voir aussi Massart (2003). Ces critères sont très utilisés pour la sélection de variables dans le cadre du modèle linéaire à travers des algorithmes "pas à pas" que nous décrirons dans la suite.

La régression Lasso (Tibshirani; 1996), est définie par

$$\hat{\beta}^{lasso} \in \arg\min_{\beta\in\mathbb{R}^p} \left\{ \hat{R}(\beta) + \lambda\|\beta\|_1 \right\}, \ \lambda > 0.$$

La pénalité ℓ_1 est la relaxation convexe de la pénalité ℓ_0 (Figure 2.3). Elle impose de la parcimonie à l'estimateur $\hat{\beta}^{lasso}$ et permet donc de sélectionner les variables. Ce problème est aussi connu sous le nom de *basis pursuit* Chen et al. (1998). Un très grand nombre de méthodes connexes basées sur ce type de pénalité ont été proposées comme par exemple le *Group Lasso* (Yuan and Lin; 2006a), le Lasso adaptatif (Zou; 2006) ainsi que des versions pour le modèle logistique (van de Geer; 2008; Meier et al.; 2008; Kwemou; 2012).

La régression *Ridge* (Tikhonov; 1963) est définie par

$$\hat{\beta}^{ridge} \in \arg\min_{\beta\in\mathbb{R}^p} \left\{ \hat{R}(\beta) + \lambda\|\beta\|_2^2 \right\}, \ \lambda > 0,$$

où la norme ℓ_2 contraint les grandes valeurs des paramètres β_j.

L'*Elastic-net* (Zou and Hastie; 2005) est défini par

$$\hat{\beta}^{elnet} \in \arg\min_{\beta\in\mathbb{R}^p} \left\{ \hat{R}(\beta) + \lambda_1\|\beta\|_1 + \lambda_2\|\beta\|_2^2 \right\}, \ \lambda_1 > 0, \ \lambda_2 > 0.$$

La combinaison des pénalités ℓ_1 et ℓ_2 a pour avantage de bien prendre en compte les corrélations entre les variables. En effet, là où le Lasso ne sélectionne qu'une variable parmi un groupe de variables corrélées, l'*Elastic Net* sélectionne l'ensemble des variables du groupe.

D'autre part, si f est supposé appartenir à un RKHS $\mathcal{H}$, alors il existe un ensemble de méthodes pénalisées étudiées notamment par Evgeniou et al. (2000). De façon générale, une

pénalité *Ridge* est ajoutée au risque empirique ce qui conduit au problème de minimisation

$$\hat{f} \in \arg\min_{f \in \mathcal{H}} \left\{ \hat{R}(f) + \lambda \|f\|_{\mathcal{H}}^2 \right\}, \ \lambda > 0,$$

où $\|.\|_{\mathcal{H}}$ est la norme associée à $\mathcal{H}$. Ce problème concerne la régression *Ridge* à noyau utilisant la perte quadratique (Saunders et al.; 1998; Suykens et al.; 2002), les SVM pour la régression lorsque $\ell(y, f(\mathbf{x})) = |y - f(\mathbf{x})|_{\varepsilon} = \max(0, |y - f(\mathbf{x})| - \varepsilon)$ (Drucker et al.; 1996). Les SVM pour la classification est aussi considéré. La perte utilisée est la perte *Hinge* (Boser et al.; 1992). L'ouvrage de Schölkopf and Smola (2001) est une introduction complète des méthodes à noyau.

2.2 Sélection de variables

Dans cette section, nous présentons des techniques de sélection de variables, thème dominant de nos travaux de thèse. Sélectionner les variables pertinentes pour la prédiction est un des problèmes majeurs en statistique et en *Machine Learning* tant d'un point de vue théorique qu'appliqué. D'un point de vue industriel et tout particulièrement dans le cas de l'analyse des risques opérationnels en aéronautique, l'identification des facteurs de risques est crucial pour conserver un niveau de sécurité élevé.

En toute généralité, nous pouvons voir la sélection de variables comme une réponse au problème du compromis biais-variance. En effet, lorsque le nombre de variables est important (potentiellement très supérieur au nombre d'observations), il peut se produire une situation de sur-apprentissage. La sélection de variables vise à réduire le nombre de variables utilisées dans le modèle pour améliorer les performances prédictives. De plus, un modèle parcimonieux – ayant un faible nombre de variables – est plus interprétable en pratique. L'article de Guyon and Elisseeff (2003) présente une revue des trois approches principales : *filter*, *embedded* et *wrapper*.

Algorithme *filter*

Un algorithme *filter* consiste à classer les variables selon un critère d'importance et à sélectionner les q variables qui maximisent ce critère. Cette approche vise à pré-traiter les variables en éliminant celles qui portent le moins d'information. On parle de *sreening* dans ce cas. Les critères usuels sont le coefficient de corrélation entre X_j et Y, l'information mutuelle qui mesure la distance entre les distributions de probabilité entre X_j et Y et le test de Student (voir Lazar et al.; 2012). Le principal inconvénient de cette approche vient du fait que le choix final des variables se fait indépendamment des performances de l'algorithme de classification ou de régression.

Algorithme *embedded*

Un algorithme *embedded*, que l'on peut traduire par "intégré", sélectionne les variables directement dans le processus d'apprentissage. Parmi les plus célèbres, nous pouvons citer les arbres aléatoires (l'algorithme CART entre autres, voir Section 2.5) et les méthodes de type Lasso introduites précédemment. Dans ce dernier cas, la sélection est due à la pénalité ℓ_1 qui annule les coefficients de régression des variables non informatives.

Algorithme *wrapper*

Un algorithme de type *wrapper*, formalisé par Kohavi and John (1997) et Blum and Langley (1997), utilise la procédure d'apprentissage pour trouver un sous-ensemble optimal de variables au sens de l'erreur de prédiction. Une procédure efficace serait d'évaluer les performances de tous les sous-ensembles possibles, ce qui est évidemment impossible dès lors que la dimension du problème croît. C'est pourquoi, un algorithme *wrapper* emploie une stratégie pas-à-pas où les variables sont itérativement ajoutées (procédure ascendante ou *forward*) ou bien retirées (procédure descendante ou *backward*). Ces algorithmes sont un moyen d'approcher les performances d'une sélection exhaustive en guidant la recherche du meilleur sous-ensemble. En effet, pour déterminer les variables à ajouter ou à retirer, il est nécessaire de se munir d'un critère mesurant l'information portée par chacune des variables. Par exemple, Kohavi and John (1997) utilisent l'erreur de prédiction mais il est possible d'employer les mesures de rang décrites plus haut. Les critères d'information (AIC, BIC, Cp de Mallows, etc.) sont également utilisés dans le cas du modèle de régression linéaire (pour plus de détails, consulter Hastie et al.; 2001, Chap. 3). L'algorithme *wrapper* le plus connu est l'algorithme SVM *Recursive Feature Elimination* (ou SVM-RFE) introduit par Guyon et al. (2002). Nous développons plus en détail cet algorithme dans la section 2.4. Une adaptation aux forêts aléatoires est également proposée dans les chapitres 4 et 5.

Ambroise and McLachlan (2002) ont identifié un biais de sélection dans le cas de l'algorithme SVM-RFE. Guyon et al. (2002) estiment l'erreur de prédiction par validation croisée *Leave-One-Out* à chaque étape de l'algorithme. Ambroise and McLachlan (2002) notent que procéder à une validation croisée de façon interne à l'algorithme SVM-RFE entraine une sous-estimation de l'erreur. Au même titre que le sur-apprentissage, la sélection et le test de chaque sous-ensemble doit s'effectuer sur des échantillons distincts. Ambroise and McLachlan (2002) suggèrent de procéder à la validation croisée de façon externe à l'algorithme. Cette approche est retenue notamment par Svetnik et al. (2004) et Duan et al. (2005).

Sélection groupée

Lorsque les p variables explicatives sont naturellement regroupées, il est possible de sélectionner les variables d'un même groupe de façon jointe. Le Lasso groupé (ou *Group Lasso*) introduit par Yuan and Lin (2006a) fait office de méthode de référence dans ce contexte. Soit le modèle de régression linéaire

$$Y = \sum_{g=1}^{G} \mathbf{X}_g^\top \beta_g + \varepsilon,$$

où $\mathbf{X}_g$ est le vecteur des covariables appartenant au groupe g et β_g le vecteur de paramètres associé. Les vecteurs $\beta_1, \ldots, \beta_G$ sont estimés via le problème pénalisé

$$\hat{\beta} \in \underset{\beta \in \mathbb{R}^p}{\arg\min} \left\{ \|Y - \sum_{g=1}^{G} X_g^\top \beta_g\|_2^2 + \lambda \sum_{g=1}^{G} \sqrt{p_g} \|\beta_g\|_2 \right\},$$

avec p_g la taille du groupe g. Le terme $\|\beta_g\|_2$ est une pénalité intermédiaire entre la norme ℓ_1 et la norme ℓ_2 (Yuan and Lin; 2006a). Il agit donc de la même façon que la norme ℓ_1, c'est-à-dire qu'il induit la sélection des groupes entiers.

Plusieurs extension de la méthode existent, par exemple la régression logistique (Meier et al.; 2008; Kwemou; 2012), le *Sparse-Group Lasso* Zhou and Zhu (2010); Xiang et al.

(2013); Friedman et al. (2010); Simon et al. (2013) où l'on impose de la parcimonie intra-groupes en plus de la parcimonie des groupes eux mêmes. On peut citer également Zhao et al. (2009); Jacob et al. (2009) et Obozinski et al. (2009) qui étudient le cas où les groupes de variables ne forment pas une partition de l'ensemble des variables.

Sélection par méthode d'ensembles

L'instabilité est un problème récurent en sélection de variables et surtout dans le contexte de la grande dimension. En effet, une faible perturbation des données d'apprentissage peut radicalement changer l'ensemble des variables sélectionnées. Aussi, dans l'objectif d'identifier précisément les variables les plus influentes, il est nécessaire d'avoir un algorithme stable. Une solution, proposée par Bach (2008) puis par Meinshausen and Bühlmann (2010) est de répéter le processus de sélection sur des échantillons *bootstrap* et d'agréger les résultats. Si la solution naturelle proposée par Bach (2008) consiste à choisir l'intersection des différentes sélections, Meinshausen and Bühlmann (2010) randomisent les différentes sélections en affectant un poids aléatoire aux variables explicatives puis sélectionnent les variables selon leur fréquence moyenne de sélection. Ces méthodes ont été proposées pour le Lasso mais il est possible de stabiliser toute procédure de sélection de variables.

Variables corrélées

En pratique, les variables explicatives sont rarement indépendantes. L'analyse devient alors plus difficile. En effet, plus les variables sont corrélées, plus la structure de la loi jointe des variables explicatives est complexe. De plus, les procédures standards de sélection se trouvent déstabilisées, voir notamment l'article de Bühlmann et al. (2013). Les auteurs notent, qu'en cas de forte corrélations, le Lasso ne sélectionne qu'un représentant par groupe de variables corrélées sans tenir compte de l'"importance" de l'ensemble des variables du groupe. En conséquence, si deux variables sont fortement linéairement corrélées, la Lasso les sélectionnera indifféremment l'une ou l'autre.

Par ailleurs, la méthode *Elastic-net* Zou and Hastie (2005) est connue pour tenir compte de la corrélation entre les variables. Le terme de pénalité mixte ℓ_1/ℓ_2 a pour effet de sélectionner toutes les variables du groupe.

Notons enfin que Tolo̧si and Lengauer (2011) ont identifié numériquement un effet de la corrélation pour plusieurs méthodes dont la régression logistique avec une pénalité Lasso et les forêts aléatoires. La solution proposée est similaire à Bühlmann et al. (2013). Ils effectuent une étape de clustering sur les variables afin d'identifier des représentants par groupes de variables corrélées. Ils sélectionnent ensuite les clusters à travers leur représentant. Le Chapitre 4 est consacré à l'effet de la corrélation sur la sélection de variable avec les forêts aléatoires et s'inspire des résultats obtenus par Tolo̧si and Lengauer (2011).

2.3 Régression logistique

La régression logistique est une des méthodes les plus utilisées en classification. Si Y est à valeurs dans $\{-1, 1\}$, il s'agit d'inférer les probabilités *a posteriori* $\mathbb{P}[Y = 1|\mathbf{X} = \mathbf{x}]$ et $\mathbb{P}[Y = -1|\mathbf{X} = \mathbf{x}]$. Le modèle est défini par

$$\text{Logit}(\mathbb{P}[Y = 1|\mathbf{X} = \mathbf{x}]) = \beta^\top \mathbf{x}.$$

Le vecteur de paramètres β est estimé par la maximisation de la vraisemblance donnée par

$$\begin{aligned}\ell(\beta) &= \prod_{i=1}^{n} \mathbb{P}[Y = y_i | \mathbf{X} = \mathbf{x}_i]) \\ &= \prod_{i=1}^{n} \frac{1}{1 + e^{-y_i \beta^\top \mathbf{x}_i}}\end{aligned}$$

ou de façon équivalente en maximisant la log-vraisemblance

$$\mathcal{L}(\beta) = -\sum_{i=1}^{n} \log\left(1 + e^{-y_i \beta^\top \mathbf{x}_i}\right).$$

Maximiser la vraisemblance revient donc à minimiser le risque empirique pour la perte logistique $\log(1 + \exp(-yf(\mathbf{x})))$ et pour f linéaire.

Le problème de maximisation de la (log-)vraisemblance n'admet cependant pas de solution analytique. C'est pourquoi, des techniques d'optimisation numérique, par exemple l'algorithme de Newton-Raphson, sont employées pour obtenir un estimateur $\hat{\beta}$ et par suite un estimateur des probabilités a posteriori. Notons qu'un modèle logistique additif a été proposé comme une application aux modèles additifs généralisés par Hastie and Tibshirani (1986).

La sélection de variables avec le modèle logistique peut se faire par différentes méthodes. La première consiste à utiliser un algorithme "pas à pas" comme nous l'avons décrit dans la section précédente. Les critères d'information AIC, BIC ou le Cp de Mallows sont classiquement utilisés à cet effet. Une seconde approche vise à pénaliser la log-vraisemblance afin d'imposer de la parcimonie sur l'estimateur $\hat{\beta}$. Ce problème est étudié notamment par Zhu and Hastie (2004), van de Geer (2008), Meier et al. (2008) et Kwemou (2012) pour différentes pénalités de type Lasso.

2.4 Machines à vecteurs de support

Dans cette section, nous définissons tout d'abord le principe général des machines à vecteurs de support (Support Vector Machines en anglais, SVM) pour la classification. Nous abordons ensuite les techniques de sélection associées. Pour une introduction complète de la méthode, nous renvoyons le lecteur aux ouvrages Vapnik (1995), Cristianini and Shawe-Taylor (2000) et Shawe-Taylor and Cristianini (2004). Dans la suite, le produit scalaire dans $\mathbb{R}^p$ est noté $\langle w, \mathbf{x} \rangle_p = w^\top \mathbf{x}$.

2.4.1 Principe général

Les machines à vecteurs de support sont un ensemble de techniques d'apprentissage supervisé développé à partir des travaux de Vapnik durant les années 1960 sur les classifieurs linéaires. Le but est de construire un hyperplan séparateur dans l'espace des observations ($\mathbb{R}^p$ dans notre cas), c'est-à-dire de séparer linéairement les données des différentes classes.

Boser et al. (1992) introduisent l'idée de classifieur à marge maximale. Il s'agit de choisir parmi l'ensemble des hyperplans séparateurs celui qui maximise la distance entre l'hyperplan et les points les plus proches. Cette distance est appelée marge et les points les plus proches sont dits vecteurs de supports (voir la Figure 2.4).

L'intérêt majeur de cette méthode est qu'il est possible de généraliser la classification

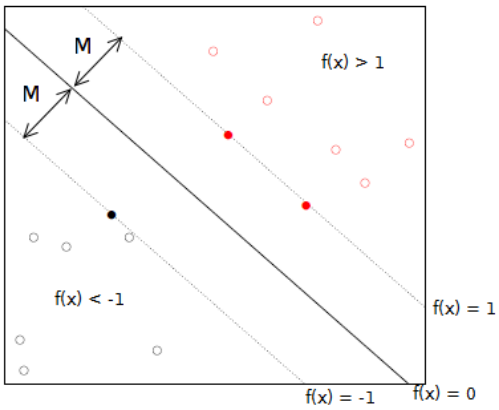


Figure 2.4 – Classification par SVM linéaire en 2 dimensions. Les vecteurs de support sont positionnés sur les marges (ronds pleins).

au cas où les données ne sont pas linéairement séparables. Dans ce cas, nous appliquons une transformation non linéaire des données afin de reconsidérer le problème dans un espace plus grand, dans lequel une séparation linéaire est possible. Nous présentons plus précisément la démarche de Boser et al. (1992) dans les paragraphes qui suivent.

Hyperplan séparateur optimal

Rappelons que nous observons un échantillon $\mathcal{D}_n = \{(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)\}$ pour lequel chaque Y_i prend ses valeurs dans l'ensemble $\{-1, 1\}$. Dans le cas linéaire, on souhaite trouver un classifieur $g : \mathbb{R}^p \longrightarrow \{-1, 1\}$ de la forme

$$g(\mathbf{x}) = \begin{cases} 1 & \text{si } f(\mathbf{x}) > 0 \\ -1 & \text{sinon} \end{cases}$$

avec $f(\mathbf{x}) = \langle w, \mathbf{x} \rangle_p + b$, w et b étant des paramètres inconnus à estimer. L'hyperplan $f(\mathbf{x}) = 0$ est appelé hyperplan séparateur. L'hyperplan séparateur optimal, construit à partir des données d'apprentissage, est celui dont les marges sont maximales (voir la Figure 2.4). La marge, notée M, est définie comme la distance entre l'hyperplan et les vecteurs de support. La distance d'un point $\mathbf{X}_i$ à l'hyperplan est donnée par

$$d(\mathbf{X}_i) = \frac{|Y_i f(\mathbf{X}_i)|}{\|w\|}.$$

Chaque donnée d'apprentissage $\mathbf{X}_i$ satisfait donc l'inégalité $d(\mathbf{X}_i) \geqslant M$ (Figure 2.4). En particulier, si $\mathbf{X}_i$ est un vecteur de support, alors $Y_i = f(\mathbf{X}_i) = 1$ et on a

$$d(\mathbf{X}_i) = M = \frac{1}{\|w\|}. \tag{2.4.1}$$

L'hyperplan optimal est donné par la résolution du problème

$$\begin{cases} \max\limits_{w,b} & M \\ t.q. & d(\mathbf{X}_i) \geqslant M, \quad i = 1, \dots, n, \end{cases} \tag{2.4.2}$$

ce qui est équivalent à

$$\begin{cases} \min\limits_{w,b} & \|w\| \\ t.q. & Y_i f(\mathbf{X}_i) \geqslant 1, \quad i = 1, \dots, n, \end{cases} \tag{2.4.3}$$

d'après l'Equation (2.4.1). Par suite, on obtient le problème de minimisation quadratique suivant, appelé problème primal :

$$\begin{cases} \min\limits_{w,b} & \frac{1}{2}\|w\|^2 \\ t.q. & Y_i f(\mathbf{X}_i) \geqslant 1, \quad i = 1, \dots, n. \end{cases} \tag{2.4.4}$$

La résolution de ce problème repose sur les conditions de Karush-Khun-Tucker (K.K.T.) qui assurent l'optimalité d'une solution dans le cas d'un problème de minimisation sous contraintes non linéaires. Nous renvoyons le lecteur au chapitre 5 du livre de Boyd and Vandenberghe (2004) pour davantage de détails sur ces conditions. Le Lagrangien associé au problème primal (2.4.4), noté $\mathcal{L}_P$, est donné par

$$\mathcal{L}_P(w, b, \alpha) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{n} \alpha_i [Y_i(\langle w, \mathbf{X}_i \rangle_p + b) - 1], \tag{2.4.5}$$

où α_i est le multiplicateur de Lagrange associé à la i-ème contrainte de (2.4.4) et où $\alpha = (\alpha_1, \dots, \alpha_n)$.

Supposons, à présent, qu'il existe un couple $(w^\star, b^\star)$ minimisant la fonction

$$(w, b) \longmapsto \mathcal{L}_P(w, b, \alpha).$$

D'après la théorie de la dualité, il existe un vecteur $\alpha^\star$ qui maximise la fonction (duale) $\mathcal{L}_D(\alpha) = \mathcal{L}_P(w^\star, b^\star, \alpha)$. Ce résultat est connu sous le nom de théorème de la dualité faible. Si de plus, le problème primal est strictement convexe, ce qui est le cas ici, alors le théorème de la dualité forte nous dit que les solutions des problèmes primal et dual sont égales et sont atteintes en un unique point $((w^\star, b^\star), \alpha^\star)$ (voir Boyd and Vandenberghe; 2004). Nous pouvons donc exprimer les SVM comme un problème de maximisation de la fonction $\mathcal{L}_D$ dont les solutions peuvent être explicitées au moyen de la première condition d'optimalité de K.K.T. pour une solution $(w^\star, b^\star)$:

$$\begin{cases} \dfrac{\partial \mathcal{L}_P}{\partial w}(w^\star, b^\star, \alpha) &= 0 \\ \dfrac{\partial \mathcal{L}_P}{\partial b}(w^\star, b^\star, \alpha) &= 0 \end{cases} \Longrightarrow \begin{cases} w^\star &= \sum_{i=1}^n \alpha_i Y_i \mathbf{X}_i \\ 0 &= \sum_{i=1}^n \alpha_i Y_i \end{cases}$$

En injectant ces équations dans (2.4.5), on obtient

$$\begin{aligned} \mathcal{L}_D(\alpha) &= \mathcal{L}_P(w^\star, b^\star, \alpha) \\ &= \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j Y_i Y_j \langle \mathbf{X}_i, \mathbf{X}_j \rangle_p, \end{aligned} \tag{2.4.6}$$

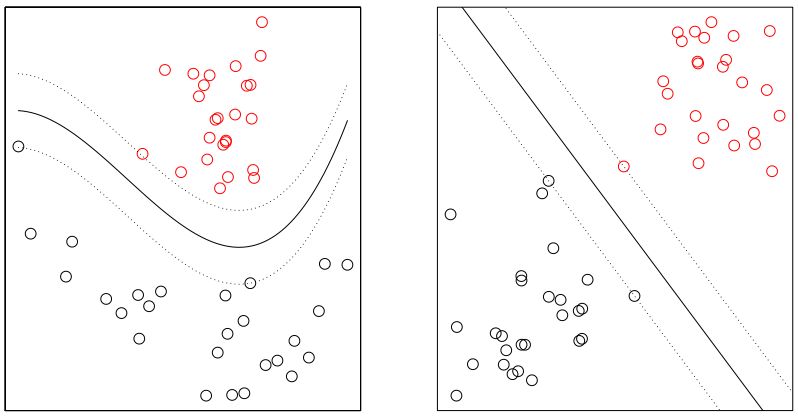


Figure 2.5 – Classification non linéaire par transformation de l'espace des observations (graphique de gauche) vers l'espace de redescription (graphique de droite).

que l'on doit maximiser sous la contrainte $\sum_{i=1}^{n} \alpha_i Y_i = 0$. Par ailleurs, la quatrième condition de K.K.T.

$$\alpha_i(Y_i f(\mathbf{X}_i) - 1) = 0, \quad i = 1, \ldots, n, \tag{2.4.7}$$

nous donne un résultat intéressant. Elle montre que tous les points qui ne sont pas des vecteurs de support ont un multiplicateur α_i nul. En effet, les vecteurs de support sont les points qui satisfont l'hypothèse $Y_i f(X_i) = 1$, les autres points sont tels que $Y_i f(\mathbf{X}_i) > 1$. Dans ce cas, l'équation (2.4.7) implique $\alpha_i = 0$. Une solution $w^\star = \sum_{i=1}^{n_S} \alpha_i Y_i \mathbf{X}_i$ est donc uniquement déterminée par les n_S vecteurs de support ($n_S \leqslant n$), en supposant qu'ils correspondent aux n_S premières observations.

Finalement, maximiser la fonction duale $\mathcal{L}_D$ sous la contrainte $\sum_{i=1}^{n} \alpha_i Y_i = 0$ permet de trouver une solution optimale du problème primal (w, b) avec w de la forme $w = \sum_{i=1}^{n_S} \alpha_i Y_i \mathbf{X}_i$. L'hyperplan séparateur de marge maximale est alors donné par $f(\mathbf{x}) = 0$ avec f de la forme

$$\begin{aligned} f(x) &= \langle w, x \rangle_p + b \\ &= \sum_{i=1}^{n_S} \alpha_i Y_i \langle \mathbf{X}_i, x \rangle_p + b. \end{aligned}$$

Cas non linéairement séparable

Dans le cas où il est impossible de trouver un hyperplan séparateur, Boser et al. (1992) proposent de projeter les données initialement dans $\mathbb{R}^p$ dans un espace de Hilbert $\mathcal{H}$ dans lequel la classification linéaire est possible. Cet espace, appelé espace de redescription (ou *feature space* en anglais), peut éventuellement être de dimension infinie. La projection se fait grâce à une fonction $\Phi : \mathbb{R}^p \longrightarrow \mathcal{H}$. Effectuer une classification non linéaire des données d'apprentissage $\{(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)\}$ revient à trouver un hyperplan séparateur optimal dans $\mathcal{H}$ à partir de l'échantillon $\{(\Phi(\mathbf{X}_1), Y_1), \ldots, (\Phi(\mathbf{X}_n), Y_n)\}$ (Figure 2.5). On parle souvent de nommée *kernel trick* pour désigner cette astuce.

On cherche donc une fonction de décision non linéaire de la forme

$$f(\mathbf{x}) = \langle w, \Phi(\mathbf{x}) \rangle_{\mathcal{H}} + b,$$

avec $\langle .,. \rangle_{\mathcal{H}}$ le produit scalaire de l'espace de Hilbert $\mathcal{H}$ et $w \in \mathcal{H}$.

En appliquant les outils du paragraphe précédent dans $\mathcal{H}$, les estimateurs de w et de f sont de la forme :

$$w = \sum_{i=1}^{n_S} \alpha_i Y_i \Phi(\mathbf{X}_i)$$

et

$$f(\mathbf{x}) = \sum_{i=1}^{n_S} \alpha_i Y_i \langle \Phi(\mathbf{X}_i), \Phi(\mathbf{x}) \rangle_{\mathcal{H}} + b. \tag{2.4.8}$$

Le calcul de la fonction Φ peut s'avérer très couteux dès lors que la dimension de l'espace $\mathcal{H}$ est importante (en particulier s'il est de dimension infinie). Mais comme le montre l'équation (2.4.8), l'expression de f ne dépend des observations $\mathbf{X}_i$ qu'à travers le produit scalaire $k(\mathbf{X}_i, \mathbf{x}) := \langle \Phi(\mathbf{X}_i), \Phi(\mathbf{x}) \rangle_{\mathcal{H}}$. La fonction k, appelée fonction noyau, permet d'effectuer les calculs dans l'espace de départ $\mathbb{R}^p$ au lieu de calculer un produit scalaire dans un espace plus grand. Il n'est donc pas nécessaire de connaître l'application Φ ainsi que l'espace $\mathcal{H}$ pour résoudre le problème de classification non linéaire. En réalité, nous savons que $\mathcal{H}$ muni du produit scalaire $\langle .,. \rangle_{\mathcal{H}}$ est un espace de Hilbert à noyau reproduisant (ou RKHS). La définition suivante est tirée du livre de Schölkopf and Smola (2001).

Définition 1 (Espace de Hilbert à noyau auto-reproduisant)**.** Soit $\mathcal{H}$ un espace de Hilbert de fonctions définies sur un ensemble non vide $\mathcal{X}$ et à valeurs dans $\mathbb{R}$.

1. Une fonction noyau $k : \mathcal{X} \times \mathcal{X} \longrightarrow \mathbb{R}$ est appelée noyau auto-reproduisant de $\mathcal{H}$ s'il vérifie les conditions
   - $\forall \mathbf{x} \in \mathcal{X}, k(., \mathbf{x}) \in \mathcal{H}$
   - $\forall \mathbf{x} \in \mathcal{X}$ et $\forall f \in \mathcal{H}, \langle f, k(., \mathbf{x}) \rangle_{\mathcal{H}} = f(\mathbf{x})$.
2. L'espace $\mathcal{H}$ est un espace de Hilbert à noyau auto-reproduisant (ou RKHS) s'il est engendré par un noyau auto-reproduisant.

Le théorème qui suit donne l'existence et l'unicité d'un RKHS. Précisons qu'un noyau k est symétrique défini positif si pour $\mathbf{x}, \mathbf{y} \in \mathcal{X}$,

$$k(\mathbf{x}, \mathbf{y}) = k(\mathbf{y}, \mathbf{x}),$$

et si pour tout suite de réels $a_1, \ldots, a_p$,

$$\sum_{i,k \geq 1}^{p} a_i a_k k(\mathbf{x}_i, \mathbf{y}_k) \geq 0.$$

Théorème 1 (Aronszajn; 1950)**.** *Soit $k : \mathcal{X} \times \mathcal{X} \longrightarrow \mathbb{R}$ un noyau symétrique, défini, positif. Il existe alors un unique RKHS de fonctions définies sur $\mathcal{X}$ pour lequel k est reproduisant.*

Le théorème de représentation de Mercer (1909) donne les conditions sur une fonction k définie positive pour être une fonction noyau et permet d'identifier une décomposition spectrale de k à partir d'éléments de $\mathcal{H}$.

Théorème 2 (Mercer; 1909)**.** *Si k est un noyau symétrique défini positif vérifiant la condition*

$$\int \int k^2(\mathbf{x}, \mathbf{y}) d\mathbf{x} d\mathbf{y} < \infty,$$

pour $\mathbf{x}, \mathbf{y} \in \mathcal{X}$*, alors il existe une base* $\{\varphi_1, \varphi_2, \ldots\}$ *de fonctions orthogonales et une suite de réels positifs* $\lambda_1, \lambda_2, \ldots$ *satisfaisant* $\sum_{k\geq 1} \lambda_k^2 < \infty$ *tels que*

$$k(\mathbf{x}, \mathbf{y}) = \sum_{k\geq 1} \lambda_k \varphi_k(\mathbf{x}) \varphi_k(\mathbf{y}).$$

Issu de l'analyse des opérateurs intégraux, ce théorème permet de décrire l'espace $\mathcal{H}$. Il est aussi couramment utilisé pour spécifier une représentation de la fonction de covariance d'un processus stochastique dans l'analyse des données fonctionnelles, voir Horváth and Kokoszka (2012, Chap. 2).

En toute rigueur, il faudrait trouver un noyau s'adaptant à la structure des données afin d'obtenir une procédure de classification optimale. Mais nous avons en général peu d'informations sur la structure des données. C'est pourquoi un ensemble de noyaux "simples" sont couramment employés (Cristianini and Shawe-Taylor; 2000) :

- le noyau linéaire : $k(\mathbf{x}, \mathbf{y}) = \langle \mathbf{x}, \mathbf{y} \rangle_p$;
- le noyau polynomial d'ordre d : $k(\mathbf{x}, \mathbf{y}) = (\langle \mathbf{x}, \mathbf{y} \rangle_p + c)^d, c \geq 0$;
- le noyau Gaussien de paramètre $\sigma^2 > 0$: $k(\mathbf{x}, \mathbf{y}) = e^{-\frac{\|\mathbf{x}-\mathbf{y}\|^2}{2\sigma^2}}$;
- le noyau ANOVA : $k(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^n \exp(-\sigma(x_i - y_i)^2)^d$.

Le noyau Gaussien est en général choisi car il donne des performances acceptables en pratique. Le noyau ANOVA est bien adapté aux problèmes de régression comme le montre Stitson et al. (1997). Néanmoins, la structure des données que l'on étudie peut conduire à un choix différent, notamment dans le cas de séries temporelles, de chaînes de caractères ou de graphes. La construction d'un noyau adapté aux données peut se faire par le choix d'une mesure de similarité ou bien en combinant un ensemble de noyaux "simples". Dans Shawe-Taylor and Cristianini (2004), les auteurs font une revue des noyaux existants pour différentes structures de données.

Marge souple

Dans certaines situations, il est difficile de trouver un hyperplan séparateur, que ce soit dans l'espace des observations ou dans l'espace de redescription. Cortes and Vapnik (1995) se sont intéressés à cette situation et considèrent que l'on ne peut pas toujours séparer linéairement sans faire d'erreurs de classement. L'hyperplan séparateur doit donc être optimisé de sorte à minimiser le nombre d'observations mal classées. Dans cet objectif, les auteurs introduisent des variables "ressort" (*slack variables* en anglais) positives $\xi_1, \ldots, \xi_n$ qui permettent de pénaliser le taux d'erreur. Le problème des SVM est dans ce cas défini par

$$\begin{cases} \min\limits_{w,b,\xi} & \|w\|^2 + C \sum\limits_{i=1}^n \xi_i \\ t.q. & Y_i(\langle w, \Phi(\mathbf{X}_i) \rangle_{\mathcal{H}} + b) \geqslant 1 - \xi_i, \quad i = 1, \ldots, n, \\ & \xi_i \geqslant 0 \end{cases} \tag{2.4.9}$$

où C est une constante strictement positive permettant de contrôler l'erreur induite par le terme de pénalité. Plus C est grand, plus la marge est réduite au prix d'un taux d'erreur de classification plus important. En pratique, cette constante est choisie par validation croisée. La présence du terme $C \sum_{i=1}^p \xi_i$ ne change pas la forme de l'hyperplan séparateur optimal donnée par l'équation (2.4.8). En effet, lorsque l'on écrit le problème sous une forme

lagrangienne et en optimisant le paramètre w, les termes faisant intervenir les variables ξ_i sont supprimés et on obtient

$$f(x) = \sum_{i=1}^{n} \alpha_i Y_i k(\mathbf{X}_i, \mathbf{x}) + b,$$

où $0 \leqslant \alpha_i \leqslant C$ pour tout $i = 1, \ldots, n$.

Estimation par critère pénalisé

Il est intéressant de noter que l'on peut considérer les SVM comme une méthode d'estimation par critère pénalisé. Le critère à minimiser, tiré de l'article de Evgeniou et al. (2000), est donné par

$$R_{reg}(f) = \frac{1}{n} \sum_{i=1}^{n} |1 - Y_i f(\mathbf{X}_i)|_+ + \lambda \|f\|^2_{\mathcal{H}},$$

dans lequel apparaît la fonction de perte *hinge* $|1 - Yf(\mathbf{X})|_+ = \max(0, 1 - Yf(\mathbf{X}))$ (terme empirique) ainsi qu'une pénalité *Ridge* donnée par la norme de f dans le RKHS $\mathcal{H}$. Cette formulation des SVM est équivalente au problème (2.4.9). Pour s'en convaincre, il suffit de remarquer que la contrainte

$$Y_i f(\mathbf{X}_i) \geqslant 1 \tag{2.4.10}$$

est vérifiée si et seulement si $|1 - Yf(\mathbf{X})|_+ = 0$. Donc minimiser le critère $R_{reg}(f)$ revient à minimiser la norme de f tout en s'assurant qu'en moyenne, la contrainte (2.4.10) soit satisfaite.

Les SVM pour la régression peuvent également être définis par la minimisation d'un critère pénalisé à partir de la perte ε-*insensitive* $\ell(y, g(\mathbf{x})) = |y - g(\mathbf{x})|_\varepsilon = \max(0, |y - g(\mathbf{x})| - \varepsilon)$.

Cas de données déséquilibrées

Dans le cas où les données étudiées sont déséquilibrées, c'est-à-dire lorsque la proportion des labels positifs est très différente de celle des labels négatifs, les performances de la classification se trouvent altérées. En effet, l'estimation de l'hyperplan séparateur optimal minimise le taux de données mal classées sans prendre en compte la proportion de chaque classe ce qui induit une erreur de classification très faible pour la classe majoritaire mais très forte pour la classe minoritaire (parfois égale à 1). Cela pose problème si la classe à prédire est la moins représentée. C'est précisément le cas pour l'analyse des atterrissages longs.

Plusieurs propositions ont été faites pour prendre en compte le déséquilibre des données. Osuna et al. (1997) ainsi que Veropoulos et al. (1999) proposent de pénaliser le taux de mal classés différemment selon la classe. Nous ajoutons au problème d'optimisation deux constantes de coût C^- et C^+ respectivement pour la classe négative et positive :

$$\begin{cases} \min\limits_{w,b,\xi} & \|w\|^2 + C^+ \sum\limits_{i:Y_i=1} \xi_i + C^- \sum\limits_{i:Y_i=-1} \xi_i \\ s.t. & Y_i(\langle w, \Phi(\mathbf{X}_i)\rangle_{\mathcal{H}} + b) \geqslant 1 - \xi_i, \quad i = 1, \ldots, n. \\ & \xi_i \geqslant 0 \end{cases}$$

Les constantes C^- et C^+ définissent une pondération sur chaque classe. Cette technique est implémentée dans la plupart des modules dédiés aux SVM. C'est l'approche que nous retiendrons dans la Section 2.7. D'autres propositions ont été faites comme par exemple celle

de Yan et al. (2003) qui utilisent les SVM dans une méthode de type Bagging (Breiman; 1996). Les auteurs construisent à partir de l'échantillon d'apprentissage, K sous-ensembles contenants toutes les observations minoritaires et un sous-échantillon tiré aléatoirement de l'ensemble des données majoritaires. Une collection de K classifieurs est ainsi construite et le classifieur final est obtenu par le vote majoritaire des éléments de la collection.

Probabilités *a posteriori*

À l'inverse de la régression logistique, les SVM n'estiment pas directement les probabilités *a posteriori* $\eta(\mathbf{x}) = \mathbb{P}[Y = 1|\mathbf{X} = \mathbf{x}]$ et $1 - \eta(\mathbf{x})$. Plusieurs méthodes ont été proposées pour estimer ces quantités. Hastie and Tibshirani (1998) proposent d'estimer $\eta(\mathbf{x})$ en supposant que les lois conditionnelles $P_{\mathbf{X}|Y=1}$ et $P_{\mathbf{X}|Y=-1}$ sont Gaussiennes. L'estimation de $\eta(\mathbf{x})$ revient alors à estimer les paramètres de lois gaussiennes et à utiliser la formule de Bayes

$$\eta(\mathbf{x}) = \frac{p_{\mathbf{X}|Y=1}(\mathbf{x})\mathbb{P}(Y=1)}{p_{\mathbf{X}|Y=1}(\mathbf{x})\mathbb{P}(Y=1) + p_{\mathbf{X}|Y=-1}(\mathbf{x})\mathbb{P}(Y=-1)},$$

où $p_{\mathbf{X}|Y=i}$ et $\mathbb{P}(Y = i)$ sont respectivement la densité conditionnelle et la probabilité *a priori* de la classe i. Cette approche est discutable puisque l'hypothèse gaussienne effectuée sur les distributions conditionnelles n'est pas toujours vérifiée en pratique. C'est pourquoi nous avons choisi de retenir l'approche de Platt (1999) qui consiste à estimer $\eta(\mathbf{x})$ à partir des valeurs de la fonction de décision f donnée par l'optimisation des SVM. L'auteur utilise un modèle paramétrique similaire au modèle logistique au lieu d'estimer directement des densités conditionnelles, c'est-à-dire

$$\hat{\eta}(\mathbf{x}) = \frac{1}{1 + \exp(Af(\mathbf{x}) + B)},$$

où A et B sont des paramètres estimés par maximum de vraisemblance.

2.4.2 Sélection de variables

Rappelons que le vecteur des variables explicatives s'écrit $\mathbf{X} = (X_1, \ldots, X_p)$ et que $(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)$ sont n couples i.i.d. de $(\mathbf{X}, Y)$.

Guyon et al. (2002) ont proposé un algorithme de sélection de variables qu'ils nomment SVM-RFE pour SVM *Recursive Feature Elimination*. Cet algorithme récursif *backward* a été établi pour une application à la sélection de gènes. Il est inspiré des algorithmes *greedy* proposés par Kohavi and John (1997). L'algorithme SVM-RFE vise à trouver un sous-ensemble de variables maximisant les performances du classifieur SVM, c'est-à-dire minimisant $||w||^2$. La procédure, détaillée dans l'Algorithme 2, élimine récursivement les variables qui minimisent un certain critère de rang R_C. Chaque sous-ensemble ainsi sélectionné dans la procédure backward est évalué au moyen de la procédure *Leave-One-Out* (loo) : un modèle SVM est optimisé successivement sur $n-1$ observations et est testé sur l'observation restante. L'estimation de l'erreur est alors l'erreur moyenne des n prédictions effectuées. Finalement, le sous-ensemble de variables optimal est celui dont l'erreur loo est minimale.

Le critère de rang utilisé par Guyon et al. (2002) est directement issu de l'optimisation du problème des SVM linéaires. Donné par $R_c(X_j) = w_j^2$, il mesure l'augmentation de $||w||^2$ lorsque la variable est retirée du modèle. En effet, on peut montrer que

$$w_j^2 = ||w||^2 - ||w_{(j)}||^2,$$

Algorithm 2 SVM-RFE

1: Optimiser le vecteur de poids w
2: Calculer le critère de rang $R_c(X_j)$ pour tout $j \in \{1, \dots, p\}$
3: Éliminer la variable qui minimise $R_c(X_j)$
4: Répéter les étapes 1 à 3 tant qu'il reste des variables

où $w_{(j)}$ est le vecteur de poids optimisé après retrait de la variable X_j. Dans le cas non linéaire, on a

$$||w||^2 = \alpha^\top H \alpha,$$

avec $\alpha \in \mathbb{R}^n$ le vecteur des multiplicateurs de Lagrange du problème dual et H la matrice de dimension $n \times n$ telle que $H_{ik} = Y_i Y_k k(\mathbf{X}_i, \mathbf{X}_k)$, pour $i, k \in \{1, \dots, n\}$. Le critère non linéaire proposé par Guyon et al. (2002) est le suivant

$$R_c(X_j) = \frac{1}{2}\left(\alpha^\top H \alpha - \alpha^\top H_{(j)} \alpha\right),$$

où $H_{(j)}$ est la matrice H calculée après retrait de la variable X_j. Dans Rakotomamonjy (2003), l'auteur propose d'autres critères de rang pour l'algorithme SVM-RFE.

Afin d'améliorer les performances algorithmiques de l'algorithme RFE, notamment en très grande dimension (par exemple pour la sélection de gènes où plusieurs milliers de variables sont présentes), il est possible de retirer un groupe de variables à chaque itération. C'est l'approche que nous choisissons dans l'application aux données de vol : l'algorithme élimine 20 % des variables à chaque itération jusqu'à réduire le nombre de variables de moitié puis élimine une variable à chaque itération ensuite.

Signalons également les travaux de Zhu et al. (2004) ainsi que ceux de Bi et al. (2003) qui ont proposé une méthode SVM *sparse* basée un terme de pénalisation ℓ_1 du vecteur w. Ceci permet d'introduire une sélection automatique des variables suivant le même principe que le LASSO défini par Tibshirani (1996) dans le cas de la régression linéaire.

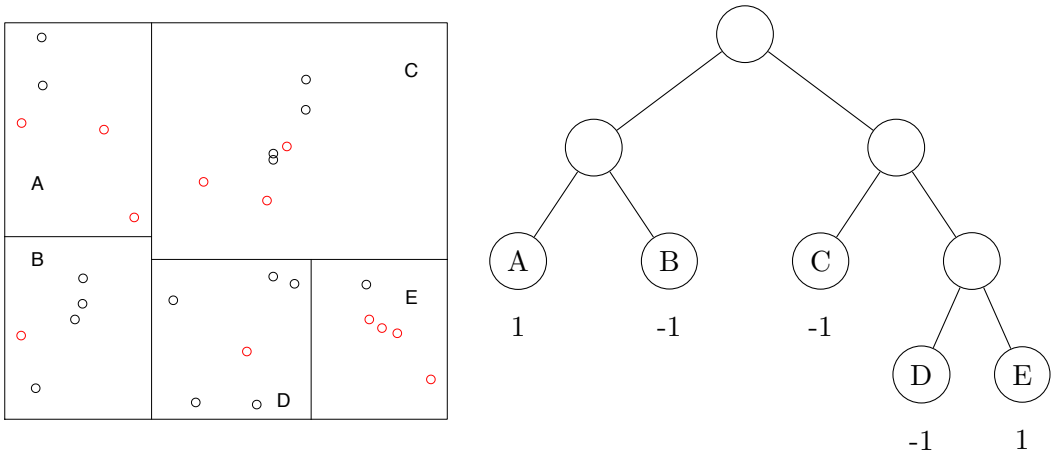
2.5 Forêts aléatoires

Cette section présente une deuxième méthode de classification non linéaire, les forêts aléatoires, introduite par Breiman (2001). Nous comparerons par la suite les forêts aléatoires, les SVM et la régression logistique pour la prédiction des atterrissages longs.

L'algorithme des forêts aléatoires est une variante du bagging où est agrégé un ensemble d'arbres aléatoires proches de la méthode CART (Breiman et al.; 1984). Utilisable à la fois en régression et en classification, cet algorithme a montré de très bonnes performances en pratique notamment pour des problèmes complexes (relations non linéaires, interactions, grande dimension, etc.).

2.5.1 Arbre de décision

Les arbres de décision, ou arbres aléatoires, sont un ensemble de techniques permettant de construire un classifieur en partitionnant l'espace des observations de façon récursive. La découpe récursive se fait de façon dyadique ce qui donne une structure d'arbre binaire à l'objet final (Figure 2.6). Le premier nœud de l'arbre est la racine et les éléments les plus bas sont les feuilles et constituent la partition de l'espace des observations. Il existe plusieurs algorithmes de construction des arbres aléatoires dont le plus connu est l'algorithme CART (*Classification And Regression Trees*, Breiman et al.; 1984). L'algorithme CART procède



(a) (b)

Figure 2.6 – Classification par arbre. À chaque feuille est associée le vote majoritaire des éléments qu'elle contient (-1 pour prédire noir et 1 pour prédire rouge).

en deux étapes pour construire un arbre de décision optimal : la phase d'expansion et la phase d'élagage.

La racine de l'arbre contient toutes les observations $\mathcal{D}_n$. L'algorithme CART recherche la meilleure découpe possible parmi toutes les variables explicatives. Autrement dit, il construit deux sous-parties N_1 et N_2 (les nœuds fils) comme suit :

$$\begin{aligned} N_1 &= \{\mathbf{X}_i, \mathbf{X}_{ij^\star} \leqslant d^\star\}, \\ N_2 &= \{\mathbf{X}_i, \mathbf{X}_{ij^\star} > d^\star\}. \end{aligned}$$

Le couple $(j^\star, d^\star)$ est choisi de sorte que chaque nœud fils soit le plus homogène possible. L'homogénéité d'un nœud N se mesure par l'indice de Gini $\sum_{k=1}^{K} \hat{p}_N^k(1-\hat{p}_N^k)$, où $\hat{p}_N^k$ est la proportion des éléments de classe k dans le nœud N. L'algorithme CART cherche donc à minimiser l'indice de Gini dans l'ensemble des découpes possibles. Une fois la racine ainsi partitionnée, la procédure est répétée pour chacun de ses fils jusqu'à ce que le chaque nœud ne contienne qu'un seul élément ou bien des observations de même classe. L'arbre ainsi obtenu est appelé arbre maximal (noté T_{max}) et les derniers nœuds construits sont les feuilles. À chaque feuille est associée une prédiction définie par la classe majoritaire des observations qu'elle contient. Le prédicteur de l'arbre est alors l'histogramme des prédictions de chaque feuille, c'est-à-dire la fonction

$$\hat{f}_T(x) = \sum_{m=1}^{|T|} k(m) \mathbb{1}_{x \in N_m},$$

avec $k(m) \in \arg\max_k \hat{p}_{N_m}^k$ la classe majoritaire du nœud N_m et $|T|$ le nombre de feuilles de l'arbre. Notons par ailleurs que l'algorithme CART est adapté à la régression en calculant la moyenne empirique des valeurs de Y_i dans chaque feuille.

L'arbre maximal T_{max} ainsi construit a l'avantage d'avoir un biais faible, mais il a

cependant une variance élevée. C'est pourquoi il existe une seconde étape dans l'algorithme CART pour optimiser les performances d'un arbre maximal en construisant un sous-arbre qui réalise le compromis biais-variance ; c'est la phase d'élagage. Une suite de sous-arbres emboités de l'arbre maximal est construite en minimisant un critère pénalisé. Parmi cette collection d'arbres, l'arbre optimal est celui qui admet les meilleures performances. Il est en général obtenu par validation croisée (voir Gey and Nédélec; 2005 ainsi que les thèses de Gey; 2002 et de Tuleau; 2005 pour un panorama complet sur le sujet).

2.5.2 Forêts aléatoires de Breiman

Même si la phase d'élagage améliore les performances d'un arbre de classification en terme de biais et de variance, l'algorithme CART reste une technique instable. En effet, une simple permutation de deux observations de l'ensemble d'apprentissage peut produire un arbre très différent. Les forêts aléatoires de Breiman (2001) permettent de résoudre cette faiblesse de l'algorithme CART et en améliorent les performances. Plus précisément, les forêts aléatoires sont une variante du Bagging (Breiman; 1996) où la règle de décision est un arbre aléatoire. On construit M arbres aléatoires $\hat{f}_1, \dots, \hat{f}_M$ sur des échantillons *bootstrap* $\mathcal{D}_n^1, \dots, \mathcal{D}_n^M$ contenant des observations tirées aléatoirement (avec ou sans remise) dans $\mathcal{D}_n$. À la différence de CART, un petit nombre de variables est choisi aléatoirement à chaque nœud pour déterminer la meilleure découpe possible. Par défaut, le nombre de variables choisies à chaque nœud est de $\sqrt{p}$ pour la classification et de $p/3$ pour la régression. Les arbres ainsi randomisés sont pleinement développés et ne sont pas élagués ce qui permet la construction d'une collection variée de classifieurs. L'estimateur final est donné par l'agrégation de ces estimateurs, soit le vote majoritaire dans le cas de la classification et la moyenne empirique pour la régression.

Il existe peu de résultats théoriques sur les forêts aléatoires ce qui est notamment du à la complexité de la procédure. C'est pourquoi des études ont été menées dans des cadres simplifiés, voir par exemple Breiman (2004), Biau et al. (2008), Biau (2012), Genuer (2012) et Zhu et al. (2012). Très récemment, Scornet et al. (2014) ont établi un premier résultat de convergence de l'algorithme de Breiman pour le modèle de régression additive.

Erreur Out-of-bag

Les algorithmes de type bagging – non nécessairement les forêts aléatoires – proposent une estimation de l'erreur de prédiction en tirant parti de l'information apportée par les différents estimateurs agrégés (Breiman; 1997). Chaque échantillonnage *bootstrap* laisse de coté un certain nombre d'observations (environ un tiers en pratique) qui sont utilisées pour calculer l'erreur Out-of-bag (OOB), pour *Out of bagging* ("en dehors du *bootstrap*").

La procédure d'estimation se formalise comme suit. Soit $\bar{\mathcal{D}}_n^m = \mathcal{D}_n \setminus \mathcal{D}_n^m$ avec $m = 1, \dots, M$, les échantillons OOB constitués des observations non retenues dans les ensembles *bootstrap*. Pour chaque donnée $(\mathbf{X}_i, Y_i)$, un classifieur $\hat{f}_i^{oob}$ est construit en agrégeant les arbres ne contenant pas $(\mathbf{X}_i, Y_i)$. Une prédiction est alors donnée par $\hat{Y}_i = \hat{f}_i^{oob}(\mathbf{X}_i)$ et par suite l'erreur OOB de la forêt :

$$\hat{R}^{oob} = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}_{\hat{Y}_i \neq Y_i}.$$

L'erreur OOB estime l'erreur de classification au même titre que la validation croisée *Leave-One-Out* au sens où cette erreur est calculée à partir des observations qui ne sont pas utilisées pour l'estimation des $\hat{Y}_i$. L'avantage de cette méthode comparée aux tech-

niques de validation croisée est qu'elle s'effectue durant le processus de construction de la forêt et ne nécessite pas de développements algorithmiques supplémentaires. Néanmoins, il est connu dans la littérature que cette procédure d'estimation de l'erreur est légèrement optimiste, c'est-à-dire qu'elle a tendance à sous évaluer l'erreur de prédiction comme le souligne Breiman (2001). En conséquence, il est préférable d'utiliser des techniques telles que la validation croisée pour estimer l'erreur avec précision. Il est cependant tout à fait envisageable d'utiliser l'erreur OOB pour comparer des classifieurs entre eux comme le fait par exemple Genuer et al. (2010) dans un contexte de sélection de variables.

Cas de données déséquilibrées

Comme nous l'avons mentionné précédemment, si la proportion des labels positifs est très différente de celle des labels négatifs, les performances de la classification peuvent être altérées. Par exemple, le taux d'erreur peut être très faible pour la classe majoritaire mais à l'inverse très élevé pour la classe minoritaire. Il est donc nécessaire d'adapter les forêts aléatoires pour des données déséquilibrées. Plusieurs approches existent pour résoudre ce problème. La première consiste à forcer l'algorithme à construire des échantillons *bootstrap* équilibrés soit en sous-échantillonnant la classe majoritaire, soit en sur-échantillonnant les données de la classe minoritaire par un tirage aléatoire avec remise (Chen et al.; 2004). Sous-échantillonner la classe majoritaire semble une bonne approche si cela n'induit pas de perte d'information ou de la structure des données. C'est la démarche que nous choisissons dans la Section 2.7.

Notons que Chawla et al. (2002) proposent un sur-échantillonnage intelligent à travers une méthode générale qu'ils nomment SMOTE pour Synthetic Minority Over-sampling TEchnique. Au lieu de répliquer les observations minoritaires, ils créent, pour chaque observation minoritaire, de nouvelles données synthétiques dans la direction des k plus proches voisins de même classe. Les auteurs combinent cette technique à un sous-échantillonnage des données majoritaires afin d'améliorer les performances et testent la procédure avec l'algorithme C4.5.

Probabilités *a posteriori*

Comme nous l'avons écrit, chaque observation est classifiée selon le vote majoritaire des prédictions de chaque arbre de la forêt. La proportion des votes fournit donc une estimation des probabilités *a posteriori*, simple et rapide à calculer.

2.5.3 Importance des variables et algorithmes de sélection

Les arbres aléatoires sont une règle de décision interprétable pour la prédiction des classes. En effet, les variables de découpe sont choisies selon leur capacité à classifier les données. De ce fait, une variable permettant de découper les premiers nœuds a un pouvoir discriminant plus important qu'une variable apparaissant dans les dernières découpes (ou n'apparaissant pas dans l'arbre). Les arbres fournissent donc naturellement un critère de sélection des variables pour la prédiction de Y.

Dans les forêts aléatoires, l'interprétation est perdue du fait de la randomisation des arbres. Afin de pallier la perte d'information, plusieurs critères d'importance sont définis pour la prédiction de Y. Nous pouvons citer les plus connues, l'importance de Gini qui mesure la décroissance moyenne de l'indice de Gini et la mesure d'importance par permutation introduite par Breiman (2001) et que l'on détaille ici. Ce critère sera utilisé dans la suite du manuscrit.

Une variable X_j est considérée comme importante si en cassant le lien entre X_j et Y, l'erreur de prédiction augmente. Pour casser ce lien, Breiman propose de permuter aléatoirement les réalisations de X_j dans les échantillons OOB. La permutation a également pour effet de casser le lien entre X_j et les autres variables. Plus formellement, l'erreur de prédiction de chaque arbre $\hat{f}_m$ est évaluée à partir de son échantillon OOB associé par l'estimateur empirique

$$\hat{R}(\hat{f}_m, \bar{\mathcal{D}}_n^m) = \frac{1}{|\bar{\mathcal{D}}_n^m|} \sum_{i:(\mathbf{X}_i, Y_i) \in \bar{\mathcal{D}}_n^m} \ell\left(Y_i, \hat{f}_m(\mathbf{X}_i)\right).$$

Définissons ensuite une collection d'ensemble OOB permutés $\{\bar{\mathcal{D}}_n^{mj},\, m \in \{1, \dots, M\}\}$ en permutant aléatoirement les valeurs de la variable j dans chaque ensemble OOB. La mesure d'importance par permutation est alors définie par l'augmentation moyenne de l'erreur de prédiction sur l'ensemble des arbres

$$\hat{\mathcal{I}}(X_j) = \frac{1}{M} \sum_{m=1}^{M} \left[\hat{R}(\hat{f}_m, \bar{\mathcal{D}}_n^{mj}) - \hat{R}(\hat{f}_m, \bar{\mathcal{D}}_n^m)\right]. \tag{2.5.1}$$

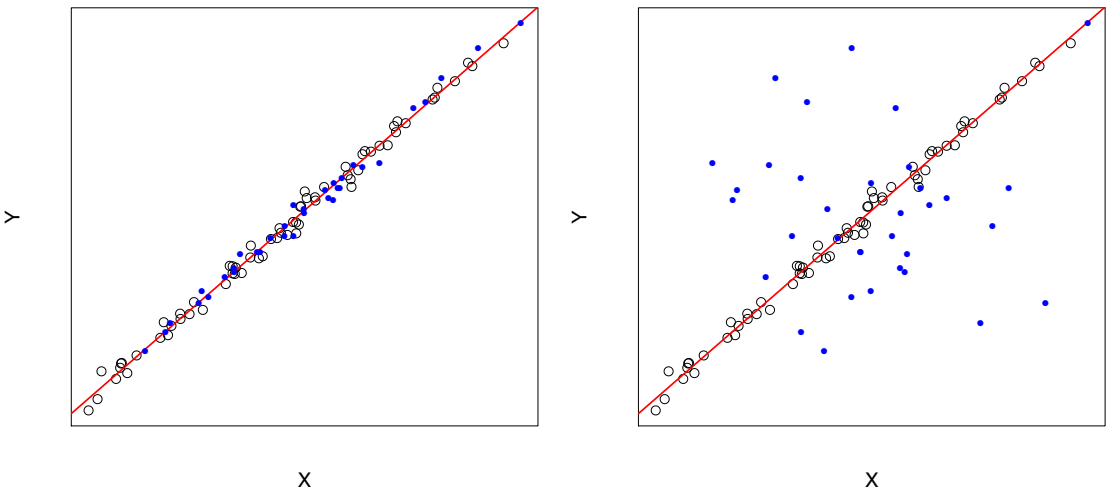
Ce critère permet de mesurer la sensibilité de l'erreur commise par chaque arbre suite à la permutation des valeurs de X_j. Si la permutation aléatoire induit une forte augmentation de l'erreur alors le critère (2.5.1) sera grand en moyenne (voir Figure 2.7). À l'inverse, si la permutation n'a aucun effet sur l'erreur, alors l'importance de X_j sera proche de zéro. Il est à noter que cette mesure peut être légèrement négative. Ceci s'explique par fait que les termes d'erreur sont des estimateurs empiriques. Ils sont donc sujets à une certaine variance. Pour des variables non informatives, la différence entre ces deux termes varie autour de zéro. Notons de plus qu'une version normalisée de cette mesure existe, le z-score, où $\hat{\mathcal{I}}(X_j)$ est divisé par son écart type.

Un grand nombre d'auteurs se sont intéressés à l'étude numérique de cette mesure d'importance, voir Strobl and Zeileis (2008), Archer and Kimes (2008), Nicodemus et al. (2010), Altmann et al. (2010), Nicodemus (2011) et Auret and Aldrich (2011). À ce jour, il existe peu de résultats théoriques. Une première étude a été menée par Ishwaran (2007) sur une version simplifiée du critère de Breiman. Zhu et al. (2012) vont plus loin dans le formalisme. Ils définissent le critère empirique $\hat{\mathcal{I}}(X_j)$ comme un estimateur de

$$\mathcal{I}(X_j) = \mathbb{E}\ell\left(Y, f(\mathbf{X}_{(j)})\right) - \mathbb{E}\ell\left(Y, f(\mathbf{X})\right), \tag{2.5.2}$$

où le vecteur aléatoire $\mathbf{X}_{(j)} = (X_1, \dots, X_j', \dots, X_p)$ est tel que X_j' est une copie indépendante de X_j, également indépendante de Y et des autres covariables. Zhu et al. (2012) étudient la régression avec la perte quadratique. Nous donnons ici une définition plus générale, avec une fonction de perte ℓ quelconque. Le remplacement de X_j par X_j' dans (2.5.2) imite la permutation aléatoire dans les ensembles OOB. Zhu et al. (2012) montrent en particulier que $\hat{\mathcal{I}}(X_j)$ converge vers $\mathcal{I}(X_j)$ avec vitesse exponentielle pour une version simplifiée des forêts aléatoires. Notons que leur résultat est démontré pour le z-score mais il est aussi valable pour $\hat{\mathcal{I}}(X_j)$. Dans le Chapitre 4, nous étudions théoriquement le comportement de l'importance par permutation dans le cas où les variables explicatives sont corrélées et pour un modèle de régression additive.

D'autres mesures d'importance ont été introduites : Strobl et al. (2008) proposent de permuter les variables conditionnellement aux variables qui lui sont corrélées. La mesure proposée par Janitza et al. (2013) est basée sur le score AUC (pour Area Under the Curve)



(a) Avant permutation (b) Après permutation

Figure 2.7 – Augmentation de l'erreur après permutation des valeurs d'une variable X pour l'échantillon de validation (points bleus). Les points noirs représentent l'ensemble *bootstrap*. Les graphes on été faits en régression pour mieux visualiser le phénomène soulevé.

à la place de l'erreur de prédiction dans (2.5.1) afin de mieux s'adapter aux données déséquilibrées.

Les mesures d'importance des variables sont généralement utilisées dans l'objectif d'identifier celles qui portent le plus d'information prédictive. Plusieurs utilisations sont possibles dont la plus naïve qui consiste à filtrer les variables selon le classement induit par le critère. On parle de *screening* dans ce cas. Comme nous l'avons écrit précédemment, cette vision peut être sous-optimale au sens où le choix final du sous-ensemble de variables ne tient pas compte des performances de la méthode de classification. Une solution proposée est de construire une collection imbriquée de forêts en ajoutant ou en retirant de façon itérative des variables selon leur valeur d'importance. C'est l'approche retenue notamment par Svetnik et al. (2004), Díaz-Uriarte and Alvarez de Andrés (2006), Genuer et al. (2010) et se réfère dans la suite à la procédure *Non Recursive Feature Elimination* ou NRFE. Une troisième solution consiste à intégrer la mesure d'importance dans un algorithme *greedy* similaire à la procédure SVM-RFE décrite dans la section précédente. Dans le Chapitre 4, nous comparons ces deux algorithmes dans le cas de variables corrélées. Nous discutons également le choix du calcul de l'erreur, OOB ou par validation croisée.

2.6 Mesure d'importance par permutation et indices de sensibilité

Dans cette section, nous montrons que la mesure d'importance par permutation introduite dans la section précédente est liée aux indices de sensibilité utilisés couramment pour l'analyse des modèles numériques de simulation. Nous introduisons tout d'abord des généralités sur l'analyse de sensibilité, sans chercher à être exhaustif. L'article de Iooss

and Lemaître (2014) constitue une introduction méthodologique complète du sujet.

2.6.1 Éléments d'analyse de sensibilité

Afin d'étudier des phénomènes physiques complexes, les ingénieurs ont recours à des modèles de simulations numériques leurs permettant de s'affranchir d'expérimentations réelles impossibles à mettre en œuvre. Un modèle est constitué de systèmes d'équations physiques. Il est évalué grâce à un code de calcul, programme informatique pouvant être coûteux en temps d'exécution. Parmi les nombreuses applications industrielles (de Rocquigny; 2006; de Rocquigny et al.; 2008), nous pouvons citer entre autres l'étude des risques environnementaux (Iooss et al.; 2006; Volkova et al.; 2008) ou la sécurité des réacteurs nucléaires (Auder et al.; 2012).

Plus formellement, le modèle est représenté par une fonction déterministe f qui, pour un vecteur de paramètres d'entrée $\mathbf{X} = (X_1, \ldots, X_j, \ldots, X_p) \in \mathbb{R}^p$, fait correspondre une sortie scalaire $Y \in \mathbb{R}$ par la relation

$$Y = f(\mathbf{X}).$$

Dans ce contexte, l'analyse de sensibilité vise à identifier les paramètres qui influent le plus sur la sortie du modèle (Saltelli et al.; 2009). En particulier, les indices de Sobol (Sobol; 1993) permettent de mesurer la contribution de chaque paramètre d'entrée à la variance de Y. L'estimation de ces indices utilise la décomposition de Hoeffding (Hoeffding; 1948) de f en une somme de fonctions de dimension croissante :

$$f(\mathbf{x}) = f_0 + \sum_{j=1}^{p} f_j(x_j) + \sum_{j<k}^{p} f_{jk}(x_j, x_k) + \cdots + f_{1\ldots p}(\mathbf{x}),$$

où f est de carré intégrable sur $[0,1]^p$ et f_0 est une constante réelle. Sobol (1993) a montré que cette décomposition est unique sous certaines hypothèses supplémentaires sur f.

Si l'on suppose à présent que les paramètres d'entrée sont mutuellement indépendants, alors la variance de Y se décompose de la même manière :

$$\mathrm{Var}(Y) = \sum_{j=1}^{p} \mathrm{Var}(f_j(X_j)) + \sum_{j<k}^{p} \mathrm{Var}(f_{jk}(X_j, X_k)) + \cdots + \mathrm{Var}(f_{1\ldots p}(\mathbf{X})),$$

avec $f_j(X_j) = \mathbb{E}[Y|X_j]$, $f_{jk}(X_j, X_k) = \mathbb{E}[Y|X_j, X_k] - f_j(X_j) - f_k(X_k)$, etc. L'effet individuel de la variable X_j est donné par

$$D_j(Y) := \mathrm{Var}(\mathbb{E}[Y|X_j]),$$

l'effet de l'interaction entre les variables X_j et X_k est

$$D_{jk}(Y) := \mathrm{Var}(\mathbb{E}[Y|X_j, X_k]) - \mathrm{Var}(\mathbb{E}[Y|X_j]) - \mathrm{Var}(\mathbb{E}[Y|X_k]),$$

et ainsi de suite. Les indices de Sobol sont finalement définis par

$$S_j = \frac{D_j(Y)}{\mathrm{Var}(Y)}, \; S_{jk} = \frac{D_{jk}(Y)}{\mathrm{Var}(Y)}, \; \cdots$$

et vérifient la relation

$$\sum_{j=1}^{p} S_j + \sum_{j<k}^{p} S_{jk} + \cdots = 1.$$

Ces critères mesurent la part de variance expliquée par les variables individuelles (indices d'ordre 1) et par les interactions entre les variables (indices d'ordre supérieur).

Homma and Saltelli (1996) ont introduit les *indices totaux* pour mesurer la contribution de l'ensemble de tous les groupes de variables contenant X_j :

$$S_{T_j} := \sum_{\mathcal{J} \subset \{1,\ldots,p\}, j \in \mathcal{J}} S_{\mathcal{J}} = 1 - \frac{\mathrm{Var}\left(\mathbb{E}\left[Y|\mathbf{X}_{-j}\right]\right)}{\mathrm{Var}(Y)},$$

avec $\mathbf{X}_{-j} = (X_1, \ldots, X_{j-1}, X_{j+1}, \ldots, X_p)$. Nous pouvons à présent énoncer deux cas particuliers de ces indices pour une fonction f linéaire et additive :

- Si f est additive, i.e. $f(\mathbf{X}) = \sum_{j=1}^{p} f_j(X_j)$, alors $S_j = S_{T_j}$.
- Si f est linéaire, i.e. $f(\mathbf{X}) = \sum_{j=1}^{p} \beta_j X_j$, alors l'indice de Sobol d'ordre 1 S_j et l'indice total S_{T_j} concordent avec le carré du *coefficients de régression standardisé*

$$\mathrm{SRC}_j^2 = \beta_j^2 \frac{\mathrm{Var}(X_j)}{\mathrm{Var}(Y)}.$$

Il existe un grand nombre de méthodes d'estimation des indices de Sobol, notamment celles développées par Sobol (1993) et Saltelli (2002) basées sur des techniques de Monte Carlo. L'inconvénient majeur de ces méthodes historiques vient du fait qu'elles nécessitent un grand nombre d'appels au code. En effet, à partir de n sorties $Y_1, \ldots, Y_n$, les indices du premier ordre sont estimés à l'aide de $n(p+1)$ évaluations de f. Nous n'entrons pas plus dans les détails et nous renvoyons le lecteur à l'article de Iooss and Lemaître (2014) ainsi qu'au chapitre 1 de la thèse de Chastaing (2013) pour une revue complète des différentes méthodes employées.

Historiquement, les indices de sensibilités ont été développés sous l'hypothèse d'indépendance des variables d'entrée. Un certain nombre de travaux ont étendu ce contexte aux variables dépendantes, par exemple par Chastaing et al. (2012).

2.6.2 Comparaison des indices de Sobol avec la mesure d'importance par permutation

Dans cette section, nous comparons les indices de Sobol avec la mesure d'importance par permutation. Supposons que la distribution de probabilité du vecteur $(\mathbf{X}, Y)$ satisfait le modèle

$$Y = f(\mathbf{X}) + \varepsilon, \tag{2.6.1}$$

où les variables aléatoires $X_1, \ldots, X_p$ sont indépendantes et où $\mathbb{E}[\varepsilon|\mathbf{X}] = 0$ et $\mathbb{E}[\varepsilon^2|\mathbf{X}] = \sigma^2$. Dans ce cas, la définition des indices de Sobol est modifiée pour tenir compte du terme de bruit ε. Remarquons tout d'abord que

$$\mathrm{Var}(Y) = \mathrm{Var}(f(\mathbf{X})) + \sigma^2.$$

Par suite, en décomposant la variance de Y comme précédemment, nous avons

$$\mathrm{Var}(Y) = \sum_{j=1}^{p} \mathrm{Var}(f_j(X_j)) + \sum_{j<k}^{p} \mathrm{Var}(f_{jk}(X_j, X_k)) + \cdots + \mathrm{Var}(f_{1\ldots p}(\mathbf{X})) + \sigma^2,$$

avec $f_j(X_j) = \mathbb{E}[Y|X_j]$, $f_{jk}(X_j, X_k) = \mathbb{E}[Y|X_j, X_k] - f_j(X_j) - f_k(X_k)$, etc. Les indices de Sobol d'ordre 1 ne changent pas dans ce contexte outre le fait que

$$\sum_{j=1}^{p} S_j + \sum_{j<k}^{p} S_{jk} + \cdots = 1 - \frac{\sigma^2}{\mathrm{Var}(Y)}.$$

Le terme $\frac{\sigma^2}{\mathrm{Var}(Y)}$ mesure la part de la variance de Y expliquée par ε. Nous en déduisons alors les indices totaux pour le modèle de régression (2.6.1) :

$$S_{T_j} = 1 - \frac{\mathrm{Var}\left(\mathbb{E}\left[Y|\mathbf{X}_{-j}\right]\right)}{\mathrm{Var}(Y)} - \frac{\sigma^2}{\mathrm{Var}(Y)}.$$

Cette définition n'est pas fondamentalement différente de celle qui est utilisée habituellement en analyse de sensibilité. En effet, nous pouvons voir la variable ε comme un paramètre d'entrée additionnel qui influe sur Y par son niveau de variance σ^2.

Par ailleurs, la mesure d'importance par permutation définie dans la section précédente (Équation (2.5.2)) s'écrit, pour la perte quadratique,

$$\begin{aligned} \mathcal{I}(X_j) &= \mathbb{E}\left[\left(Y - f(\mathbf{X}_{(j)})\right)^2\right] - \mathbb{E}\left[(Y - f(\mathbf{X}))^2\right] \\ &= \mathbb{E}\left[\left(f(\mathbf{X}) - f(\mathbf{X}_{(j)})\right)^2\right], \end{aligned}$$

où le vecteur aléatoire $\mathbf{X}_{(j)} = (X_1, \ldots, X'_j, \ldots, X_p)$ est tel que X'_j une est réplication indépendante de X_j.

La proposition suivante montre que la mesure d'importance par permutation est égale, à un facteur de normalisation près, à l'indice de Sobol total défini plus haut.

Proposition 1. *Supposons que sous le modèle (2.6.1), les variables $X_1, \ldots, X_p$ sont indépendantes. Alors pour tout $j \in \{1, \ldots, p\}$,*

$$S_{T_j} = \frac{\mathcal{I}(X_j)}{2\,\mathrm{Var}(Y)}.$$

Démonstration. Par définition,

$$\mathcal{I}(X_j) = \mathbb{E}\left[\left(f(\mathbf{X}) - f(\mathbf{X}_{(j)})\right)^2\right].$$

En conditionnant par le vecteur $\mathbf{X}_{-j}$ et par indépendance des variables, nous obtenons

$$\begin{aligned} \mathcal{I}(X_j) &= \mathbb{E}\left(\mathbb{E}\left[\left(f(\mathbf{X}) - f(\mathbf{X}_{(j)})\right)^2 |\mathbf{X}_{-j}\right]\right) \\ &= 2\mathbb{E}\left(\mathrm{Var}\left[f(\mathbf{X})|\mathbf{X}_{-j}\right]\right) \end{aligned}$$

Sous le modèle (2.6.1), la variance de Y s'écrit $\mathrm{Var}(Y) = \mathrm{Var}(f(\mathbf{X})) + \sigma^2$. La loi de la

variance totale implique

$$\begin{aligned}\mathcal{I}(X_j) &= 2\left(\mathrm{Var}\left(f(\mathbf{X})\right) - \mathrm{Var}\left(\mathbb{E}\left[f(\mathbf{X})|\mathbf{X}_{-j}\right]\right)\right)\\ &= 2\Big(\mathrm{Var}(Y) - \sigma^2 - \mathrm{Var}\left(\mathbb{E}\left[Y|\mathbf{X}_{-j}\right] - \mathbb{E}\left[\varepsilon|\mathbf{X}_{-j}\right]\right)\Big)\\ &= 2\Big(\mathrm{Var}(Y) - \sigma^2 - \mathrm{Var}\left(\mathbb{E}\left[Y|\mathbf{X}_{-j}\right]\right)\Big)\\ &= 2\,\mathrm{Var}(Y)S_{T_j}.\end{aligned}$$

□

Ce résultat nous donne une méthode alternative d'estimation à moindre coût des indices de Sobol totaux avec les forêts aléatoires. En effet, comme nous l'avons écrit précédemment, les méthodes d'estimation par Monte Carlo nécessitent $n(p+1)$ évaluations du code à partir de n simulations. Dans le cas où le code de calcul est coûteux, le nombre de simulations n est limité et les estimations peuvent être imprécises. Dans le contexte des forêts aléatoires, nous montrons ici que les indices de Sobol totaux peuvent être directement estimés sans utiliser des méthodes d'échantillonnage et surtout en évaluant le code de calcul n fois.

Dans la section suivante, nous revenons à notre objectif initial de l'analyse des données de vol. Nous étudions le problème de l'atterrissage long lorsque les paramètres de vol sont observés à l'altitude de 500 pieds.

2.7 Application au risque d'atterrissage long

Dans cette section, nous étudions le risque d'atterrissage long à partir des données observées lors de la phase de stabilisation à 500 pieds. Rappelons que ce risque est défini par le dépassement des distances d'atterrissage d'un seuil réglementaire de 60 % de la longueur de piste (voir la Section 1.3). Au delà de cette limite, l'atterrissage est considéré comme long.

Le jeu de données `Class1` utilisé contient 10 819 vols d'une même compagnie et de type avion identique. Sur les 250 paramètres de vol enregistrés, une présélection de 89 variables (dont 49 variables binaires) a été effectuée par les experts aéronautiques afin d'en retirer les moins informatifs. La Table 2.1 donne les effectifs de chaque classe et montre que nous sommes en présence d'un problème déséquilibré.

Classe	-1	1	Total
Effectif	9754	1065	10819
Proportion (en %)	90	10	100

Table 2.1 – Effectif de la base de données par classes d'atterrissage. L'atterrissage long est codé 1.

Nous avons choisi, dans un premier temps, d'analyser les paramètres de vol à l'altitude de 500 pieds soit environ une minute avant le toucher de la piste. À cette altitude, l'avion doit être stabilisé pour que l'atterrissage se passe dans de bonnes conditions. Notre objectif est d'identifier un faible nombre de variables permettant de prédire le risque d'atterrissage long avec une bonne précision. Pour cela, nous comparons trois méthodes de classification,

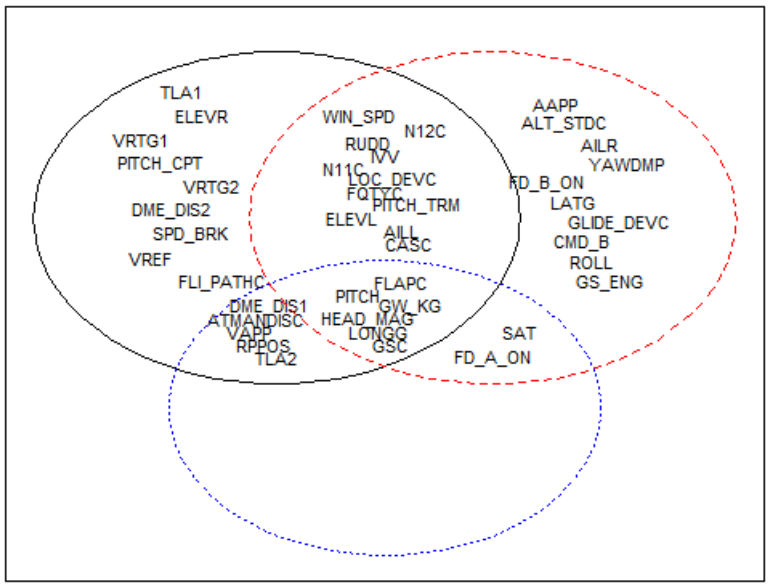


FIGURE 2.8 – Résultats de la sélection de variables de la régression logistique (pointillés bleu), des forêts aléatoires (trait plein) et des machines à vecteurs de support (tirets rouges).

la régression logistique, les SVM et les forêts aléatoires selon deux critères : le nombre de variables sélectionnées et l'erreur de prédiction.

Pour les trois méthodes, nous utilisons un algorithme *backward*. La sélection par les SVM s'effectue par l'algorithme SVM-RFE décrit dans la Section 2.4.2 avec un noyau Gaussien. Nous utilisons également la procédure RFE dans le cadre des forêts aléatoires. La mesure d'importance par permutation est utilisée pour guider la recherche de la variable à éliminer (Section 2.5.3). Pour la régression logistique, le choix des variables s'effectue avec le critère BIC selon l'algorithme décrit dans Hastie et al. (2001, Chap. 3).

Sélection des paramètres de vol

Le diagramme 2.8 représente les paramètres de vol sélectionnés pour chacune des trois méthodes. Nous observons que les trois méthodes donnent des résultats cohérents. En effet, les six variables les plus sélectionnées sont la vitesse sol (GSC), le cap magnétique (HEAD_MAG), l'accélération longitudinale (LONGG), la masse de l'avion (GW_KG), l'assiette (PITCH) et la position des volets (FLAPC).

La présence de la vitesse est assez naturelle puisqu'elle influe directement sur l'énergie accumulée par l'avion. Cela montre aussi que la vitesse du vent est déterminante, la vitesse air étant égale à la vitesse sol à laquelle s'ajoute la vitesse du vent. En effet, nous savons qu'un fort vent arrière peut engendrer un atterrissage long. Le cisaillement de vent peut également perturber la stabilisation de l'avion. Ce phénomène météorologique se définit comme un changement brutal de la vitesse ou de la direction du vent.

Le paramètre HEAD_MAG n'est pas un paramètre de pilotage mais sa présence parmi les variables les plus influentes montre que certains aéroports ont un risque d'atterrissage

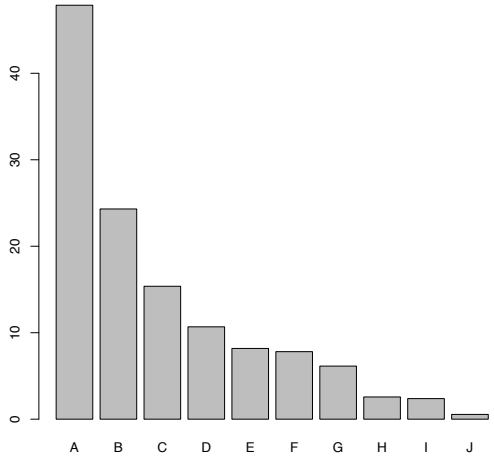


Figure 2.9 – Proportions des atterrissages longs pour les dix aéroports les plus fréquentés.

long plus élevé. En effet, chaque piste a un cap spécifique et une analyse plus approfondie permet de mettre en évidence plusieurs pistes où les atterrissages longs sont nombreux (Figure 2.9). En particulier, sur une des pistes de l'aéroport A, plus de 40 % des atterrissages dépassent la limite réglementaire . Cette information permet alors à la compagnie concernée de mettre en place une surveillance particulière sur cet aéroport pour limiter les risques d'atterrissages longs.

La masse de l'avion montre que plus un avion est lourd, plus il présente un risque d'atterrissage long du fait d'une plus grande difficulté de décélération. La présence de l'accélération longitudinale, de l'assiette et de la position des volets s'expliquent par le fait que ces variables ont une influence sur la stabilisation de l'avion et nous avons expliqué que la non stabilisation de l'avion juste avant l'atterrissage peut conduire à un atterrissage long.

Cette première étape permet de comprendre les causes principales d'un atterrissage long et répond à un premier objectif opérationnel. La section suivante compare les performances des trois méthodes utilisées afin d'établir celle qui a les meilleures propriétés prédictives.

Erreurs de classification

La Table 2.2 donne les taux d'erreur obtenus après sélection des variables pour les trois méthodes testées. Avec 13 variables sélectionnées, la régression logistique propose une sélection plus parcimonieuse. À l'inverse, le taux d'erreur le plus faible est obtenu par les forêts aléatoires (8.9 % d'erreur de classification).

Cependant, cette observation ne permet pas de conclure. En effet, les données sur lesquelles nous travaillons sont déséquilibrées : la proportion entre la classe minoritaire et la classe majoritaire est d'environ 10 %. Les trois algorithmes estiment l'erreur de classification entre 0.09 et 0.1 mais l'erreur sur les données de classe 1 est proche de 1 ce qui est un problème puisque c'est précisément la classe qu'il faut prédire avec grande précision. Il est donc important de rééquilibrer les données. Les SVM et les forêts aléatoires offrent cette possibilité ce qui les rend particulièrement intéressantes.

Méthode	**Nb de variables**	**Taux d'erreur**
Régression Logistique (BIC)	**13**	0.097
Forêts aléatoires (RFE)	31	**0.089**
SVM-RFE	29	0.097

TABLE 2.2 – Nombre de variables sélectionnées et taux d'erreur pour les trois méthodes de classification.

Plus précisément, le rééquilibrage des SVM s'effectue en affectant un poids à chaque classe égal à la proportion de l'autre classe, c'est-à-dire un poids de 0.9 pour la classe minoritaire que l'on souhaite favoriser et 0.1 pour la classe majoritaire. Pour les forêts aléatoires, nous appliquons un sous-échantillonnage de la classe majoritaire de sorte que chaque arbre de décision soit construit à partir d'un ensemble équilibré. Dans les deux cas, cette manipulation a pour effet d'améliorer les performances prédictives de la classe minoritaire au dépens de la classe majoritaire, ce qui implique un taux d'erreur global plus important. La Table 2.3 illustre cette observation dans les forêts aléatoires : l'erreur de classification des observations de classe -1 est de 0.16 alors qu'elle était nulle pour le modèle déséquilibré et le taux d'erreur de la classe 1 est de 0.40 au lieu de 1 auparavant.

Obs. / Pred.	-1	1	Taux d'erreur
-1	8174	1589	0.16
1	428	637	0.40

TABLE 2.3 – Forêts aléatoires – Erreurs OOB de chaque classe après rééquilibrage des données pour l'ensemble des variables. L'erreur OOB est de 0.19.

Cependant, évaluer les performances prédictives d'un modèle par l'erreur de classification (ou OOB dans le cas des forêts aléatoires) devient discutable en cas de déséquilibre. D'autres indicateurs existent pour mesurer la qualité de la prévision dans le cas de données déséquilibrées en prenant en compte l'information contenue dans chaque classe. La Table 2.4 introduit des notations utiles : TN est le nombre de vrais négatifs, TP est le nombre de vrais positifs, FN est le nombre de faux négatifs et FP le nombre de faux positifs. TN et TP représentent les observations bien classées par la méthode tandis que FN et FP représentent les mal classés. Définissons les quantités suivantes :

$$\text{Se} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad \text{(Sensibilité)} \tag{2.7.1}$$

$$\text{Sp} = \frac{\text{TN}}{\text{TN} + \text{FP}}. \quad \text{(Spécificité)} \tag{2.7.2}$$

Ces indicateurs sont utilisés dans la construction de la courbe ROC (pour Receiver Operating Characteristic), représentation graphique évaluant la capacité prédictive d'une méthode de classification en tenant compte à la fois la prédiction des données majoritaires et minoritaires. Elle représente Se en fonction de (1 - Sp) lorsque l'on fait varier le seuil de décision, la valeur limite à partir de laquelle la classification prédit positif. Plus précisément, la règle de Bayes est approchée en estimant les probabilités a posteriori $\eta(\mathbf{x}) = \mathbb{P}[Y = 1|\mathbf{X} = \mathbf{x}]$. La classification prédit 1 si $\hat{\eta}(\mathbf{x}) > 0.5$ et -1 sinon. La valeur

Obs. / Pred.	Négatif	Positif
Négatif	TN	FN
Positif	FP	TP

Table 2.4 – Matrice de confusion.

0.5 est le seuil limite à partir duquel la méthode de classification prédit le label positif. Construire une courbe ROC revient à calculer la sensibilité et la spécificité en faisant varier ce seuil de décision de 0 à 1. La classification parfaite est celle pour laquelle sensibilité et spécificité valent 1 simultanément, autrement dit, une classification correcte des observations dans leur classe respective. À l'inverse, lorsque Se = 1 - Sp, le modèle prédit 1 ou -1 avec probabilité $\frac{1}{2}$. Cette situation se traduit, dans la représentation graphique, par la droite d'équation $y = x$.

Pour chaque seuil, la spécificité ainsi que la sensibilité sont estimées sur un échantillon test contenant 30 % des observations choisies aléatoirement. La Figure 2.10 représente les courbes ROC des trois techniques utilisées avec les paramètres de vol issus de leur sélection de variables respective (Table 2.2). Les forêts aléatoires semblent meilleures que les SVM et la régression logistique. La Figure 2.11 compare les modèles construits avec l'intersection des variables sélectionnées dans chaque cas c'est-à-dire les six paramètres suivant : GSC, HEAD_MAG, LONGG, GW_KG, FLAPC et PITCH. Nous observons, de même, que les forêts aléatoires donnent de meilleurs résultats. Les SVM semblent moins bons dans ce deuxième cas. Cela montre que les six variables utilisées ne contiennent pas à elles seules assez d'information pour prédire le risque d'atterrissage long. Enfin, la Figure 2.12 montre l'intérêt de la pondération de chaque classe dans les SVM en terme de performances prédictives. Les SVM prédisent mieux après un rééquilibrage des données. Concernant les forêts aléatoires, le sous-échantillonnage semble ne pas avoir d'effet significatif. Les paramètres de vols utilisés sont issus des sélections de variables (31 variables pour les forêts aléatoires et 29 pour les SVM).

Même si les forêts aléatoires sélectionnent un grand nombre de paramètres de vol, elles présentent ici les meilleures performances pour la prédiction des atterrissages longs.

Cartographie du risque

À l'issue de l'étape de sélection, nous utilisons les probabilités *a posteriori* comme un indicateur de risque. Nous mesurons ainsi la probabilité que chaque vol soit dans la classe "à risque". Nous proposons ici une cartographie du risque comme une représentation graphique des probabilités *a posteriori* estimées. L'objectif est d'avoir une vision globale du risque d'atterrissage long pour les paramètres sélectionnés. La construction se fait de la façon suivante :

1. Sélection de variables ;
2. Apprentissage d'un modèle optimal ;
3. Estimation du support de la distribution des données à 500 pieds ;
4. Calcul des probabilités *a posteriori* en chaque points d'une grille appartenant au support.

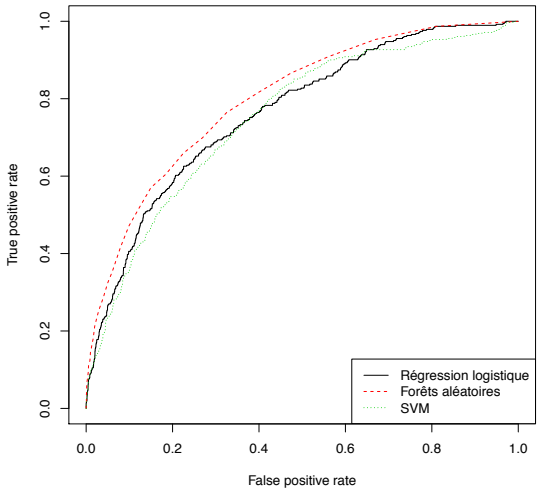


FIGURE 2.10 – Courbes ROC pour la classification des atterrissages longs. Chaque modèle est construit avec sa sélection de variables.

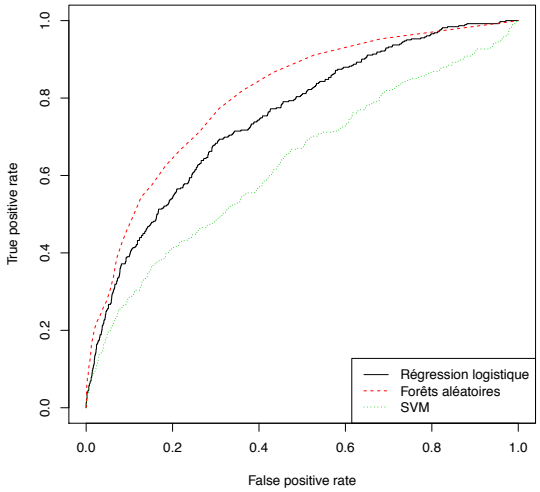


FIGURE 2.11 – Courbes ROC pour la classification des atterrissages longs. Les variables utilisées sont GSC, HEAD_MAG, LONGG, GW_KG, FLAPC et PITCH.

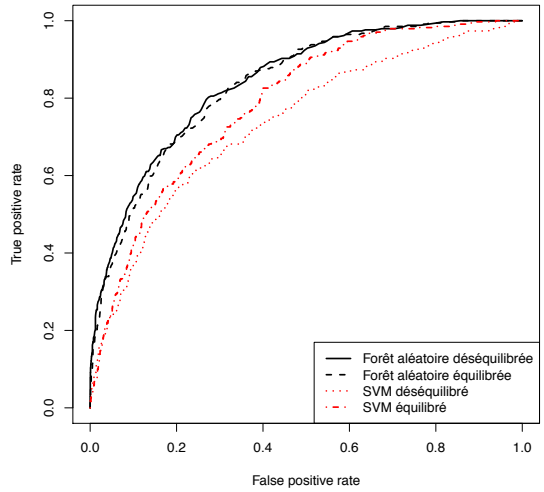


Figure 2.12 – Comparaison des forêts aléatoires et des SVM pour des données rééquilibrées.

Estimation du support

Afin de proposer une cartographie représentative des données, nous limitons les zones de cartographie aux voisinages des observations. Nous devons donc estimer au préalable le support des données à 500 pieds.

Une approche naturelle pour l'estimation du support est d'utiliser l'estimateur à noyau de la densité dans le cas multivarié. Il est donné par

$$\hat{p}_{\mathbf{X}}(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} k_H(\mathbf{x} - \mathbf{X}_i),$$

où

$$k_H(\mathbf{x}) = |H|^{-1/2} k(|H|^{-1/2}\mathbf{x}),$$

avec H une matrice de lissage $p \times p$ symétrique et définie positive et k est une fonction noyau. Un estimateur du support est alors donné par l'estimateur plug-in

$$\hat{\mathcal{S}} = \{\mathbf{x} \in \mathbb{R}^p, \hat{p}_{\mathbf{X}}(\mathbf{x}) > s\},$$

où s est un seuil fixé.

Néanmoins, cette approche présente deux inconvénients : le fléau de la dimension et le temps de calculs. Le fléau de la dimension désigne la difficulté de faire de l'inférence en grande dimension. Une solution possible à ce problème consiste à définir un modèle probabiliste. Nous pouvons émettre, par exemple, des hypothèses paramétriques sur les densités conditionnelles $p_{\mathbf{X}|Y=1}$ et $p_{\mathbf{X}|Y=0}$. L'estimation de la densité revient alors à estimer les différents paramètres des lois *a priori*, qui sont généralement simple à calculer (lois gaussiennes par exemple). Dans notre cas, cette approche n'est pas réaliste car nous ne disposons pas d'informations *a priori* sur les distributions conditionnelles. Le second inconvénient est d'ordre computationnel. En effet, pour évaluer une représentation suffisamment fine des probabilités *a posteriori*, nous devons estimer la densité sur chaque point

d'une grille suffisamment fine (par exemple 100 points par dimension). De plus, la matrice de lissage H est difficilement calculable en grande dimension. Afin de palier ces inconvénients, une méthode naïve et facilement implémentable de l'estimation du support est proposée. Nous l'estimons en dimension 2 en construisant une grille pour chaque dimension puis en trouvant les points qui sont proches d'une donnée d'apprentissage. Autrement dit, nous décidons que les points de la grille se situant au voisinage d'une observation appartiennent au support de la loi.

Construction de la cartographie

La cartographie du risque représente les probabilités *a posteriori* calculées en chaque point d'une grille construite pour chaque variable explicative issue de l'étape de sélection. Il est clair que plus la dimension augmente, plus le nombre de points à évaluer est important. Comme pour l'estimation du support, la procédure devient très rapidement limitée si l'on veut une représentation fine des probabilités *a posteriori*. À titre d'exemple, si 20 variables sont sélectionnées et si l'on veut une grille de taille 100 dans chaque dimension, alors le nombre de probabilités à calculer est égal à 100^{20}. C'est évidemment irréalisable en pratique car les calculs doivent être faits en temps raisonnable.

Deux stratégies sont possibles pour contourner le problème de la dimension. La première consiste à estimer les probabilités *a posteriori* pour les deux variables que l'on souhaite représenter, en fixant les autres dimensions à leur valeur médiane. Il s'agit, en pratique, d'appliquer la méthode de classification avec toutes les variables sélectionnées puis de les cartographier deux à deux. Cela permet d'avoir un nombre raisonnable de points à évaluer ($100^2 = 10\ 000$ points) pour chaque représentation (les autres variables étant constantes). Néanmoins, l'utilisation de la médiane induit une perte d'information et donc une incertitude dans l'estimation des probabilités *a posteriori*. La seconde approche consiste à estimer le support et à construire des cartographies à partir de modèles à deux variables. Cependant, les modèles ne contenant que deux variables ne sont clairement pas optimaux en terme d'erreur de classification comme l'illustre la Table 2.5.

Variables	**Erreur de classification**	**1 - Sp**	**1 - Se**
HEAD_MAG, GSC	0.35	0.36	0.31
HEAD_MAG, N11C	0.32	0.31	0.45
GSC, N11C	0.41	0.42	0.29

TABLE 2.5 – Taux d'erreur (en %) pour des modèles SVM à deux variables.

La Figure 2.13 montre un exemple de cartographie construite à partir de SVM à deux variables, la vitesse sol (GSC) et le cap magnétique (HEAD_MAG). La ligne de niveau correspond à un seuil de 0.1 défini *a priori* : on considère que pour une probabilité supérieure à 10 %, l'atterrissage est trop incertain. Les vols qui ont une probabilité supérieure à ce seuil ont un risque d'atterrissage long significatif.

2.8 Conclusion du chapitre

Dans ce chapitre, nous avons présenté un aperçu des méthodes d'apprentissage statistique supervisé et de sélection de variables. Nous avons introduit la régression logistique, les SVM et les forêts aléatoires. Ces trois méthodes ont été comparées pour l'analyse des données de vol à 500 pieds pour la prédiction du risque d'atterrissage long. Nous donnons

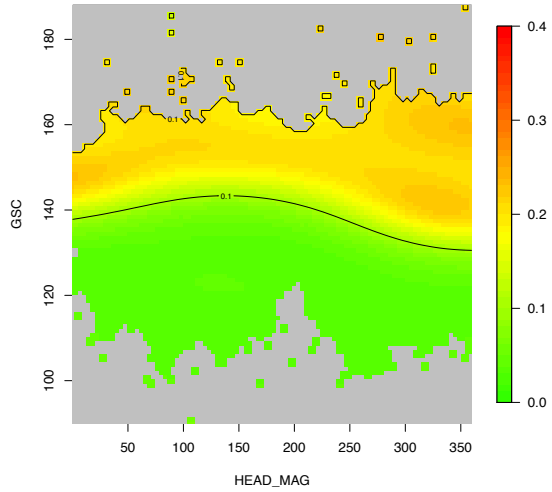


Figure 2.13 – Cartographie du risque d'atterrissage long pour la vitesse sol (GSC) et le cap magnétique (HEAD_MAG).

ainsi une première réponse satisfaisante à la problématique de l'analyse des enregistreurs de vol.

Il a été observé que le nombre de variables sélectionnées varie selon la méthode employée. Avec 13 variables sélectionnées, la régression Logistique sélectionne moins de variables que les SVM et les forêts aléatoires qui retiennent respectivement 29 et 31 paramètres de vol. Les forêts aléatoires présentent cependant de meilleures performances en prédiction. Nous avons également mesuré l'effet du déséquilibre des classes d'atterrissage et la difficulté à discriminer les vols dans ce cas. Afin de simplifier le problème dans la suite, nous considérerons systématiquement des données équilibrées quitte à effectuer un sous-échantillonnage préalable. Nous avons identifié six paramètres de vol influents pour la prédiction des classes d'atterrissage : la vitesse sol, le cap magnétique, l'accélération longitudinale, la masse de l'avion, l'assiette et la position des volets. Nous avons ainsi montré que la vitesse du vent et la non stabilisation de l'avion à 500 pieds sont des facteurs de risque. La présence du cap parmi les variables sélectionnées montre un risque plus élevé sur certains aéroports. Ces résultats suggèrent de filtrer les données selon l'aéroport de destination.

Nous avons par ailleurs proposé une cartographie représentant le risque d'atterrissage long à partir des probabilités *a posteriori* estimées. Une telle représentation des risques fournit un premier outil d'aide à la décision proposé aux opérationnels.

Notons enfin que se restreindre à l'observation des vols à l'altitude de 500 pieds est sous-optimal d'un point de vue statistique et opérationnel. Il est important de pouvoir utiliser l'ensemble de l'information contenue dans les données pour améliorer la prédiction et la compréhension des atterrissages longs. Le chapitre suivant présente les outils nécessaires à l'analyse de données échantillonnées dans le temps.

Chapitre 3

Apprentissage de données fonctionnelles

Résumé. Les enregistreurs de vol fournissent un très grand volume de données. En effet, les paramètres de vol sont enregistrés au minimum une fois par seconde. Il est donc nécessaire d'avoir des outils d'analyse adaptés à ce contexte. Une solution consiste à traiter les signaux comme des fonctions. On parle alors de données fonctionnelles. Dans ce chapitre, essentiellement bibliographique, nous présentons les outils dont nous avons besoin pour traiter les données de vol. La section 3.3.3 est dédiée à une adaptation du débruitage par ondelettes simultanément pour n processus indépendants, méthode qui nous servira à réduire la dimension des données.

Sommaire

3.1 Introduction

L'analyse des données fonctionnelles désigne la modélisation et le traitement statistique de variables aléatoires à valeurs dans un espace de fonctions. Les premiers travaux portent sur la décomposition de Karhunen-Loève d'un processus stochastique à valeurs dans un espace de Hilbert (voir notamment Deville; 1974, Dauxois and Pousse; 1976 et Dauxois et al.; 1982). L'analyse des données fonctionnelles s'est beaucoup développée ces dernières années, notamment suite aux ouvrages de Ramsay and Silverman (1997, 2002, 2005). Les

applications sont diverses. Citons par exemple la prévision de la consommation électrique (Misiti et al.; 1994; Antoniadis et al.; 2014) ou l'analyse de données biomédicales (Song et al.; 2008; Ieva et al.; 2012; Amini et al.; 2013). Des développements théoriques et appliqués ont fait l'objet d'une série d'articles récemment publiée dans Ferraty (2011).

L'extension de l'apprentissage statistique aux données fonctionnelles est la suivante. Considérons un vecteur aléatoire $(\mathbf{X}, Y)$ à valeurs dans $\mathcal{X}^p \times \mathcal{Y}$. Les covariables $X_1, \ldots, X_p$ sont supposées appartenir à un espace de Hilbert $(\mathcal{X}, \langle ., . \rangle_{\mathcal{X}})$. Nous considérons ici des problèmes de régression ($\mathcal{Y} = \mathbb{R}$) et de classification binaire ($\mathcal{Y} = \{-1, 1\}$). Ramsay and Silverman (2005) étudient des modèles de régression linéaire à sortie fonctionnelle, c'est-à-dire lorsque $\mathcal{Y}$ est un espace de fonctions. Ce cas n'est pas considéré dans nos travaux. Soit $\mathcal{D}_n = \{(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)\}$ un échantillon de n observations i.i.d. de même loi que $(\mathbf{X}, Y)$ avec la convention $\mathbf{X}_i = (X_{i1}, \ldots, X_{ip})$. Si $p \geq 2$, on parle d'analyse de *données fonctionnelles multivariées*.

Comme au chapitre précédent, les quantités d'intérêt sont la fonction de régression $f^\star(\mathbf{x}) = \mathbb{E}[Y|\mathbf{X} = \mathbf{x}]$ et les probabilités *a posteriori* $\eta(\mathbf{x}) = \mathbb{P}[Y = 1|\mathbf{X} = \mathbf{x}]$ et $1 - \eta(\mathbf{x})$. Plusieurs démarches existent dans la littérature. Une approche, purement non paramétrique, vise à estimer les fonctions $f^\star$ et η par des estimateurs à noyau. Ces méthodes sont décrites notamment dans Ferraty and Vieu (2006). Il est également possible de définir une métrique adaptée aux données et de construire un estimateur à partir de cette mesure de similarité (Shimodaira et al.; 2001; Bahlmann et al.; 2002; López-Pintado and Romo; 2006; Cuevas et al.; 2007; Alonso et al.; 2012). Enfin, l'approche que nous retenons vise à projeter les covariables fonctionnelles sur un sous-espace de $\mathcal{X}$ de dimension finie. Cette représentation fonctionnelle nous permet de nous ramener à un contexte "classique" d'apprentissage statistique où les variables explicatives sont les coefficients de base associés.

3.2 Représentation fonctionnelle des données

Supposons dans la suite que $\mathcal{X}$ est l'espace $L^2([0, 1])$ muni du produit scalaire

$$\langle f, g \rangle_{L^2} = \int_0^1 f(t)g(t)dx,$$

pour $f, g \in L^2([0, 1])$. Cet espace étant un espace de Hilbert séparable, il existe une base $\mathcal{B} = \{\varphi_1, \varphi_2, \ldots\}$ de fonctions orthogonales telle que

$$X_u(t) = \sum_{k=1}^{\infty} \langle X_u, \varphi_k \rangle_{L^2} \varphi_k(t).$$

La principale difficulté vient du fait que X_u est à valeurs dans un espace de dimension infinie. L'objectif est de réduire la dimension du problème en tronquant la longueur du développement :

$$X_u(t) = \sum_{k=1}^{d_u} \langle X_u, \varphi_k \rangle_{L^2} \varphi_k(t) + \delta_u(t),$$

où la fonction $\delta_u(t)$ est l'erreur due à la projection de X_u sur le sous-espace de $L^2([0, 1])$ de dimension d_u. Idéalement, nous souhaitons obtenir une représentation compacte, c'est-à-dire à partir d'un faible nombre de coefficients de base. En pratique, les covariables fonctionnelles $X_1, \ldots, X_p$ sont observées sur une grille discrète de points $(t_1, \ldots, t_N)$. Les

produits scalaires $\langle X_u, \varphi_k \rangle_{L^2}$ sont alors approximés par les coefficients empiriques

$$Z_{uk} = \frac{1}{N} \sum_{\ell=1}^{N} X_u(t_\ell)\varphi_k(t_\ell).$$

Ainsi, la décomposition de X_u sur $\mathcal{B}$ devient

$$X_u(t_\ell) = \sum_{k=1}^{d_u} Z_{uk}\varphi_k(t_\ell) + \delta_u(t_\ell), \tag{3.2.1}$$

avec $d_u \leq N$. De plus, tout espace de Hilbert séparable étant isomorphe à l'espace $\ell^2 = \{(x_1, x_2, \ldots) : \sum_{k=1}^{\infty} x_k^2 < \infty\}$, l'information portée par les coefficients de base $Z_{u1}, Z_{u2}, \ldots$ est la même que celle contenue dans la fonction X_u.

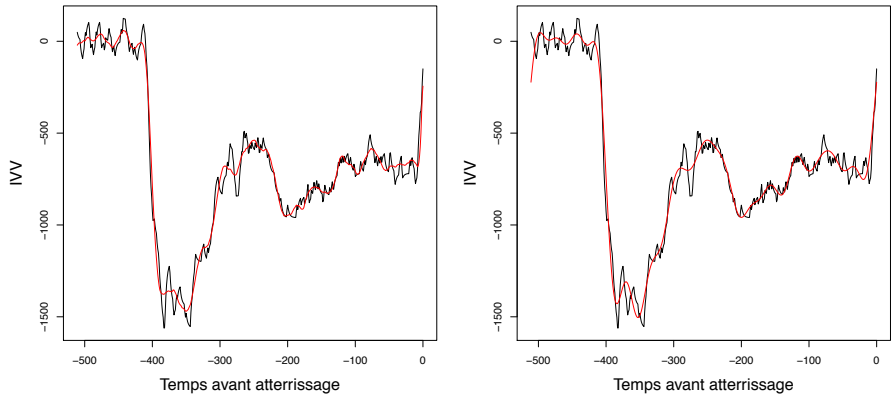
Remarque. *Dans l'équation (3.2.1), nous supposons que les variables $Z_{u1}, \ldots, Z_{ud_u}$ correspondent aux coefficients les plus informatifs pour approximer X_u. Ce n'est pas toujours vrai, en particulier pour les bases d'ondelettes (Section 3.3). Il suffit dans ce cas d'ordonner les coefficients de base pour que la décomposition (3.2.1) reste vraie.*

Dans notre cas, nous n'observons pas un unique processus stochastique $\{X_u(t_\ell), \ell \in \{1, \ldots, N\}\}$. Pour un paramètre de vol (u fixé), nous observons un échantillon de n processus indépendants $X_{u1}, \ldots, X_{un}$ correspondant à chaque vol. Dans un objectif de réduction de dimension, nous recherchons une représentation fonctionnelle commune aux n courbes. Les observations X_{iu} sont donc projetées sur une même base de fonctions $\{\varphi_1, \ldots, \varphi_{d_u}\}$, si possible de faible dimension. Cette étape est essentiellement motivée par une contrainte d'ordre computationnelle. En effet, pour analyser les données de vol en un temps raisonnable, il est important de les représenter de façon la plus parcimonieuse que possible. Par exemple, lorsque les signaux sont échantillonnés sur une grille de temps de taille $N = 512$, résumer l'information à quelques dizaines de coefficients de base est crucial. La Figure 3.1 représente un signal de vitesse verticale (IVV) approximée à partir des 32 premiers coefficients de base. Plusieurs bases de fonctions usuelles sont comparées, la base de Karhunen-Loève (Section 3.4), la base de Fourier, les B-Splines (voir le Chapitre 3 de la thèse de Andrieu; 2013) et trois bases d'ondelettes (Section 3.3).

Une démarche classique pour analyser des données fonctionnelles multivariées peut se résumer ainsi :

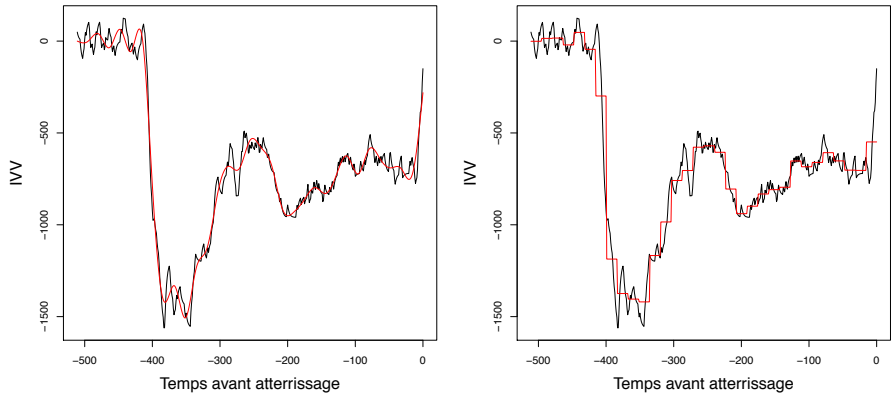
1. Projeter les variables fonctionnelles sur $\mathcal{B}$;
2. Pour chaque covariable X_u, trouver une représentation fonctionnelle commune aux n observations en les projetant sur un sous-espace de dimension d_u ;
3. Utiliser l'ensemble des coefficients comme nouvelles variables explicatives dans un algorithme d'apprentissage.

Dans la section suivante, nous détaillons différentes approches de réduction de dimension avec les ondelettes. Pour analyser les données de vol, nous souhaitons tirer parti de la localisation temporelle et fréquentielle des ondelettes afin de résumer l'information de chaque courbe selon ces deux échelles.



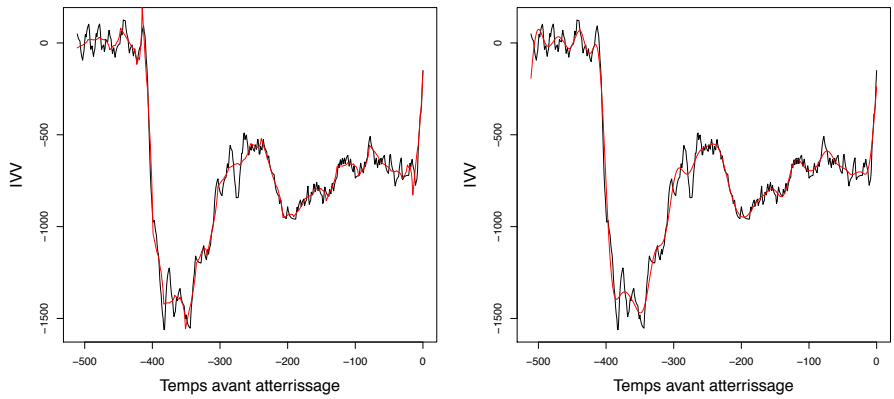
(a) Karhunen-Loève

(b) Fourier



(c) B-Splines

(d) Haar



(e) Ondelette d4

(f) Ondelette d20

FIGURE 3.1 – Reconstruction d'une courbe de vitesse verticale après réduction à 32 premiers coefficients.

3.3 Méthodes de réduction de dimension avec les ondelettes

Dans cette section, nous détaillons des méthodes de réduction de dimension avec les ondelettes. Nous introduisons en particulier le débruitage par ondelettes (Donoho and Johnstone; 1994). Nous développons ensuite une nouvelle approche de seuillage pour la réduction de dimension de n processus indépendants.

3.3.1 Bases d'ondelettes

Les ondelettes, initialement utilisées en traitement du signal, ont suscité de l'intérêt en estimation non paramétrique. À l'inverse de la base de Fourier, les ondelettes ne sont pas uniquement localisées en fréquence mais aussi en temps. Nous donnons ici les concepts généraux de la théorie des ondelettes. Pour plus de détails, le lecteur est renvoyé aux ouvrages de référence : Daubechies (1992), Percival and Walden (2000) et de Mallat (2000, 2008).

Pour construire une base orthogonale de $L^2([0,1])$, définissons une collection de sous-ensembles, pour un certain $j_0 \geq 0$,

$$V_{j_0} \subset V_{j_0+1} \subset \ldots \subset L^2([0,1]),$$

dont l'union est dense dans $L^2([0,1])$ et tels que chaque V_j est engendré par 2^j fonctions d'échelle orthogonales $\phi_{jk}(x) = 2^{j/2}\phi(2^j x - k), k = 0, \ldots, 2^j - 1$. Ces fonctions sont obtenues par translation et dilatation d'une fonction ϕ appelée ondelette père. Ces ensembles forment ce que l'on appelle *analyse multirésolution* (Mallat; 1989).

Pour tout niveau de résolution $j \geq j_0$, l'ensemble W_j est défini comme le complémentaire orthogonal de V_j dans V_{j+1}, c'est-à-dire

$$V_{j+1} = V_j \oplus W_j.$$

Cet ensemble est engendré par 2^j fonctions d'ondelettes orthogonales $\psi_{jk} = 2^{j/2}\psi(2^j - k)$ pour $k = 0, \ldots, 2^j - 1$. La fonction ψ est appelée ondelette mère. Finalement, on a pour un certain $j_0 \geq 0$,

$$L^2([0,1]) = V_{j_0} \oplus W_{j_0} \oplus W_{j_0+1} \oplus \ldots,$$

ce qui implique que la famille

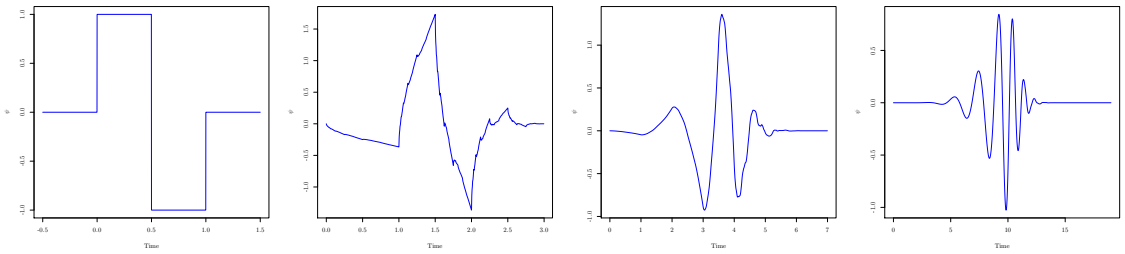
$$\mathcal{B} = \{\phi_{j_0 k}, k = 0, \ldots, 2^{j_0} - 1\} \cup \{\psi_{jk}, j \geq j_0, k = 0, \ldots, 2^j - 1\}$$

forme une base orthogonale de $L^2([0,1])$. En conséquence, la transformée en ondelettes d'une fonction s de $L^2([0,1])$ est donnée par

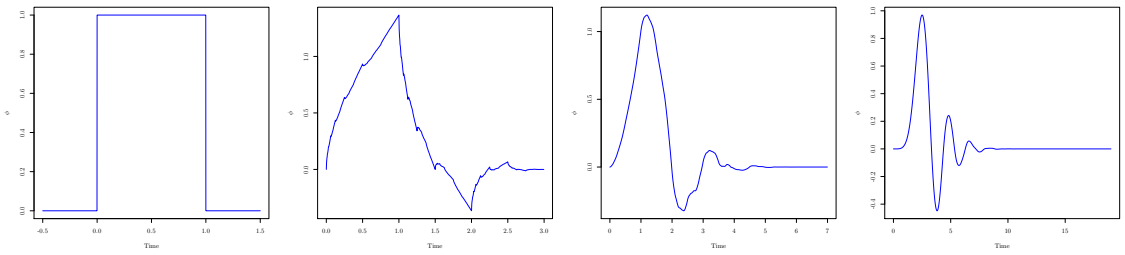
$$s(t) = \sum_{k=0}^{2^{j_0}-1} \langle s, \phi_{j_0 k}\rangle_{L^2} \phi_{j_0 k}(t) + \sum_{j=j_0}^{\infty} \sum_{k=0}^{2^j-1} \langle s, \psi_{jk}\rangle_{L^2} \psi_{jk}(t). \tag{3.3.1}$$

Le premier terme de l'équation (3.3.1) est l'approximation moyenne de s au niveau j_0 et le second terme représente le détail de la décomposition en ondelettes. Dans la suite, nous choisissons $j_0 = 0$ pour avoir un coefficient d'approximation "grossière" et un ensemble de coefficients de détails (voir par exemple Antoniadis et al.; 2013).

Un cas particulier des bases d'ondelettes est le système de Haar. C'est la seule base



(a) Haar (b) d4 (c) d8 (d) d20



(e) Haar (f) d4 (g) d8 (h) d20

FIGURE 3.2 – Ondelettes mère (haut) et père (bas) pour différentes ondelettes de Daubechies.

d'ondelettes qui admet une formulation analytique :

$$\phi(t) = \begin{cases} 1 & \text{si } 0 \leqslant t < 1 \\ 0 & \text{sinon} \end{cases}$$

$$\psi(t) = \begin{cases} 1 & \text{si } 0 \leqslant t < \frac{1}{2} \\ -1 & \text{si } \frac{1}{2} \leqslant t < 1 \\ 0 & \text{sinon.} \end{cases}$$

L'ondelette de Haar fait partie de la famille des ondelettes de Daubechies (Daubechies; 1992). Cette famille est composée de fonctions ϕ et ψ de plus en plus régulières et de moins en moins localisées dans le temps. La figure 3.2 compare différents types d'ondelettes de Daubechies en augmentant leur régularité. On remarque en particulier que plus les fonctions sont régulières, plus leur support est grand.

En pratique, la longueur du développement en ondelettes est conditionnée par le nombre de points de discrétisation N. En supposant que $N = 2^J$ avec $J \geq 1$, la transformée en ondelettes d'un processus $\{X(t_\ell), \ell = 1, \ldots, N\}$ est

$$X(t_\ell) = \zeta\phi_{00}(t_\ell) + \sum_{j=0}^{J-1} \sum_{k=0}^{2^j-1} \xi_{jk}\psi_{jk}(t_\ell),$$

où ζ et ξ_{jk} sont respectivement le coefficient d'échelle et les coefficients d'ondelettes empiriques du signal X au niveau j et à la position k. L'entier J représente le nombre maximal de niveaux que l'on peut considérer. Le calcul des coefficients empiriques se fait par la transformée en ondelettes discrète que décrit par exemple Percival and Walden (2000).

3.3.2 Débruitage par ondelettes

De nombreux auteurs se sont intéressés à l'estimation non paramétrique avec les ondelettes. En particulier, Donoho and Johnstone (1994, 1995, 1998) and Donoho et al. (1995) constituent les articles fondateurs du domaine. En pratique, les méthodes décrites sont notamment employées pour débruiter des signaux ou des images.

Soit un processus stochastique $\{X(t_\ell), \ell \in \{1, \dots, N\}\}$ satisfaisant le modèle

$$X(t_\ell) = s(t_\ell) + \sigma\varepsilon_\ell, \tag{3.3.2}$$

où s est une fonction de $L^2([0,1])$ inconnue, $\varepsilon_1, \dots, \varepsilon_N$ sont des variables aléatoires i.i.d. $\mathcal{N}(0,1)$ et $\sigma > 0$.

Si $N = 2^J$ et $J \geq 1$, la transformée en ondelettes de s est donnée par

$$s(t_\ell) = \omega_0\phi(t_\ell) + \sum_{j=0}^{J-1}\sum_{k=0}^{2^j-1} \omega_{jk}\psi_{jk}(t_\ell),$$

avec $\omega_0 = \langle s, \phi\rangle_{L^2}$ et $\omega_{jk} = \langle s, \psi_{jk}\rangle_{L^2}$ les coefficients d'échelle et d'ondelettes. Le modèle (3.3.2) se réécrit, pour tout $j \in \{0, \dots, J-1\}$ et $k \in \{0, \dots, 2^j - 1\}$,

$$\xi_{jk} = \omega_{jk} + \sigma\eta_{jk}, \tag{3.3.3}$$

et

$$\zeta = \omega_0 + \sigma\eta_0,$$

où ξ_{jk} et ζ sont les coefficients empiriques calculés par la transformée en ondelettes discrète et η_{jk} et η_0 sont des variables aléatoires i.i.d. $\mathcal{N}(0,1)$. Un estimateur de s est de la forme

$$\hat{s}(t_\ell) = \hat{\omega}_0\phi(t_\ell) + \sum_{j=0}^{J-1}\sum_{k=0}^{2^j-1} \hat{\omega}_{jk}\psi_{jk}(t_\ell),$$

où $\hat{\omega}_0 = \zeta$. Les estimateurs $\hat{\omega}_{jk}$ sont obtenus par une règle de seuillage, par exemple le seuillage "dur"

$$\hat{\omega}_{jk}^{ht} = \begin{cases} \xi_{jk} & \text{if } |\xi_{jk}| > \delta_N \\ 0 & \text{sinon,} \end{cases}$$

ou le seuillage "doux"

$$\hat{\omega}_{jk}^{st} = \text{signe}(\xi_{jk})(|\xi_{jk}| - \delta_N)_+. \tag{3.3.4}$$

Ces différents estimateurs permettent d'éliminer les coefficients d'ondelettes pour lesquels ω_{jk} est proche de zéro et d'obtenir *in fine* un estimateur $\hat{s}$ débruité.

Cette approche de seuillage par ondelettes, introduite dans un contexte de régression non paramétrique, a été étendue au seuillage de groupes de coefficients (Hall et al.; 1999; Cai and Zhou; 2009) ainsi qu'à l'estimation de la densité (Donoho et al.; 1996). Il est intéressant de noter que les estimateurs $\hat{s}^{ht}$ et $\hat{s}^{st}$ correspondent respectivement à la sélection ℓ_0 et au Lasso comme le soulignent Massart (2003); Tibshirani (1996); Chen et al. (1998).

Plusieurs seuils sont employés dans la littérature, le plus connu étant le seuil universel $\delta_N = \sigma\sqrt{2\log(N)}$ introduit par Donoho and Johnstone (1994). Donoho and Johnstone (1998) proposent un seuil minimax basé sur la minimisation d'une borne théorique du risque asymptotique et Donoho and Johnstone (1995) proposent une procédure adaptative qui sélectionne un seuil optimal au sens de l'estimateur sans biais du risque (Stein's Unbiased Risk Estimate, SURE).

Il est connu que le seuil universel $\delta_N = \sigma\sqrt{2\log(N)}$ est conservatif. Si l'on suppose que tous les coefficients ω_{jk} sont nuls, le théorème suivant tiré du chapitre 9 du livre de Wasserman (2006) montre que $\hat{\omega}_{jk} = 0$ avec grande probabilité. Il est donné pour le seuillage doux mais est aussi valable pour le seuillage dur.

Théorème 3. *Supposons que $\omega_{jk} = 0$ pour tout j, k et que σ est connu. Soit $\hat{\omega}_{jk}$ l'estimateur du seuillage doux donné par l'équation (3.3.4) avec le seuil universel $\delta_N = \sigma\sqrt{2\log(N)}$. Alors,*

$$\mathbb{P}\left[\hat{\omega}_{jk} = 0, \forall j, k\right] \longrightarrow 1.$$

L'estimation de σ est nécessaire en pratique. L'heuristique proposée par Donoho and Johnstone (1994) est d'utiliser l'estimateur MAD (pour Median Absolute Deviation) sur les coefficients d'ondelettes du niveau le plus fin, c'est-à-dire

$$\hat{\sigma} = \frac{\text{Med}(|\xi_{jk} - \text{Med}(\xi_{jk})| : j = J-1, k = 0, \ldots, 2^{J-1} - 1)}{0.6745},$$

où Med dénote la médiane empirique. L'idée sous-jacente de ce choix est que la variance des coefficients d'ondelettes est essentiellement concentrée dans le niveau le plus fin. Le facteur de normalisation 0.6745 vient de l'hypothèse de normalité faite dans le modèle (3.3.3). Citons, pour être complet, l'article de Johnstone and Silverman (1997) qui étend le seuillage universel lorsque la variance des coefficients dépend du niveau d'approximation j.

Dans notre cas, nous observons un échantillon de n processus indépendants et nous cherchons un développement commun à ces n signaux afin d'en réduire la dimension. La méthode de seuillage présentée ici n'est plus valable. En effet, les coefficients retenus en seuillant indépendamment les n courbes n'ont pas de raison d'être rigoureusement les mêmes pour chaque courbe. C'est pourquoi nous adaptons la règle du seuillage dur pour n processus stochastiques indépendants.

3.3.3 Seuillage consistant de n processus indépendants

Dans cette section, nous présentons une extension naturelle de la règle du seuillage dur pour la réduction de dimension simultanée de n processus aléatoires indépendants $X_1, \ldots, X_n$. Dans un premier temps, nous présentons le cas où les processus sont identiquement distribués.

Cas de processus identiquement distribués

Supposons que les processus $X_1, \ldots, X_n$ sont issus de la même distribution, c'est-à-dire tels que

$$X_i(t_\ell) = s(t_\ell) + \sigma\varepsilon_{i,\ell}, \quad \ell = 1, \ldots, N, \tag{3.3.5}$$

où les variables aléatoires $\varepsilon_{i,\ell}$ sont i.i.d. de loi $\mathcal{N}_1(0,1)$. Les coefficients d'ondelettes de s peuvent être estimés à partir du signal moyen $\bar{X} := \frac{1}{n}\sum_{i=1}^n X_i$ qui satisfait le modèle

$$\bar{X}(t_\ell) = s(t_\ell) + \frac{\sigma}{\sqrt{n}}\varepsilon_\ell, \quad \ell = 1, \ldots, N,$$

où les variables ε_ℓ sont i.i.d. de loi $\mathcal{N}_1(0,1)$. En appliquant la règle du seuillage dur à ce signal on obtient des estimateurs $\hat{\omega}_0 = \zeta$ et

$$\hat{\omega}_{jk} = \begin{cases} \bar{\xi}_{jk} & \text{if } |\bar{\xi}_{jk}| > \bar{\delta}_N \\ 0 & \text{sinon,} \end{cases}$$

où $\bar{\xi}_{jk}$ est le coefficient d'ondelettes de niveau (j,k) de $\bar{X}$. Le seuil est alors donné par $\bar{\delta}_N = \frac{\sigma}{\sqrt{n}}\sqrt{2\log(N)}$.

Cas de processus non identiquement distribués

Dans de nombreuses applications, supposer que les n courbes observées proviennent d'une même distribution n'est pas une hypothèse réaliste. Dans le cas des données de vol, par exemple, la distribution des vols "normaux" (sous des conditions sures) n'a aucune raison d'être la même que celle des vols "à risque".

Nous proposons donc une généralisation du modèle (3.3.5) au cas de processus non identiquement distribués. Définissons tout d'abord une variable aléatoire Z à valeurs dans un ensemble $\mathcal{Z}$. Cette variable représente tous les phénomènes aléatoires que l'on peut observer sur le signal moyen. Conditionnellement à $Z_i = z_i$, la distribution du processus X_i est maintenant définie, pour tout $i \in \{1,\dots,n\}$, par

$$X_i(t_\ell) = s(t_\ell, z_i) + \sigma\varepsilon_{i,\ell}\,, \quad \ell = 1,\dots,N, \tag{3.3.6}$$

où $\varepsilon_{i,\ell}$ sont des variables aléatoires $\mathcal{N}_1(0,1)$ et $\sigma > 0$. À partir de ce modèle, nous pouvons considérer un grand nombre de situations. Dans un contexte d'apprentissage supervisé où une variable Y est à prédire sachant X, un choix raisonnable serait de prendre $Z = Y$.

Nous introduisons à présent une règle de seuillage dur pour le modèle (3.3.6). Soit ξ_{ijk}, le coefficient d'ondelettes de niveau (j,k) de X_i et soit $\boldsymbol{\xi}_{jk} := (\xi_{1jk},\dots,\xi_{ijk},\dots,\xi_{njk})^\top$. Pour tout $z \in \mathcal{Z}$, soit $\omega_{jk}(z)$ le coefficient de $s(\cdot,z)$ et $\boldsymbol{\omega}_{jk} := (\omega_{jk}(Z_1),\dots,\omega_{jk}(Z_n))^\top$. Définissons de plus l'ensemble

$$L := \{(j,k)\,|\,\omega_{jk}(Z) = 0\ p.s.\}\,,$$

comme le support commun des coefficients d'ondelettes de s. Si $(j,k) \in L$, alors $\boldsymbol{\omega}_{jk} = (0,\dots,0)^\top$ presque sûrement et $\|\boldsymbol{\xi}_{jk}\|_2^2$ suit une loi du Chi 2 centrée à n degrés de liberté. Dans le cas contraire, $\boldsymbol{\omega}_{jk}$ n'est pas identiquement nul et la variable aléatoire $\|\boldsymbol{\xi}_{jk}\|_2^2$ suit une loi du Chi 2 décentrée. Finalement, la règle du seuillage dur appliquée à $\|\boldsymbol{\xi}_{jk}\|_2^2$ est, pour tout $j \in \{0,\dots,J-1\}$ et tout $k \in \{0,\dots,2^j-1\}$,

$$\hat{\boldsymbol{\omega}}_{jk} = \begin{cases} \boldsymbol{\xi}_{jk} & \text{if } \|\boldsymbol{\xi}_{jk}\|_2 > \delta_{N,n} \\ (0,\dots,0)^\top & \text{sinon,} \end{cases} \tag{3.3.7}$$

où le seuil $\delta_{N,n}$ dépend de N, n et σ.

Un résultat élémentaire de consistance peut alors être démontré en utilisant l'inégalité de déviation suivante, tirée de Laurent and Massart (2000, p. 1325) :

Proposition 2. *Soit W une variable aléatoire suivant une loi du Chi 2 centrée à n degrés de liberté. Alors pour tout $x > 0$,*

$$\mathbb{P}[W - n \geq 2\sqrt{nx} + 2x] \quad \leq \quad e^{-x}.$$

À partir de ce résultat, nous montrons que pour tout $(j,k) \in L$, $\hat{\boldsymbol{\omega}}_{jk} = (0,\dots,0)^\top$ avec grande probabilité. Pour cela, supposons que pour un certain réel $x > 0$, le seuil $\delta^2_{N,n}$ est de la forme $\delta^2_{N,n}(x) = \sigma^2(2x + 2\sqrt{nx} + n)$. Alors, en utilisant la Proposition 2,

$$\begin{aligned}
\mathbb{P}\left[\bigcup_{(j,k)\in L}\left\{\hat{\boldsymbol{\omega}}_{jk} \neq (0,\dots,0)^\top\right\}\right] &\leq \sum_{(j,k)\in L}\mathbb{P}\left[\|\boldsymbol{\xi}_{jk}\|_2^2 \geq \delta^2_{N,n}(x)\right] \\
&= \sum_{(j,k)\in L}\mathbb{P}\left[\frac{\|\boldsymbol{\xi}_{jk}\|_2^2}{\sigma^2} - n \geq 2x + 2\sqrt{nx}\right] \\
&\leq |L|e^{-x} \leq Ne^{-x} \qquad (3.3.8)
\end{aligned}$$

Ce résultat nous montre que si le signal est identiquement nul, la règle du seuillage (3.3.7) estime le vecteur $\boldsymbol{\omega}_{jk}$ exactement à 0 avec une probabilité qui tend vers 1 en prenant $x \gg \log(N)$. En particulier, si $x = 2\log(N)$, la vitesse de convergence est de $O(\frac{1}{N})$. Ce choix correspond au seuil $\delta_{N,n} = (4\log(N) + 2\sqrt{2n\log(N)} + n)^{\frac{1}{2}}$.

En pratique, x et $\delta_{N,n}$ peuvent être choisis de sorte que la borne supérieure Ne^{-x} soit égale à une probabilité q souhaitée (0.05 ou 0.01 par exemple). Si l'on pose $Ne^{-x} = q$, nous obtenons le seuil

$$\delta_{N,n} = \left(2\log\left(\frac{N}{q}\right) + 2\sqrt{n\log\left(\frac{N}{q}\right)} + n\right)^{\frac{1}{2}}.$$

L'écart-type σ est estimé par l'estimateur MAD des n groupes de coefficients du niveau de résolution le plus fin.

Notons que Pigoli and Sangalli (2012) ont proposé un estimateur similaire pour le seuillage simultané de n courbes (non nécessairement indépendantes) en suggérant le seuil $\delta_{N,n} = \hat{\sigma}\sqrt{3\log(N)}$ sur $\|\boldsymbol{\xi}_{jk}\|_2$. Les auteurs montrent que quelque soit $n \geq 2$,

$$\mathbb{P}\left[\max_{jk}\|\boldsymbol{\xi}_{jk}\|_2 \leq \sigma\delta_{N,n}(x)\right] \underset{N\to\infty}{\longrightarrow} 1.$$

Cependant, le seuil proposé ne dépend pas de n. Dès lors que le terme $\|\boldsymbol{\xi}_{jk}\|_2$ croît en n, le seuil devrait lui même s'adapter cette dimension. En conséquence, la valeur $\hat{\sigma}\sqrt{3\log(N)}$ peut devenir très conservative si n est très grand. Dans leurs expérimentations, les auteurs considèrent des valeurs faibles pour n, respectivement 3 pour les simulations et 8 pour les données réelles. Dans notre cadre de travail, nous faisons face à des centaines de courbes et leur approche ne donne pas de résultats concluants dans ce cas.

Pour conclure la section, précisons que des méthodes de débruitage de signaux multivariés ont été considérées par Aminghafari et al. (2006) et Mostacci et al. (2010). Les démarches proposées ne s'appliquent pas dans notre cadre de travail car nous supposons que les courbes sont indépendantes. En conséquence, l'approche introduite par Aminghafari et al. (2006) reviendrait à effectuer un seuillage univarié indépendamment pour chaque courbe.

3.3.4 Illustration numérique et commentaires

La méthode de réduction de dimension proposée dépend du seuil $\delta_{N,n}(x)$. S'il est choisi tel que $x \gg \log(N)$, alors la méthode est consistante. Nous avons testé la méthode sur les signaux observés durant la phase d'approche (jeu de données `Reg1`). Ils sont échantillonnés

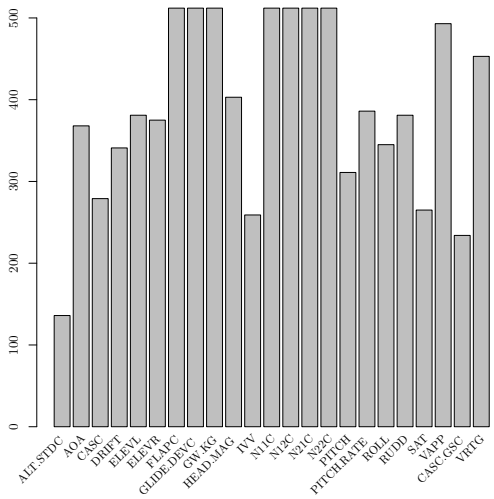


Figure 3.3 – Nombre de coefficients d'ondelettes retenus par seuillage simultané.

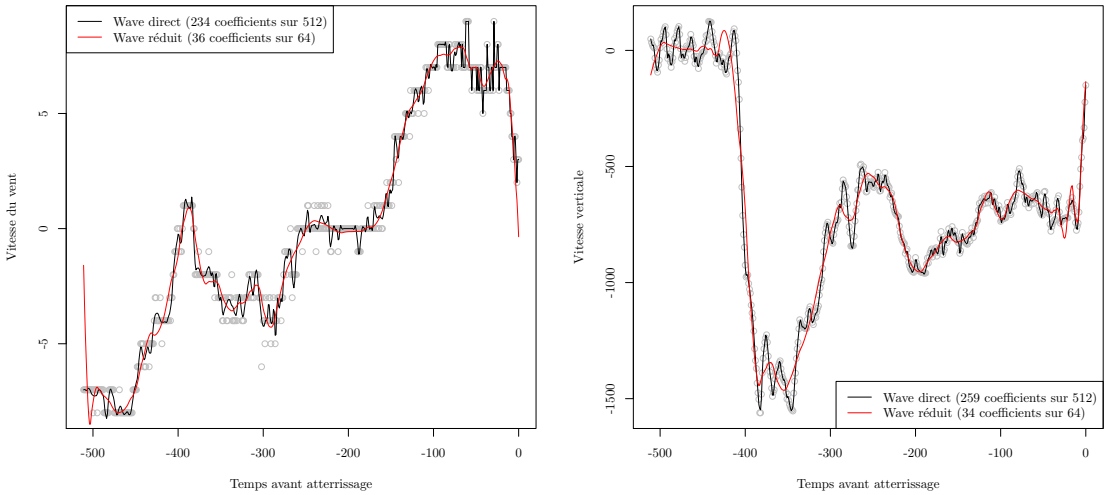
sur une grille de temps de $N = 512$ points. Pour un seuil de $\delta_{N,n}(100\log(N))$, la méthode proposée a conservé un nombre encore trop important de coefficients (voir la Figure 3.3). La réduction de dimension est insuffisante du point de vue algorithmique. Nous proposons donc de réduire la dimension en deux temps. Tout d'abord, nous éliminons les niveaux d'ondelettes les plus fins pour arriver à une dimension raisonnable (de 64 par variable par exemple). Nous seuillons ensuite les coefficients restants. À titre de comparaison, la Figure 3.4 représente deux courbes de vitesse du vent et de vitesse verticale après réduction directe des 512 premiers coefficients d'ondelettes ("Wave direct" dans la légende) et après le seuillage des 64 premiers coefficients ("Wave réduit" dans la légende). Les n signaux sont réduits simultanément et nous représentons la reconstruction d'une courbe pour les deux approches. Nous observons qu'effectivement, le seuillage "direct" est trop conservateur. Sur cet exemple, en seuillant les 64 premiers coefficients, la structure des signaux est conservée.

Notons que cette solution n'est pas complètement satisfaisante d'un point de vue méthodologique. Elle permet cependant d'accélérer les temps de calculs en garantissant un niveau d'erreur de prédiction raisonnable.

3.4 Réduction de dimension par analyse en composantes principales fonctionnelle

L'analyse en composantes principales (ACP) a été adaptée au cas de variables aléatoires à valeurs dans un espace de Hilbert suite aux travaux de Deville (1974), Dauxois and Pousse (1976) et Dauxois et al. (1982). Au même titre que l'ACP multivariée, elle permet de réduire la dimension des données en estimant la décomposition de Karhunen-Loève d'un processus $X = \{X(t), t \in [0,1]\}$ de $L^2([0,1])$:

$$X(t) = \mu(t) + \sum_{k=1}^{\infty} \langle X, \varphi_k \rangle_{L^2} \varphi_k(t),$$



(a) Vitesse du vent (b) Vitesse verticale

FIGURE 3.4 – Réduction de dimension par seuillage simultané des ondelettes. "Wave direct" correspond au seuillage des 512 coefficients initiaux et "Wave réduit" correspond au seuillage des 64 premiers coefficients.

avec $\mu(t) = \mathbb{E}X(t)$. Les fonctions φ_k sont les fonctions propres de l'opérateur de covariance de X défini par

$$V\varphi = \int_0^1 C(\cdot, t)\varphi(t)dt,$$

avec $C(s,t) = \mathrm{Cov}(X(s), X(t))$. Ces fonctions sont solutions de l'équation

$$V\varphi_k = \lambda_k \varphi_k, \tag{3.4.1}$$

où λ_k est la valeur propre de l'opérateur V associée à la fonction propre φ_k. Cette équation, analogue fonctionnel de l'ACP standard, permet de construire une base de fonctions orthogonales. En conséquence, les variables aléatoires $\langle X, \varphi_1\rangle_{L^2}, \langle X, \varphi_2\rangle_{L^2}, \ldots$ sont non corrélées et de variance λ_k. Voir l'article de Besse and Cardot (2003) ainsi que les livres de Ramsay and Silverman (2005) et Horváth and Kokoszka (2012) pour un tour d'horizon plus exhaustif. Une propriété intéressante de l'ACPF est que la somme tronquée $\sum_{k=1}^{d}\langle X, \varphi_k\rangle_{L^2}\varphi_k(t)$ est la meilleure approximation linéaire au sens L^2 de X dans un espace de dimension d.

En pratique, la résolution de l'équation (3.4.1) n'admet pas de solution analytique. Il existe cependant différentes méthodes d'approximation de ce problème pour estimer les éléments propres (λ_k, φ_k) à partir de n réalisations du processus X. La première est une approche naïve qui consiste effectuer une ACP sur les courbes discrétisées, c'est-à-dire sur la matrice de taille $n \times N$. Cependant, cette approche peut conduire à des estimations instables des éléments propres. C'est pourquoi une méthode alternative, plus robuste, consiste à lisser les courbes puis à effectuer une ACP sur les K premiers coefficients de base (Ramsay and Silverman; 2005). Un lissage Spline est en général employé à cet effet. Cette méthode est aussi valable dans le cas où la grille de temps n'est pas régulière (James et al.; 2000). En pratique, nous choisissons le degré de lissage K *a priori* de sorte à rester

suffisamment conservateur. L'utilisation des Splines de lissage (Green and Silverman; 1994) et un critère de validation croisée sont une alternative pour choisir K de façon automatique. Une fois les fonctions propres estimées, les coefficients de base empiriques sont donnés par $Z_k = \frac{1}{N}\sum_{\ell=1}^{N} X(t_\ell)\hat{\varphi}_k(t_\ell)$.

Le choix final des composantes principales se fait par l'analogue fonctionnel de la part de variance expliquée par les d premières composantes principales :

$$\begin{aligned}\nu_d &= \frac{\mathrm{Var}\left(\sum_{k=1}^{d} Z_k\hat{\varphi}_k(t)\right)}{\mathrm{Var}\left(\sum_{k=1}^{K} Z_k\hat{\varphi}_k(t)\right)} \\ &= \frac{\sum_{k=1}^{d}\hat{\lambda}_k}{\sum_{k=1}^{K}\hat{\lambda}_k},\end{aligned}$$

où K est le nombre de coefficients Spline mesurant le degré du lissage.

Notons, de plus, que cette méthode permet de construire une représentation compacte d'un processus X. En effet, les éléments propres (λ_k, φ_k) sont estimés de sorte à maximiser la variance de X. Dans un cas favorable, Les valeurs propres estimées $\hat{\lambda}_1, \hat{\lambda}_2, \ldots$ décroissent ainsi rapidement vers 0. En guise d'illustration, nous représentons un vol après la réduction de dimension par ACPF des n vols pour quatre paramètres de vol : la vitesse du vent, la vitesse verticale, l'angle de dérive et la vitesse air (Figure 3.5). Nous construisons au préalable un lissage Spline avec $K = 64$ coefficients (la valeur de K est choisie *a priori*). Les composantes principales retenues sont celles qui expliquent respectivement 90 % et 99 % de la variance totale. Nous représentons également le résultat du seuillage par ondelettes. Les données sont réduites simultanément par le seuillage des 64 premiers coefficients d'ondelettes avec le seuil $\delta_{N,n}(100\log(N))$.

Ces graphiques confirment que l'ACPF représente les signaux de façon beaucoup plus compacte que les ondelettes. Cependant, à l'inverse des ondelettes, la base de Karhunen-Loève n'est pas localisée en temps et en fréquence.

Jusqu'ici, nous avons introduit l'approche par projection pour l'analyse de données fonctionnelles. Nous avons également présenté diverses méthodes de réduction de dimension. Dans la section suivante, nous dressons un état de l'art non exhaustif de méthodes d'apprentissage supervisé pour données fonctionnelles.

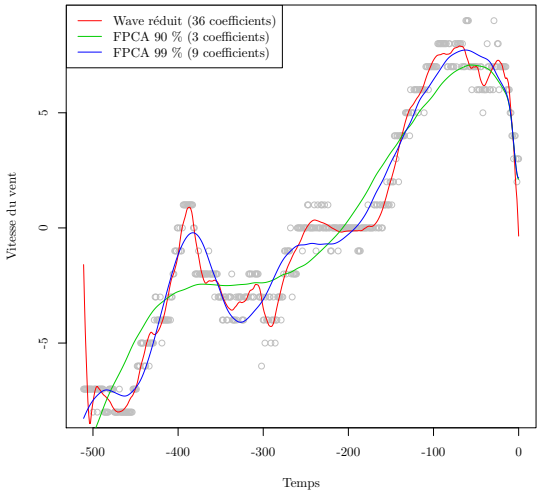
3.5 Méthodes d'apprentissage supervisé pour données fonctionnelles

Modèles linéaires fonctionnels

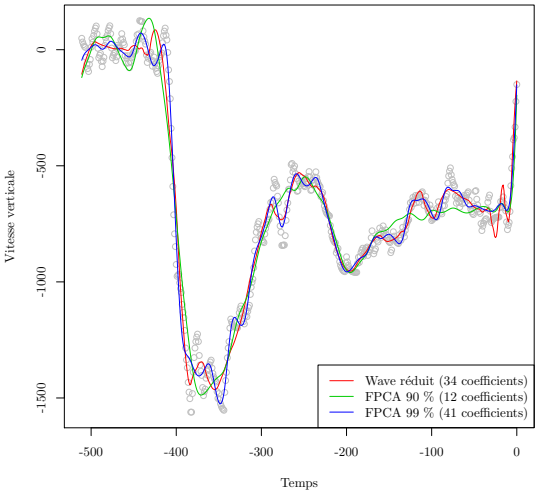
Le modèle linéaire fonctionnel généralise le modèle linéaire classique où le produit scalaire réel est remplacé par le produit scalaire de $L^2([0,1])$, c'est-à-dire,

$$Y = \sum_{u=1}^{p}\int_0^1 X_u(t)\beta_u(t)dt + \varepsilon, \tag{3.5.1}$$

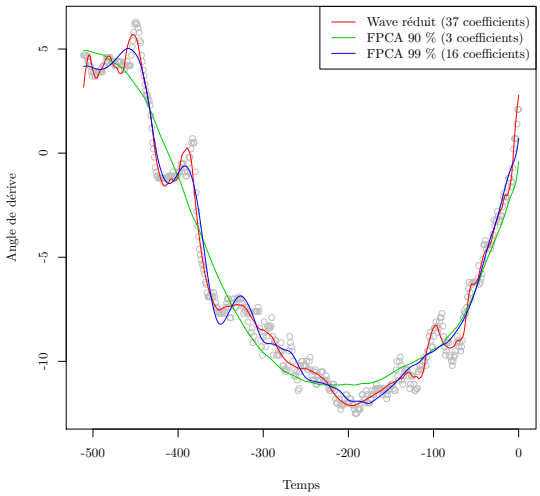
où ε est une variable aléatoire réelle centrée conditionnellement à $\mathbf{X} = (X_1, \ldots, X_p)$ et $\beta_1, \ldots, \beta_p$ sont des fonctions de $L^2([0,1])$ à estimer. Le cas où $p = 1$ a été très largement étudié, par exemple dans Cardot et al. (1999), Cardot et al. (2003), Ferraty and Vieu (2002, 2003, 2009), Ramsay and Silverman (2005), Cai and Hall (2006), González-Mantegna and



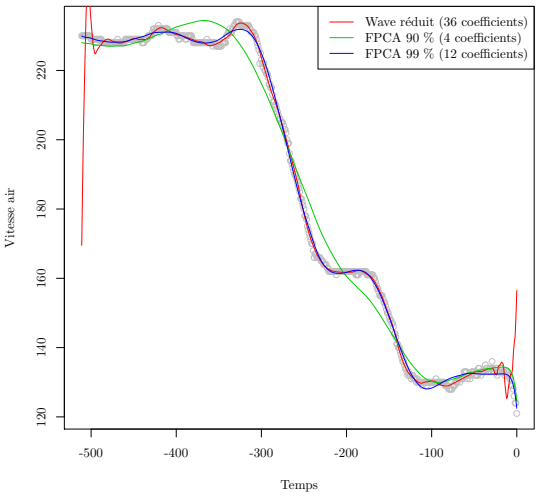
(a) Vitesse du vent



(b) Vitesse verticale



(c) Angle de dérive



(d) Vitesse air

FIGURE 3.5 – Exemples de courbes reconstruites après réduction de dimension. “Wave réduit” représente le seuillage par ondelettes sur les 64 premiers coefficients. FPCA 90 % et FPCA 99 % correspondent à respectivement 90 % et 99 % de variance expliquée.

Martínez-Calvo (2011), Randolph et al. (2012) et Tian and James (2013). Une extension au cas où p covariables fonctionnelles sont observées a été récemment proposée, voir Matsui and Konishi (2011), Fan and James (2013), Oliva et al. (2014) et Amini et al. (2013).

L'estimation du modèle (3.5.1) telle que le décrivent notamment Fan and James (2013) et Oliva et al. (2014) est basée sur la minimisation d'un critère pénalisé de type *Group Lasso* (Yuan and Lin; 2006a). Plus précisément, les fonctions X_u et β_u sont projetées sur un sous-espace de dimension finie via une base $\mathcal{B} = \{\varphi_1, \varphi_2, \ldots\}$:

$$X_u(t) = \sum_{k=1}^{d_u} Z_{uk}\varphi_k(t) + \delta_u(t)$$

et

$$\beta_u(t) = \sum_{k=1}^{d_u} \theta_{uk}\varphi_k(t) + \gamma_u(t),$$

avec $Z_{uk} = \langle X_u, \varphi_k\rangle_{L^2}$ et $\theta_{uk} = \langle \beta_u, \varphi_k\rangle_{L^2}$. Les termes $\delta_u(t)$ et $\gamma_u(t)$ sont les termes d'erreur de la projection sur $\{\varphi_1, \ldots, \varphi_{d_u}\}$. Les coefficients Z_{uk} sont approximés par les coefficients empiriques $\frac{1}{N}\sum_{k=1}^{N} X_u(t_\ell)\varphi_k(t_\ell)$.

Sous l'hypothèse de l'orthogonalité des fonctions de base, le modèle (3.5.1) se réécrit

$$Y = \sum_{u=1}^{p} \mathbf{Z}_u^\top \theta_u + \epsilon,$$

où $\mathbf{Z}_u = (Z_{u1}, \ldots, Z_{ud_u})$, $\theta_u = (\theta_{u1}, \ldots, \theta_{ud_u})$ et ϵ est une variable aléatoire réelle représentant l'erreur du modèle ε et les erreurs d'approximation $\delta_u(t)$ et $\gamma_u(t)$. Le vecteur des paramètres est estimé via un critère *Group Lasso* où les groupes sont formés par les vecteurs aléatoires $\mathbf{Z}_1, \ldots, \mathbf{Z}_p$:

$$\hat{\theta} \in \underset{\theta\in\mathbb{R}^D}{\arg\min} \left\{ \|Y - \sum_{u=1x}^{p} \mathbf{Z}_u^\top \theta_u\|_2^2 + \lambda \sum_{u=1}^{p} \sqrt{d_u}\|\theta_u\|_2 \right\},$$

où $D = \sum_{u=1}^{p} d_u$. Le terme de pénalité impose de la parcimonie sur les groupes et induit une sélection des variables fonctionnelles. Notons que le critère proposé par Fan and James (2013) est inspiré de Simon and Tibshirani (2012).

Modèle logistique fonctionnel

La régression logistique a également été adaptée aux données fonctionnelles notamment par Cardot and Sarda (2005), Leng and Müller (2006) et Araki et al. (2009). Pour une variable fonctionnelle X et une variable de sortie Y binaire, les probabilités *a posteriori* $\eta(X) = \mathbb{P}[Y = 1|X]$ et $1 - \eta(X) = \mathbb{P}[Y = -1|X]$ satisfont

$$\log\left(\frac{\eta(X)}{1-\eta(X)}\right) = \int_0^1 X(t)\beta(t)dt.$$

La fonction β est estimée de façon similaire au modèle linéaire fonctionnel. Il s'agit d'approximer β et X respectivement par $\sum_{k=1}^{d} \theta_k\varphi_k(t)$ et $\sum_{k=1}^{d} Z_k\varphi_k(t)$ ce qui permet de déduire une expression de la fonction de vraisemblance. Un estimateur $\hat{\beta}$ est finalement calculé en maximisant la log-vraisemblance. Notons que Araki et al. (2009) choisissent de pénaliser la log-vraisemblance par une pénalité *Ridge* afin de contraindre les solutions à être lisses.

Le modèle logistique fonctionnel à p covariables a peu été étudié à notre connaissance. L'estimation des paramètres est essentiellement basée sur la maximisation de la log-vraisemblance pénalisée par un terme de type *Group Lasso* (Meier et al.; 2008). Comme pour le modèle linéaire, ce terme de pénalité permet de sélectionner les variables fonctionnelles. Des modèles linéaires généralisés ont ainsi été proposés par Zhu and Cox (2009) et Gertheiss et al. (2013). Citons également Matsui (2014) qui étendent le modèle de Araki et al. (2009) et considèrent plusieurs pénalités différentes.

Approches non paramétriques

Les méthodes de classification et de régression non paramétriques ont également été adaptées au cadre fonctionnel, dans un premier temps pour une seule variable fonctionnelle X. Nous pouvons citer les k plus proches voisins fonctionnels pour la classification (Biau et al.; 2005, Fromont and Tuleau; 2006) et pour la régression (Laloë; 2008). Les auteurs projettent X sur une base de Fourier puis estiment simultanément la règle de Bayes et la dimension de projection au moyen de la minimisation d'un critère pénalisé. L'approche est validée par un résultat de type Oracle sur l'estimateur obtenu. Berlinet et al. (2008) adoptent la même méthodologie et choisissent une base d'ondelettes pour réduire la dimension de X. Plusieurs algorithmes sont testés en classification : les k plus proches voisins, l'analyse discriminante quadratique et l'algorithme CART. Les SVM sont considérés de façon similaire par Rossi and Villa (2006, 2008). Enfin, Song et al. (2008) comparent plusieurs méthodes de classification (analyses discriminantes linéaire et quadratique, k plus proches voisins et SVM).

Dans le cas multivarié, il existe peu de travaux à ce jour. Citons les travaux de Yang et al. (2005) et Yoon and Shahabi (2006) qui combinent le coefficient de corrélation et l'algorithme SVM-RFE (Guyon et al.; 2002) pour sélectionner des séries temporelles multivariées. L'article de Poggi and Tuleau (2006) est motivé par une application à l'agrément de conduite automobile. La démarche adoptée par les auteurs utilise l'algorithme CART et les ondelettes pour sélectionner des variables fonctionnelles. Elle se décline en trois étapes. Tout d'abord, les signaux sont débruités individuellement par seuillage d'ondelettes puis normalisés et recalés. Les signaux sont ensuite compressés en sélectionnant une base commune à toutes les observations et indépendamment pour chaque variable fonctionnelle X_u. Il s'agit d'identifier les niveaux d'ondelettes qui permettent une bonne approximation des variables. Le critère de choix est l'erreur quadratique entre X_u et le signal approximé par les j premiers niveaux d'ondelettes. Enfin, l'algorithme CART est utilisé pour sélectionner les groupes d'ondelettes correspondant aux variables explicatives fonctionnelles. Pour chaque variable fonctionnelle, un arbre CART est construit et l'erreur de classification est estimée par validation croisée. Un classement des groupes de coefficients est alors déduit des erreurs commises par chaque arbre. Selon ce classement initial, des arbres CART sont construits en ajoutant itérativement les groupes de coefficients les plus informatifs. Finalement, les variables fonctionnelles discriminantes sont sélectionnées en minimisant l'erreur de classification évaluée à chaque itération de l'algorithme. Cette procédure est un algorithme de sélection *forward* non récursif où le critère d'importance des variables (fonctionnelles) est l'erreur de classification estimée par validation croisée.

Dans la Section 4.6 du chapitre suivant, nous sélectionnons les coefficients d'ondelettes par l'algorithme RFE avec les forêts aléatoires et la mesure d'importance par permutation. Les variables fonctionnelles finalement choisies sont celles dont les coefficients sont les plus fréquemment sélectionnés. Ainsi, nous nous différentions de Poggi and Tuleau (2006) en employant une procédure de sélection récursive *backward* qui calcule le critère d'importance à chaque étape de l'algorithme.

Chapter 4

Corrélation et importance des variables dans les forêts aléatoires

Abstract. This paper is about variable selection with the random forests algorithm in presence of correlated predictors. In high-dimensional regression or classification frameworks, variable selection is a difficult task, that becomes even more challenging in the presence of highly correlated predictors. Firstly we provide a theoretical study of the permutation importance measure for an additive regression model. This allows us to describe how the correlation between predictors impacts the permutation importance. Our results motivate the use of the Recursive Feature Elimination (RFE) algorithm for variable selection in this context. This algorithm recursively eliminates the variables using permutation importance measure as a ranking criterion. Next various simulation experiments illustrate the efficiency of the RFE algorithm for selecting a small number of variables together with a good prediction error. Finally, this selection algorithm is tested on a real life dataset from aviation safety where the flight data recorders are analysed for the prediction of a dangerous event.

Ce chapitre fait l'objet d'un article écrit en collaboration avec Bertrand Michel et Philippe Saint Pierre, maîtres de conférence à l'Université Pierre et Marie Curie (Gregorutti et al.; 2014a).

Contents

4.1 Introduction

This paper is motivated by a real life problem in the context of aviation safety. The recent recommendations of the International Civil Aviation Organisation forces airlines to evaluate the risk of incidents and more generally to ensure the operational safety. Indeed, flight data recorders provide a large amount of raw data. The available data contains many flight parameters which are highly correlated. A challenging problem is to provide an efficient prediction of dangerous events. In addition, an interpretable model based on a small number of flight parameters is crucial for pilot training and developing new flight procedures. Variable selection techniques is a natural answer to this issue.

In such large scale learning problems, in particular when the number of variables is much larger than the number of observations, not all of the variables are relevant for predicting the outcome of interest. Some irrelevant variables may have a negative effect on the model accuracy. Variable selection techniques, also called feature selection or subset selection, involve eliminating irrelevant variables and provide two main advantages. First, a model with a small number of variables is more interpretable. Secondly, the model accuracy might be improved and then avoid the risk of overfitting.

Many studies about variable selection have been conducted during the last decade. In Guyon and Elisseeff (2003), the authors review three existing approaches: filter, embedded and wrapper. A filter algorithm, also known as variable ranking, orders the variables in a preprocessing step and the selection is done independently of the choice of the learning technique. Two classical ranking criteria are the Pearson correlation coefficient and the mutual information criterion as mentioned in the recent survey of Lazar et al. (2012). The main drawback of this approach is that the choice of the selected variables is not induced by the performance of the learning method. The embedded approach selects the variables during the learning process. The two main examples are the Lasso (Tibshirani; 1996) for regression problems (the selection process is done through the ℓ_1 regularization of the least square criterion) and decision trees (the selection is induced by the automatic selection of the splitting variables) such as CART algorithm (Breiman et al.; 1984). A wrapper algorithm uses the learning process to identify an optimal set of variables among all possible subsets (see Kohavi and John; 1997; Blum and Langley; 1997). The measure of optimality is usually defined by the error rate estimate. As it is impossible to evaluate all variable subsets when the dimension of the data is too large, the wrapper approach consists of using greedy strategies such as forward or backward algorithms. A heuristic is required to select the variables to be introduced or eliminated. This algorithm has been adapted for various contexts in the literature (see for instance Guyon et al.; 2002; Rakotomamonjy; 2003; Svetnik et al.; 2004; Díaz-Uriarte and Alvarez de Andrés; 2006; Louw and Steel; 2006; Genuer et al.; 2010).

A classical issue of variable selection methods is their instability: a small perturbation of the training sample may completely change the set of selected variables. This instability is a consequence of the data complexity in high dimensional settings (see Kalousis et al.; 2007; Křížek et al.; 2007). In particular, the instability of variable selection methods increases when the predictors are highly correlated. For instance, Bühlmann et al. (2013) have shown that the lasso tends to discard most of the correlated variables even if they are discriminants and randomly selects one representative among a set of correlated predictors. In the context of random forests, the impact of correlated predictors on variable selection methods has been highlighted by several simulation studies, see for instance Toloşi and Lengauer (2011). For real life applications it is of first importance to select a subset of variables which is the most stable as possible. One popular solution to answer

the instability issue of variable selection methods consists in using bootstrap samples: a selection is done on several bootstrap subsets of the training data and a stable solution is obtained by aggregation of these selections. Such generic approach aims to improve both the stability and the accuracy of the method. This procedure is known as "ensemble feature selection" in the machine learning community. Several classification and regression techniques based on this approach have been developed (Bi et al.; 2003 in the context of Support vector regression, Meinshausen and Bühlmann; 2010 with the stability selection). Haury et al. (2011) provide a comparison of ensemble feature selections combined with several classification methods.

The random forests algorithm, introduced by Breiman (2001), is a modification of bagging that aggregates a large collection of tree-based estimators. This strategy has better estimation performances than a single random tree: each tree estimator has low bias but high variance whereas the aggregation achieve a bias-variance trade-off. The random forests are very attractive for both classification and regression problems. Indeed, these methods have good predictive performances in practice, they work well for high dimensional problems and they can be used with multi-class output, categorical predictors and imbalanced problems. Moreover, the random forests provide some measures of the importance of the variables with respect to the prediction of the outcome variable.

Several studies have used the importance measures in variable selection algorithms (Svetnik et al.; 2004; Díaz-Uriarte and Alvarez de Andrés; 2006; Genuer et al.; 2010). The effect of the correlations on these measures has been studied in the last few years by Archer and Kimes (2008); Strobl et al. (2008); Nicodemus and Malley (2009); Nicodemus et al. (2010); Nicodemus (2011); Auret and Aldrich (2011) and Toloşi and Lengauer (2011). However, there is no consensus on the interpretation of the importance measures when the predictors are correlated and more precisely there is no consensus on what is the effect of this correlation on the importance measures (see e.g. Grömping; 2009; Neville; 2013). One reason for this is that, as far as we know, no theoretical description of this effect has been proposed in the literature. This situation is particularly unsatisfactory as the importance measures are intensively used in practice for selecting the variables.

The contributions of this paper are two-fold. First, we give some theoretical descriptions of the effect of the correlations on the ranking of the variables produced by the permutation importance measure introduced in Breiman (2001). More precisely, we consider a particular additive regression model for which it is possible to express the permutation importance in function of the correlations between predictors. The results of this section are validated by a simulation study.

The second contribution of this paper is of algorithmic nature. We take advantage of the previous results to compare wrapper variable selection algorithms for random forests in the context of correlated predictors. Note that most of the variable selection procedures using the random forests are wrapper algorithms. It can be used also as a filter algorithm as in Hapfelmeier and Ulm (2013). Two main wrapper algorithms are considered in the literature. These two rely on backward elimination strategies based on the ranking produced by the permutation importance measure. The first algorithm computes the permutation importance measures in the full model which produces a ranking of the variables. This ranking is kept unchanged by the algorithm (Svetnik et al.; 2004; Díaz-Uriarte and Alvarez de Andrés; 2006; Genuer et al.; 2010). The second algorithm was first proposed by Guyon et al. (2002) in the context of support vector machines (SVM) and is referred to as Recursive Feature Elimination (RFE). This algorithm requires to update the ranking criterion at each step of a backward strategy: at each step the criterion is evaluated and the variable which minimizes this measure is eliminated. In the random forests setting,

although less popular than the first one, this strategy has been implemented for instance in Jiang et al. (2004). As far as we know, only one study by Svetnik et al. (2004) compared the two approaches and concluded that the non recursive algorithm provides better results. However, their findings are based on a real life dataset without taking into account the effect of correlated predictors and are not confirmed by simulation studies. Moreover, this position goes against the results we find in Section 4.3.

A simulation study has been performed to compare the performances of the recursive and the non recursive strategies. Several designs of correlated data encounters in the literature have been simulated for this purpose. As expected, the simulations indicate that the recursive algorithm provides better results.

The paper is organized as follows. We first introduce the statistical background of the random forests algorithm and the permutation importance measure. Section 4.3 provides some theoretical properties of this criterion in the special case of an additive regression model. Section 4.4 describes the RFE algorithm used for variable selection in a random forests analysis. Next, the effect of the correlations on the permutation importance and the good performances of the RFE algorithm in the case of correlated variables are emphasized in a simulation study. This algorithm is finally carried out for studying the risk of aircraft incidents using flight data.

4.2 Random forests and variable importance measures

Let us consider a variable of interest Y and a vector of random variables $\mathbf{X} = (X_1, \ldots, X_p)$. In the regression setting a rule $\hat{f}$ for predicting Y is a measurable function taking its values in $\mathbb{R}$. The prediction error of $\hat{f}$ is then defined by $R(\hat{f}) = \mathbb{E}\left[(\hat{f}(\mathbf{X}) - Y)^2\right]$ and our goal is to estimate the conditional expectation $f(\mathbf{x}) = \mathbb{E}[Y|\mathbf{X} = \mathbf{x}]$.

Let $\mathcal{D}_n = \{(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)\}$ be a learning set of n i.i.d. replications of $(\mathbf{X}, Y)$ where $\mathbf{X}_i = (X_{i1}, \ldots, X_{ip})$. Since the true prediction error of $\hat{f}$ is unknown in practice, we consider an estimator based on the observation of a validation sample $\bar{\mathcal{D}}$:

$$\hat{R}(\hat{f}, \bar{\mathcal{D}}) = \frac{1}{|\bar{\mathcal{D}}|} \sum_{i:(\mathbf{X}_i, Y_i) \in \bar{\mathcal{D}}} (Y_i - \hat{f}(\mathbf{X}_i))^2.$$

Classification and regression trees, particularly CART algorithm due to Breiman et al. (1984), are competitive techniques for estimating f. Nevertheless, these algorithms are known to be unstable insofar as a small perturbation of the training sample may change radically the predictions. For this reason, Breiman (2001) introduced the random forests as a substantial improvement of the decision trees. It consists in aggregating a collection of such random trees, in the same way as the bagging method also proposed by Breiman (1996): the trees are built over n_{tree} bootstrap samples $\mathcal{D}_n^1, \ldots, \mathcal{D}_n^{n_{tree}}$ of the training data $\mathcal{D}_n$. Instead of CART algorithm, a small number of variables is randomly chosen to determine the splitting rule at each node. Each tree is then fully grown or until each node is pure. The trees are not pruned. The resulting learning rule is the aggregation of all of the tree-based estimators denoted by $\hat{f}_1, \ldots, \hat{f}_{n_{tree}}$. The aggregation is based on the average of the predictions.

In parallel the random forests algorithm allows us to evaluate the relevance of a predictor thanks to variable importance measures. The original random forests algorithm computes three measures, the permutation importance, the z-score and the Gini importance. We focus here on the permutation importance due to Breiman (2001). Broadly

speaking, a variable X_j can be considered as important for predicting Y if by breaking the link between X_j and Y the prediction error increases. To break the link between X_j and Y, Breiman proposes to randomly permute the observations of the X_j's. It should be noted that the random permutations also breaks the link between X_j and the other covariates. The empirical permutation importance measure can be formalized as follows: define a collection of out-of-bag samples $\{\bar{\mathcal{D}}_n^t = \mathcal{D}_n \setminus \mathcal{D}_n^t,\, t = 1, \ldots, n_{tree}\}$ which contains the observations not selected in the bootstrap subsets. Let $\{\bar{\mathcal{D}}_n^{tj},\, t = 1, \ldots, n_{tree}\}$ denote the permuted out-of-bag samples by random permutations of the values of the j-th variable in each out-of-bag subsets. The empirical permutation importance of the variable X_j is defined by

$$\hat{I}(X_j) = \frac{1}{n_{tree}} \sum_{t=1}^{n_{tree}} \left[\hat{R}(\hat{f}_t, \bar{\mathcal{D}}_n^{tj}) - \hat{R}(\hat{f}_t, \bar{\mathcal{D}}_n^{t}) \right]. \tag{4.2.1}$$

This quantity is the empirical counterpart of the permutation importance measure $I(X_j)$, as formalized recently in Zhu et al. (2012). Let $\mathbf{X}_{(j)} = (X_1, \ldots, X_j', \ldots, X_p)$ be the random vector such that X_j' is an independent replication of X_j which is also independent of Y and of all of the others predictors, the permutation importance measure is given by

$$I(X_j) = \mathbb{E}\left[\left(Y - f(\mathbf{X}_{(j)}) \right)^2 \right] - \mathbb{E}\left[(Y - f(\mathbf{X}))^2 \right].$$

The permutation of the values of X_j in the definition of $\hat{I}(X_j)$ mimics the independence and the identical copy of the distribution of X_j in the definition of $I(X_j)$. Zhu et al. (2012) show that $\hat{I}(X_j)$ converges to $I(X_j)$ at an exponential rate for particular tree-based algorithms, their result is given for the z-score in the regression but the proof given also provides the convergence for $\hat{I}(X_j)$.

The permutation importance measure can be used to rank or select the predictors. Among others criteria, the permutation importance measure has shown good performances for leading variable selection algorithms. Nevertheless variable selection is a difficult issue especially when the predictors are highly correlated. In the next section we investigate deeper the properties of the permutation importance measure in order to understand better how this quantity depends on the correlation between the predictors.

4.3 Permutation importance measure of correlated variables

Previous results about the impact of correlation on the importance measures are mostly based on experimental considerations. We give a non exhaustive review of these contributions and we compare them with our theoretical results. Archer and Kimes (2008) observe that the Gini measure is less able to detect the most relevant variables when the correlation increases and they mention that the same is true for the permutation importance. The experiments of Auret and Aldrich (2011) confirm these observations. Genuer et al. (2010) study the sensitivity of the empirical permutation importance measure to many parameters, in particular they study the sensitivity to the number of correlated variables. Recently, Toloşi and Lengauer (2011) identify what they call the "correlation bias". Note that it does not correspond to a statistical bias. More precisely, these authors observe two key effects of the correlation on the permutation importance measure: first, the importance values of the most discriminant correlated variables are not necessarily higher than a less discriminant one, and secondly the permutation importance measure depends on the size of the correlated groups.

Since previous studies are mainly based on numerical experiments, there is obviously

a need to provide theoretical validations of these observations. We propose below a first theoretical analysis of this issue, in a particular statistical framework. In the rest of the section, we assume that the random vector $(\mathbf{X}, Y)$ satisfies the following additive regression model:

$$Y = \sum_{j=1}^{p} f_j(X_j) + \varepsilon, \tag{4.3.1}$$

where ε is such that $\mathbb{E}[\varepsilon|\mathbf{X}] = 0$, $\mathbb{E}[\varepsilon^2|\mathbf{X}]$ is finite and the f_j's are measurable functions. In other words, we require that the regression function can be decomposed into $f(\mathbf{x}) = \sum_{j=1}^{p} f_j(x_j)$. In the sequel, $\mathbb{V}$ and $\mathbb{C}$ denote variance and covariance.

Proposition 3. *1. Under model (4.3.1), for any $j \in \{1, \dots, p\}$, the permutation importance measure satisfies*

$$I(X_j) = 2\mathbb{V}[f_j(X_j)].$$

2. Assume moreover that for some $j \in \{1, \dots, p\}$ the variable $f_j(X_j)$ is centered. Then,

$$I(X_j) = 2\mathbb{C}[Y, f_j(X_j)] - 2\sum_{k \neq j} \mathbb{C}[f_j(X_j), f_k(X_k)].$$

Proof. see Appendix 4.8 □

In this framework, the permutation importance corresponds to the variance of $f_j(X_j)$, up to a factor 2. The second result of Proposition 3 is the key point to study the influence of the correlation on the permutation measure. This result strongly depends on the additive structure of the regression function f and it seems difficult to give such a simple expression of the permutation importance without assuming this additive form for the regression function.

If $(\mathbf{X}, Y)$ is assumed to be a normal vector it is possible to specify the permutation importance measure. Note that in this context the conditional distribution of Y over $\mathbf{X}$ is also normal and the conditional mean f is a linear function: $f(\mathbf{x}) = \sum_{j=1}^{p} \alpha_j x_j$ with $\alpha = (\alpha_1, \dots, \alpha_p)^\top$ a sequence of deterministic coefficients (see for instance Rao; 1973, p. 522).

Proposition 4. *Consider a Gaussian random vector $(\mathbf{X}, Y) \sim \mathcal{N}_{p+1}\left(0, \begin{pmatrix} C & \boldsymbol{\tau} \\ \boldsymbol{\tau}^\top & \sigma_y^2 \end{pmatrix}\right)$, where $\boldsymbol{\tau} = (\tau_1, \dots, \tau_p)^\top$ with $\tau_j = \mathbb{C}(X_j, Y)$, $\sigma_y^2 > 0$ and $C = [\mathbb{C}(X_j, X_k)]$ is the non degenerated variance-covariance matrix of $\mathbf{X}$. Then, for any $j \in \{1, \dots, p\}$,*

$$I(X_j) = 2\alpha_j^2 \mathbb{V}(X_j) = 2\alpha_j \mathbb{C}(X_j, Y) - 2\alpha_j \sum_{k \neq j} \alpha_k \mathbb{C}(X_j, X_k),$$

where $\alpha_j = [C^{-1}\boldsymbol{\tau}]_j$.

Proof. see Appendix 4.8 □

Note that if we consider a linear function $f : \mathbb{R}^p \mapsto \mathbb{R}$, a random vector $\mathbf{X}$ of $\mathbb{R}^p$ and a random variable ε such that $(\mathbf{X}, \varepsilon)$ is a multivariate normal vector, and if we define the outcome by $Y = f(\mathbf{X}) + \varepsilon$, then the vector $(\mathbf{X}, Y)$ is clearly a multivariate normal vector. Thus, the assumption on the joint distribution of $(\mathbf{X}, Y)$ is in fact mild in the regression framework. Note also that Proposition 4 corresponds to the criterion used in Guyon et al. (2002) as a ranking criterion in the SVM-RFE algorithm with $\mathbb{V}(X_j) = 1$.

We now discuss the effect of the correlation between predictors on the importance measure by considering Gaussian regression models with various configurations of correlation between predictors:

Case 1: Two correlated variables. Consider the simple context where $(X_1, X_2, Y) \sim \mathcal{N}_3\left(0, \begin{pmatrix} C & \boldsymbol{\tau} \\ \boldsymbol{\tau}^\top & 1 \end{pmatrix}\right)$ with

$$C = \begin{pmatrix} 1 & c \\ c & 1 \end{pmatrix},$$

and $\boldsymbol{\tau}^\top = (\tau_0, \tau_0)$ with $\tau_0 \in (-1; 1)$. Since the two predictors have the same correlation τ_0 with the outcome, and according to Proposition 4, we have for $j \in \{1, 2\}$:

$$\alpha_j = \frac{\tau_0}{1+c}. \tag{4.3.2}$$

Consequently, the permutation importance for both X_1 and X_2 is

$$I(X_j) = 2\left(\frac{\tau_0}{1+c}\right)^2, \ j \in \{1, 2\}. \tag{4.3.3}$$

For positive correlations c, the importance of the two variables X_1 and X_2 decreases when c increases. This result is quite intuitive: when one of the two correlated variables is permuted, the error does not increase that much because of the presence of the other variable, which carries a similar information. The value of the prediction error after permutation is then close to the value of the prediction error without permutation and the importance is small.

Case 2: Two correlated and one independent variables. We add to the previous case an additional variable X_3 which is assumed to be independent of X_1 and X_2, X_1 and X_2 being unchanged. It corresponds to

$$C = \begin{pmatrix} 1 & c & 0 \\ c & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

and $\boldsymbol{\tau}^\top = (\tau_0, \tau_0, \tau_3)$. It can be easily checked that

$$C^{-1}(\tau_0, \tau_0, \tau_3)^\top = \left(\frac{\tau_0}{1+c}, \frac{\tau_0}{1+c}, \tau_3\right)^\top.$$

Thus, (4.3.2) and (4.3.3) still hold and $\alpha_3 = \tau_3$. As a consequence, $I(X_3) = 2\tau_3^2$ can be larger than $I(X_1)$ and $I(X_2)$ if the correlation c is sufficiency large even if $\tau_0 > \tau_3$. This phenomenon corresponds to the observation made by Toloşi and Lengauer (2011).

Case 3: p correlated variables. We now consider p correlated variables where

$$C = \begin{pmatrix} 1 & c & \cdots & c \\ c & 1 & \cdots & c \\ \vdots & \vdots & \ddots & \vdots \\ c & c & \cdots & 1 \end{pmatrix},$$

and $\boldsymbol{\tau}^\top = (\tau_0, \ldots, \tau_0)$. In this context the α_j's are equal. The results of Case 1 can be generalized to that situation as shown by the following result:

Proposition 5. *Assume that the correlation matrix C can be decomposed into $C = (1-c)I_p + c\mathbb{1}\mathbb{1}^\top$, where I_p is the identity matrix and $\mathbb{1} = (1, \ldots, 1)^\top$. Let $\boldsymbol{\tau} = (\tau_0, \ldots, \tau_0)^\top \in \mathbb{R}^p$. Then for all $j \in \{1, \ldots, p\}$:*

$$[C^{-1}\boldsymbol{\tau}]_j = \frac{\tau_0}{1-c+pc},$$

and consequently

$$I(X_j) = 2\left(\frac{\tau_0}{1-c+pc}\right)^2. \tag{4.3.4}$$

Proof. see Appendix 4.8 □

The proposition shows that the higher the number of correlated variables is, the faster the permutation importance of the variables decreases to zero (see Fig. 4.1). It confirms the second key observation of Tolo̧si and Lengauer (2011).

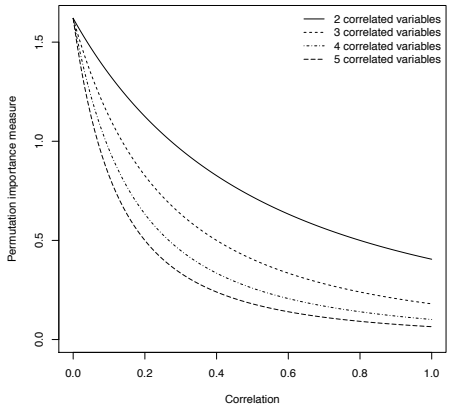


Figure 4.1 – Case 3 - Permutation importance measure (4.3.4) versus the predictor correlation for $p \in \{2, 3, 4, 5\}$ and for $\tau_0 = 0.9$.

Case 4: One group of correlated variables and one group of independent variables. Let us now assume that two independent blocks of predictors are observed. The first block corresponds to the p correlated variables $X_1, \ldots, X_p$ of Case 3 whereas the second block is composed of q independent variables $X_{p+1}, \ldots, X_{p+q}$. This case is thus a generalization of Case 2 where

$$C = \begin{pmatrix} 1 & \cdots & c & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & & \vdots \\ c & \cdots & 1 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 1 & \cdots & 0 \\ \vdots & & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & 1 \end{pmatrix},$$

and $\boldsymbol{\tau}^\top = (\tau_0, \ldots, \tau_0, \tau_{p+1}, \ldots, \tau_{p+q})$. It can be checked that the importance of the correlated variables is still given by (4.3.4) and that $I(X_j) = 2\tau_j^2$ for $j \in \{p+1, \ldots, p+q\}$. Again, the independent variables may show higher importance values even if they are less informative than the correlated ones.

Case 5: Anti-correlation. All the previous cases consider positive correlations between the predictors. We now look at the effect of anti-correlation on the permutation measure. Let us consider two predictors X_1 and X_2 such that $X_2 = -\rho X_1 + \varepsilon$ where X_1 and ε are independent and $\rho \in (0, 1]$. The correlation between X_1 and X_2 equals $-\rho$, assuming that the variances of X_1 and X_2 are equal to 1. The permutation importance increases when ρ grows to 1 according to Equation (4.3.2). This surprising phenomenon can be explained intuitively: if ρ is close to -1, we need both X_1 and X_2 in the model to explain Y because they vary in two opposite directions. Consequently, the random permutations of one of the two variables induces a high prediction error. Finally, the permutation importance of these two variables is high for ρ close to -1.

Permutation importance measure for classification

We close this section by giving few elements regarding permutation importance measures in the classification framework. In this context, Y takes its values in $\{0, 1\}$. The error of a rule f for predicting Y is $R(f) = \mathbb{P}[f(\mathbf{X}) \neq Y]$. The function minimizing R is the Bayes classifier defined by $f(\mathbf{x}) = \mathbb{1}_{\eta(\mathbf{x})>0.5}$, where $\eta(\mathbf{x}) = \mathbb{E}[Y|\mathbf{X} = \mathbf{x}] = \mathbb{P}[Y = 1|\mathbf{X} = \mathbf{x}]$ is the regression function. Given a classification rule $\hat{f}$, we consider its empirical error based on the learning set $\mathcal{D}_n$ and a validation sample $\bar{\mathcal{D}}$:

$$\hat{R}(\hat{f}, \bar{\mathcal{D}}) = \frac{1}{|\bar{\mathcal{D}}|} \sum_{i:(\mathbf{X}_i, Y_i) \in \bar{\mathcal{D}}} \mathbb{1}_{\hat{f}(\mathbf{X}_i) \neq Y_i}.$$

Consequently, the permutation importance measure is

$$I(X_j) = \mathbb{P}\left[Y \neq f(\mathbf{X}_{(j)})\right] - \mathbb{P}\left[Y \neq f(\mathbf{X})\right],$$

and its empirical counterpart is

$$\hat{I}(X_j) = \frac{1}{n_{tree}} \sum_{t=1}^{n_{tree}} \left[\hat{R}(\hat{f}_t, \bar{\mathcal{D}}_n^{tj}) - \hat{R}(\hat{f}_t, \bar{\mathcal{D}}_n^{t})\right],$$

as in Equation (4.2.1). We can equivalently rewrite the importance $I(X_j)$ as

$$\begin{aligned} I(X_j) &= \mathbb{E}[(Y - f(\mathbf{X}_{(j)}))^2] - \mathbb{E}[(Y - f(\mathbf{X}))^2] \\ &= \mathbb{E}[(Y - \eta(\mathbf{X}_{(j)}))^2] - \mathbb{E}[(Y - \eta(\mathbf{X}))^2] \end{aligned} \tag{4.3.5}$$

Of course the regression function does not satisfy the additive model (4.3.1) but we can consider alternatively the additive logistic regression model:

$$\text{Logit}(\eta(\mathbf{x})) = \sum_{j=1}^{p} f_j(X_j).$$

However, the permutation importance measure (4.3.5) cannot be easily related to the variance terms $\mathbb{V}[f_j(X_j)]$. In fact, this is possible by defining a permutation importance

measure $\tilde{I}$ on the odd ratios $\frac{\eta(\mathbf{x})}{1-\eta(\mathbf{x})}$ rather than on the regression function as follows:

$$\tilde{I}(X_j) = \mathbb{E}\left[\left(\log\frac{\eta(\mathbf{X})}{1-\eta(\mathbf{X})} - \log\frac{\eta(\mathbf{X}_{(j)})}{1-\eta(\mathbf{X}_{(j)})}\right)^2\right].$$

Indeed, straightforward calculations show that

$$\tilde{I}(X_j) = 2\mathbb{V}[f_j(X_j)].$$

Roughly, the permutation of X_j has an impact in I only if the permutations change the predicted class (for instance when the odd ratios are close to 1). In contrast the perturbation of the odd ratio due to a permutation of X_j in $\tilde{I}$ is taken into account in $\tilde{I}$ whatever the value of the odd ratio. Nevertheless, the calculations we propose for the regression framework can be hardly adapted in this context, essentially because $\tilde{I}$ cannot be easily expressed in function of the correlations between variables. Moreover, as explained before, $\tilde{I}$ is less relevant than I for the classification purpose.

The results of this section show that the permutation importance is strongly sensitive to the correlation between the predictors. Our results also suggest that, for backward elimination strategies, the permutation importance measure should be recomputed each time a variable is eliminated. We study this question in the next section.

4.4 Wrapper algorithms for variable selection based on importance measures

In this section we study wrapper variable selection algorithms with random forests in the context of highly correlated predictors. In the applications we have in mind, the number of predictors is large and it is then impossible to evaluate the error of the all subsets of variables. Such an exhaustive exploration is indeed ruled out by the computational cost. One solution to this issue, which has been investigated in previous studies, is to first rank the variables according to some criterion and then to explore the subsets of variables according to this ranking. Several papers follow this strategy, they differ to each other first on the way the error is computed and second on the way the permutation importance of variables is updating during the algorithm.

Choosing the error estimator is out of the scope of this paper although various methods are proposed in the literature on this issue. For instance, Díaz-Uriarte and Alvarez de Andrés (2006) and Genuer et al. (2010) use the out-of-bag (OOB) error estimate whereas Jiang et al. (2004) use both the OOB and a validation set to estimate the error. Finally, in order to avoid the selection bias described in Ambroise and McLachlan (2002), Svetnik et al. (2004) use an external 5-fold cross-validation procedure: they produce several variable selections on the 5 subsets of the training data and compute the averaged CV errors. In the sequel, the algorithms are performed by computing two kinds of errors : (i) OOB error which is widely used but often too optimistic as discussed in Breiman (2001), (ii) validation set error which is more suitable but can not be considered in all practical situations.

We focus on the way the permutation importance measure is used in the algorithms. The first approach consists in computing the permutation importance only at the initialization of the algorithm and then to follow a backward strategy according to this "static" ranking. The method can be summarized as follows:

1. Rank the variables using the permutation importance measure
2. Train a random forest
3. Eliminate the less relevant variable(s)
4. Repeat steps 2 and 3 until no further variables remain

This strategy is called Non Recursive Feature Elimination (NRFE). Svetnik et al. (2004); Díaz-Uriarte and Alvarez de Andrés (2006) have developed such backward algorithms. More elaborated algorithms based on NRFE has been proposed in the literature as in Genuer et al. (2010). Since we are interested here in the effect of updating the measure importance, we only consider here the original version of NRFE.

The second approach called Recursive Feature Elimination (RFE) is inspired by Guyon et al. (2002) for SVM. It requires an updating of the permutation importance measures at each step of the algorithm. This strategy has been implemented in Jiang et al. (2004). The RFE algorithm implemented in this paper can be summarized as follows:

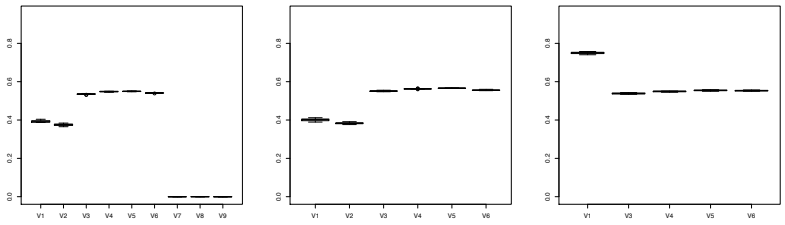
1. Train a random forest
2. Compute the permutation importance measure
3. Eliminate the less relevant variable(s)
4. Repeat steps 1 to 3 until no further variables remain

The two approaches are compared in Svetnik et al. (2004). The authors find that NRFE has better performance than RFE algorithm for their real life application. But as far as we know, no simulation studies have been carried out in the literature to confirm their observations. Moreover, this position goes against the theoretical considerations detailed above.

The results of the previous section show that the permutation importance measure of a given variable strongly depends on its correlation with the other ones and thus on the set of variables not yet eliminated in the backward process. As a consequence, RFE algorithm might be more reliable than NRFE since the ranking by the permutation importance measure is likely to change at each step. In the end, RFE algorithm can select smaller size models than NRFE since the most informative variables are well ranked in the last steps of the backward procedure even if they are correlated. In addition, by recomputing the permutation importance measure, we make sure that the ranking of the variables is consistent with their use in the current forest.

Let us consider a simple example to illustrate these ideas: we observe two correlated variables highly correlated with the outcome, four independent variables less correlated to the outcome and six irrelevant variables. More precisely, the variables are generated under the assumptions of Proposition 4: the correlation between the two relevant variables and the outcome is set to 0.7, the correlation between these variables is 0.9. The correlation between the independent variables and the outcome is 0.6. In addition, the variance of the outcome is set to 2 in order to have a positive-definite covariance matrix in the normal multivariate distribution. Figure 4.2 represents the boxplots of the permutation importance measures at several steps of the RFE algorithm. At the beginning of the algorithm, the permutation importance measure of the two first variables is lower than the independent ones (V3 to V6) even if they are more correlated to the outcome. Regarding the prediction performances of the selection procedure, one would like to select firstly one of the most informative variables (V1 or V2 in our example). At the last steps of the

algorithm (Fig. 4.2c), one of the most relevant variables is eliminated and there is no more correlations between the remaining variables. The permutation importance of variable V1 becomes larger than the other variables. Consequently, RFE algorithm firstly selects this variable whereas it is selected in fifth position when using NRFE according to the ranking shown in Figure 4.2a.



(a) Step 1 (b) Step 4 (c) Step 5

Figure 4.2 – RFE algorithm step by step with six relevant variables (two correlated and four independent) and three irrelevant variables.

In real life applications we usually need to find small size models with good performances in prediction. Thus, it is of first importance to efficiently reduce the effect of the correlations at the end of the backward procedure. By recomputing the variable importances at each step of the algorithm, the RFE algorithm manages to find small models which are efficient in term of prediction.

Remark. *Note that another importance measure has been proposed in Strobl et al. (2008) for variable ranking with random forests in the context of correlated predictors. This importance measure, called conditional importance measure, consists in permuting variables conditionally to correlated ones. This method shows good performances for a small number of predictors. However it is computationally demanding and consequently it can be hardly implemented for problems with several hundreds of predictors.*

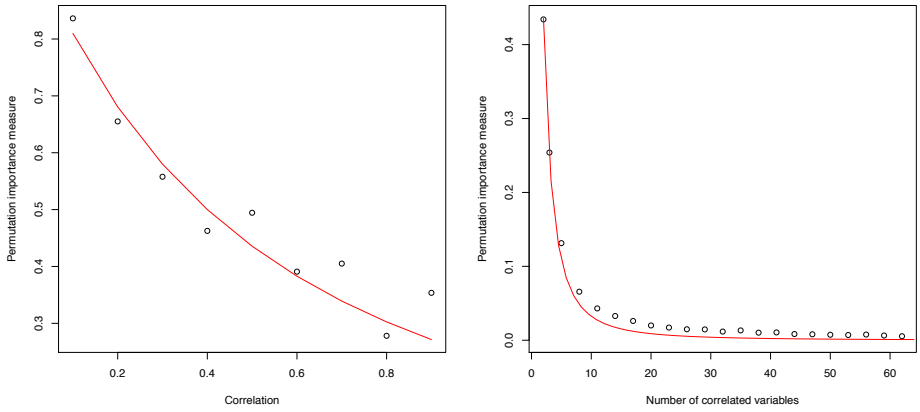
4.5 Numerical experiments

In this section, we verify with several experiments that the results proved in Section 4.3 for the permutation importance measure are also valid for its empirical version (4.2.1). RFE and NRFE approaches are compared for both classification and regression problems.

In the experiments, the number of trees in a forest is set to $n_{tree} = 1000$ and the number of variables randomly chosen for each split of the trees is set to the default value $m_{try} = \sqrt{p}$. For the NRFE algorithm, the permutation importance measure is averaged over 20 iterations as a preliminary ranking of the variables.

4.5.1 Correlation effect on the empirical permutation importance measure

This experiment is carried out under the assumptions of Proposition 4 and more precisely it corresponds to the regression problem presented in Case 1 and Case 3. Using the notations introduced in Section 4.3, the variance-covariance matrix C of the p covariates



(a) (b)

Figure 4.3 – Permutation importance measure versus the correlation (left) and the number of correlated predictors (right). The curves come from the expression of the permutation importance given in Proposition 5.

$X_1, \ldots, X_p$ has the form

$$C = \begin{pmatrix} 1 & c & \cdots & c \\ c & 1 & \cdots & c \\ \vdots & \vdots & \ddots & \vdots \\ c & c & \cdots & 1 \end{pmatrix},$$

where $c = \mathbb{C}(X_j, X_k)$, for $j \neq k$. The correlation between the X_j's and Y is denoted by τ_0. Two situations are considered. First, we take $p = 2$ (Case 1). The permutation importance measure is given by Proposition 5:

$$I(X_1) = I(X_2) = 2\left(\frac{\tau_0}{1+c}\right)^2. \tag{4.5.1}$$

Figure 4.3a represents the permutation importance measure of X_1 and its empirical counterpart versus the correlation c. The correlation τ_0 is set to 0.7 and c is varying between 0 and 1. We observe that the empirical permutation importance measure averaged over 100 simulations shares the same behaviour with the permutation importance measure (solid line in Fig. 4.3a) under predictor correlations.

Secondly, we consider p correlated predictors (Case 3) with $\tau_0 = 0.7$ and $c = 0.5$. The permutation importance is given by Proposition 5:

$$I(X_j) = 2\left(\frac{\tau_0}{1-c+pc}\right)^2. \tag{4.5.2}$$

In Figure 4.3b, the permutation importance measure and its empirical version are drawn versus the number of correlated predictors chosen among a grid between 2 and 62. We observe again that the empirical permutation importance measure fits with the permutation importance measure (solid line in Fig. 4.3b).

4.5.2 Variable selection for classification and regression problems

The RFE and NRFE algorithms are compared on several classification and regression experiments.

Experiment 1. This classification problem is inspired by Genuer et al. (2010). The procedure generates two groups of three relevant variables respectively highly, moderately and weakly discriminant and a group of irrelevant variables. The relevant variables are drawn conditionally to a realisation of the outcome Y. More precisely, the first three relevant variables are generated from the distribution $\mathcal{N}_1(Yj, 1)$ with probability 0.7 and from $\mathcal{N}_1(0, 1)$ with probability 0.3, $j \in \{1, 2, 3\}$. The variables 4 to 6 are simulated from the distribution $\mathcal{N}_1(0, 1)$ with probability 0.7 and from $\mathcal{N}_1(Y(j-3), 1)$ with probability 0.3, $j \in \{4, 5, 6\}$. The irrelevant variables are generated independently from the Gaussian distribution $\mathcal{N}_1(0, 20)$. Thus, conditionally to Y, the X_j's are drawn according to the following Gaussian mixtures densities:

$$p_{X_j}(x) = 0.7\varphi(x; Yj, 1) + 0.3\varphi(x; 0, 1),\ j \in \{1, 2, 3\},$$

$$p_{X_j}(x) = 0.7\varphi(x; 0, 1) + 0.3\varphi(x; Y(j-3), 1),\ j \in \{4, 5, 6\},$$

and

$$p_{X_j}(x) = \varphi(x; 0, 20),\ j \in \{7, \ldots, p\},$$

where $\varphi(\cdot; \mu, \sigma)$ is the normal density function with mean μ and standard error σ. We generate $n = 100$ samples and $p = 200$ variables.

Experiment 2. This classification problem is inspired by Tolo̧si and Lengauer (2011). Three groups of correlated variables are generated with a decreasing discriminative power: the variables are highly relevant in first group, they are weakly relevant in the second and irrelevant in the last group. The three groups respectively contain p_1, p_2 and p_3 variables. The simulation procedure is the following: let $\mathbf{U} = (U_1, \ldots, U_n)^\top$ be a vector of i.i.d. variables drawn according to the mixture density $\frac{1}{2}\varphi(\cdot\,; 0, 0.2) + \frac{1}{2}\varphi(\cdot\,; 1, 0.3)$. For $j \in \{1, \ldots, p_1\}$, let $\mathbf{U}^j$ a random vector defined by adding to a Gaussian noise $\mathcal{N}_1(0, 0.5)$ to 20 % of the elements of $\mathbf{U}$, the perturbated coordinates being chosen at random. Independently, some vectors $\mathbf{V}, \mathbf{V}^1, \ldots, \mathbf{V}^{p_2}$ and $\mathbf{R}$, $\mathbf{R}^1, \ldots, \mathbf{R}^{p_3}$ are drawn in the same way. Finally, the outcomes Y_i's are defined for $i \in \{1, \ldots, n\}$ by

$$Y_i = \begin{cases} 1 & \text{if } 5U_i + 4V_i - (\overline{5\mathbf{U} + 4\mathbf{V}}) + \varepsilon_i > 0 \\ 0 & \text{otherwise} \end{cases}, \tag{4.5.3}$$

where $\overline{5\mathbf{U} + 4\mathbf{V}}$ denotes the mean of the vector $5\mathbf{U} + 4\mathbf{V}$ and the ε_i's are i.i.d. random variables drawn according to $\mathcal{N}_1(0, 0.1)$. The problem considered here consists in predicting Y using as predictors all the $\mathbf{U}^j$, the $\mathbf{V}^j$ and the $\mathbf{R}^j$, but not $\mathbf{U}$, $\mathbf{V}$ and $\mathbf{R}$. Since only $\mathbf{U}$ and $\mathbf{V}$ are involved in the definition of Y, it follows that only the $\mathbf{U}^j$ and the $\mathbf{V}^j$ are relevant. Moreover, the $\mathbf{U}^j$ are more relevant than the $\mathbf{V}^j$. This model is motivated in Tolo̧si and Lengauer (2011) by a genomic application. It can be seen as slight perturbation of a linear discriminant protocol. We set the number of observations to $n = 100$, the number of variables to $p = 250$ with $p_1 = p_2 = 100$ and $p_3 = 50$.

Experiment 3. This experiment is an original classification problem based on Gaussian mixture distributions with a large number of predictors. We generate n_b groups $\mathcal{B}_1, \ldots, \mathcal{B}_{n_b}$ of correlated variables and an additional group $\mathcal{B}_{ind}$ of independent variables. Within a group $\mathcal{B}_\ell$, each variables are simulated from the mixture density

$$p_{X_j}(x) = \frac{1}{2}\varphi(x; 0, 1) + \frac{1}{2}\varphi(x; \mu_\ell, 1),\ j \in \mathcal{B}_\ell,\ \ell \in \{1, \ldots, n_b\},$$

and the variables in $\mathcal{B}_{ind}$ are simulated from

$$p_{X_j}(x) = \frac{1}{2}\varphi(x; 0, 1) + \frac{1}{2}\varphi(x; \mu_j, 1),\ j \in \mathcal{B}_{ind}.$$

The parameters μ_ℓ and μ_j give the discriminative power of the variables. We choose these mean parameters decreasing linearly from 1 to 0.5. In this way, the groups $\mathcal{B}_1, \ldots, \mathcal{B}_{n_b}$ have a decrease discriminative power which are higher than the independent group $\mathcal{B}_{ind}$. The variables in a group $\mathcal{B}_\ell$ of size q_ℓ are simulated from the multivariate Gaussian distribution $\mathcal{N}_{q_\ell}(0, C)$ with $C = (1 - c)Id + c\mathbb{1}\mathbb{1}^\top$ where c is the correlation between two different variables of $\mathcal{B}_\ell$. For the experiment, we simulate $n = 250$ samples and $p = 500$ variables: 4 blocks of 15 variables highly correlated with $c = 0.9$, one block $\mathcal{B}_{ind}$ of 10 independent variables and 430 irrelevant variables.

Experiment 4. This classification problem is inspired by Archer and Kimes (2008). We simulate $n = 100$ independent vectors $\mathbf{X}_1, \ldots, \mathbf{X}_n$ as follows: the mean vectors μ_i's are chosen uniformly distributed in $[6; 12]^p$ and the $\mathbf{X}_i$'s are drawn from the Gaussian distribution $\mathcal{N}_p(\mu_i, C)$. Each observation is composed of $L = 20$ independent blocks of $K = 40$ covariates ($p = 800$). The corresponding covariance matrix of the observed complete vector has the block-diagonal form

$$C = \begin{pmatrix} C_1 & 0 & \cdots & 0 \\ 0 & C_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & C_L \end{pmatrix}.$$

The covariance matrix C_ℓ of the ℓ-th group is

$$C_\ell = \begin{pmatrix} 1 & \rho_\ell & \cdots & \rho_\ell \\ \rho_\ell & 1 & \cdots & \rho_\ell \\ \vdots & \vdots & \ddots & \vdots \\ \rho_\ell & \rho_\ell & \cdots & 1 \end{pmatrix},$$

where the correlations are set to $\rho_\ell = 0.05\ell - 0.05$. The correlation of each block is taken from 0 to 0.95 by increments of 0.05. We then compute the probability that the observation $\mathbf{X}_i$ is from class 1 by

$$\pi_i = \frac{e^{\mathbf{X}_i^\top \beta}}{1 + e^{\mathbf{X}_i^\top \beta}},$$

for $i \in \{1, \ldots, n\}$. The regression coefficients β are sparse, that is $\beta_j = \beta_0$ if $j = (\ell - 1)K + 1$ for some $\ell \in \{1, \ldots, L\}$ and 0 otherwise. In other words, the posterior probability π_i is generated using only informations from the first variables of each group. Finally, for

$i \in \{1, \dots, n\}$ the response Y_i is generated from

$$Y_i = \begin{cases} 1 & \text{if } \pi_i < U_i, \text{ where } U_i \text{ is an uniform distribution on } [0,1] \\ 0 & \text{otherwise.} \end{cases}$$

Experiment 5. We now propose a complex linear regression design which satisfies the assumptions of Proposition 4. We simulate $n = 100$ i.i.d. copy of the random vector $(\mathbf{X}, Y)$ from the multivariate Gaussian distribution $\mathcal{N}_{p+1}\left(0, \begin{pmatrix} C & \boldsymbol{\tau} \\ \boldsymbol{\tau}^\top & 1 \end{pmatrix}\right)$ where the vector $\boldsymbol{\tau} = (\tau_0, \dots, \tau_0, 0, \dots, 0)^\top$ contains the covariances between each predictor X_j and Y, $\tau_j = \tau_0$ for the relevant variables and $\tau_j = 0$ for the irrelevant ones. We consider L groups of correlated variables and some additional irrelevant and independent variables. The matrix C has a block-diagonal form

$$C = \begin{pmatrix} C_1 & 0 & \cdots & 0 & 0 \\ 0 & C_2 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & C_L & 0 \\ 0 & 0 & \cdots & 0 & Id \end{pmatrix},$$

where

$$C_\ell = \begin{pmatrix} 1 & \rho & \cdots & \rho \\ \rho & 1 & \cdots & \rho \\ \vdots & \vdots & \ddots & \vdots \\ \rho & \rho & \cdots & 1 \end{pmatrix}.$$

This design differs from the experiment 5 since the correlations are all equal in each group. Note that the blocks can be simulated with different size to each other in order to highlight the effect of the number of correlated variables. We take 4 groups of 5 variables, 2 groups of 15 variables and 50 irrelevant variables. The correlation ρ is set to 0.9 and τ_0 is equal to 0.3 for the relevant variables and is equal to 0 for the irrelevant ones.

Variable selection

We compare the performances of the two wrapper algorithms RFE and NRFE for the five experiments described above. The model error corresponds to the misclassification rate for the classification and to the mean square error in regression case. As mentioned in Section 4.4, the error is estimated in two different ways : the out-of-bag error embedded in the random forests and the error obtained using a test set simulated independently. For each algorithm, the errors are computed in function of the number of variables in the model. The procedure is repeated 100 times to eliminate the estimation variability. We also provide for each experiment the boxplots of the initial permutation importance measures, that is the permutation importance measures used by NRFE for ranking the variables.

Table 4.1 summarizes the performances of the two variable selection approaches. The errors are given for a model which minimizes the error among the selection path induced by the backward search (Minimum error model). We also consider the errors of a parsimonious model chosen with a few number of predictors. Clearly, RFE have better prediction performances for all experiments. As expected, the differences between the RFE and the NRFE errors are higher in the parsimonious models than in the minimum error models.

		Minimum error model		Parsimonious model		
		RFE	NRFE	p^*	RFE	NRFE
OOB	Exp. 1	**0.0139**	0.0175	5	**0.0188**	0.0208
	Exp. 2	**0.0385**	0.0628	8	**0.0445**	0.1487
	Exp. 3	**0.1220**	0.1383	4	**0.1664**	0.2398
	Exp. 4	**0.1726**	0.2080	5	**0.2219**	0.2820
	Exp. 5	**0.4860**	0.4880	8	**0.5324**	0.6591
Test	Exp. 1	**0.0167**	0.0215	5	**0.0242**	0.0267
	Exp. 2	**0.0794**	0.0843	8	**0.0950**	0.1788
	Exp. 3	**0.1462**	0.1508	8	**0.1822**	0.2488
	Exp. 4	**0.3821**	0.3910	6	**0.4472**	0.4667
	Exp. 5	**0.4995**	0.5051	8	**0.5706**	0.6954

Table 4.1 – Averaged error estimates over 100 runs. The column p^* gives the number of variables in the parsimonious models we choose.

Indeed, RFE is particularly useful to reduce the effect of correlations in the last steps of the variable selection procedure and tends to select firstly the variables which are the most able to predict the outcome.

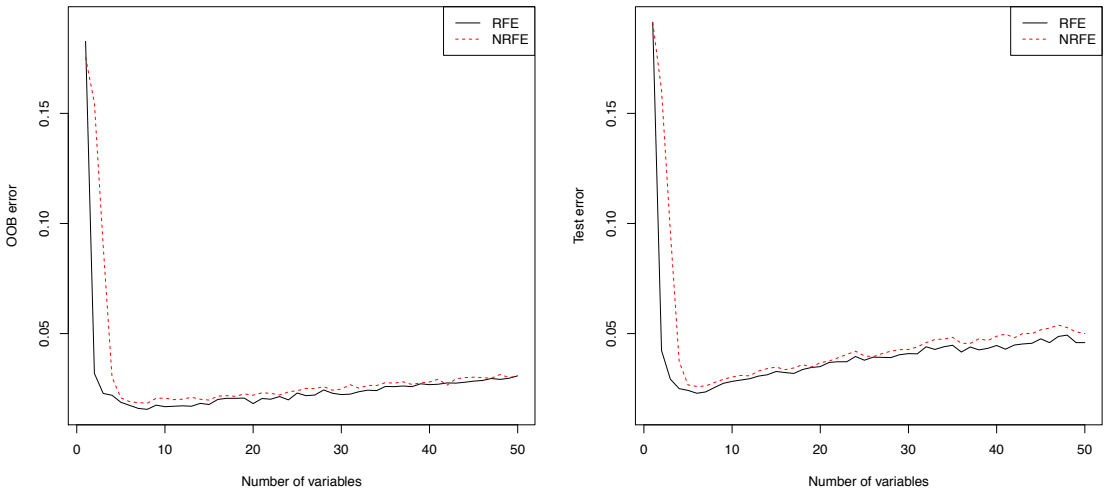
As regards Experiment 1, RFE and NRFE achieve similar performances (see Fig. 4.4). This can be explained by the fact that the design used for this experiment does not involve correlations between the predictors since all of the variables are simulated to be independent. For this reason, RFE algorithm does not bring anything more than the initial ranking obtained by the permutation importance measure computed with all the variables as shown in Figure 4.5.

Concerning Experiments 2 and 3, Figures 4.6 and 4.8 show a meaningful difference between RFE and NRFE regarding the number of involved variables. NRFE gives a minimum error close to RFE but it needs 60 variables whereas RFE procedure only needs less than 20 variables to reach the minimum error rate. Indeed, the boxplots of the initial permutation importance measures highlight the effects of the correlations (see Fig. 4.7 and Fig. 4.9).

Experiment 4 is a tricky problem in a high dimensional setting: only 100 observations of 800 variables are available. Without surprise, as illustrated in Figure 4.11, it is a difficult task to find the relevant variables from the empirical permutation importances. In particular, we cannot clearly discriminate the first variables of each group as they do not show higher importances than the other variables in the same block. In this complex simulation design related to gene expressions (Archer and Kimes; 2008), RFE and NRFE give similar performances according to the test error whereas RFE seems more efficient according to OOB error (see Fig. 4.10).

Regarding the regression problem studied in Experiment 5, RFE also outperforms NRFE according to mean squared error estimated from OOB sample and test sample (see Table 4.1). Figures 4.12 and 4.13 illustrates the fact that RFE reaches an error level close to the minimum error much faster than NRFE. Moreover, the relevant variables are well detected by the empirical permutation importance measure even if we cannot clearly identify the different groups of correlated variables (see Fig. 4.13).

These simulations confirm the well-known fact that OOB error tends to be too optimistic. We have also checked that the correlation increases the instability of the empirical permutation importance, as explained in Toloşi and Lengauer (2011); Genuer et al. (2010).



(a) (b)

Figure 4.4 – Exp. 1 - Out-of-bag error estimate (left) and test set estimate (right) versus the number of variables for RFE and NRFE algorithms. The curves are averaged over 100 runs of variable selections.

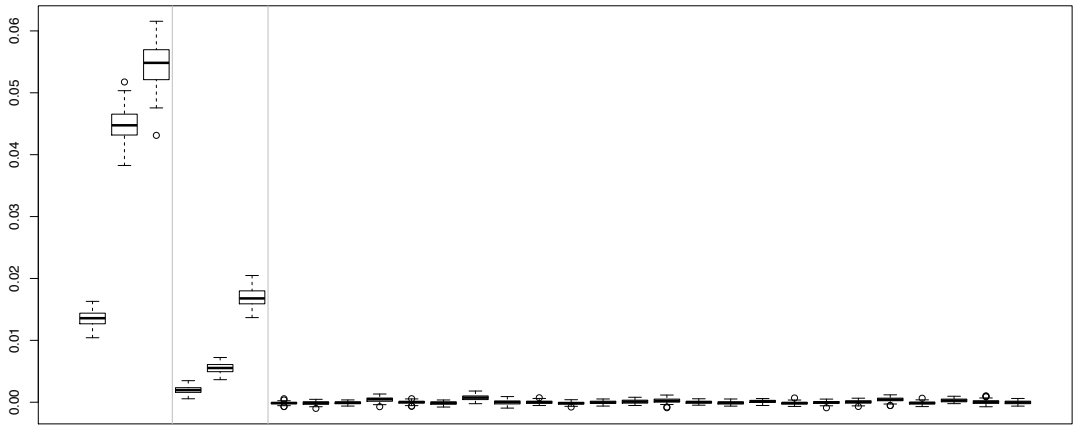
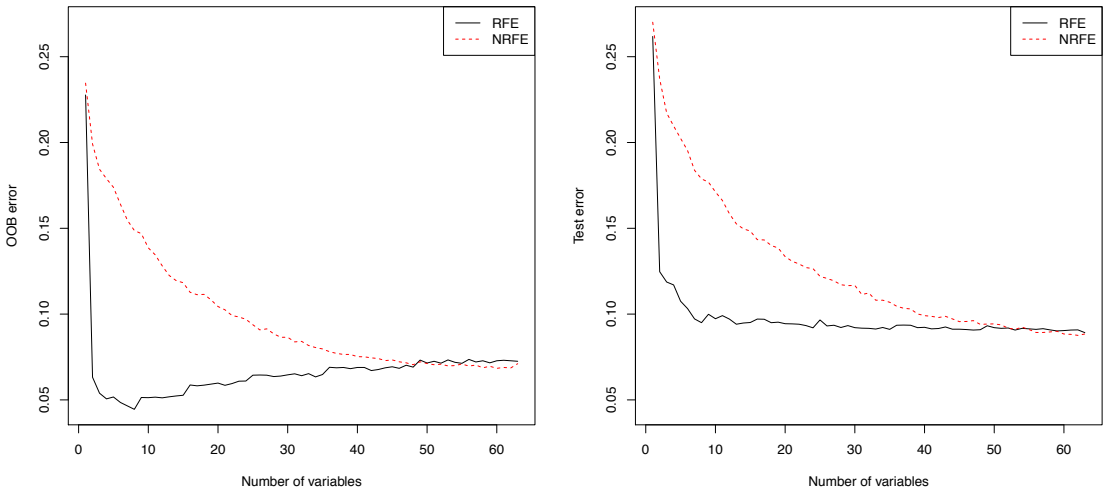


Figure 4.5 – Exp. 1 - Boxplots of the initial permutation importance measures. Only the 6 relevant variables and 24 irrelevant variables are displayed.



(a) (b)

Figure 4.6 – Exp. 2 - Out-of-bag error estimate (left) and test set estimate (right) versus the number of variables for RFE and NRFE algorithms. The curves are averaged over 100 runs of variable selections.

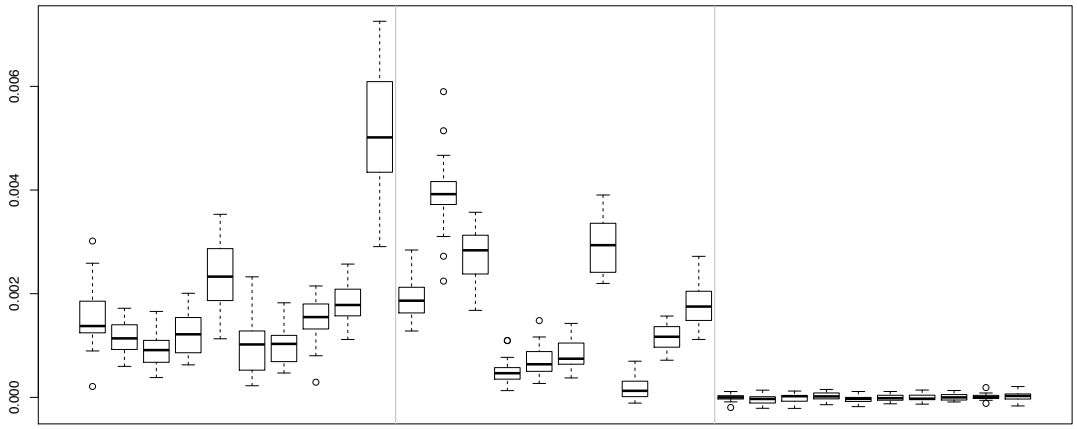
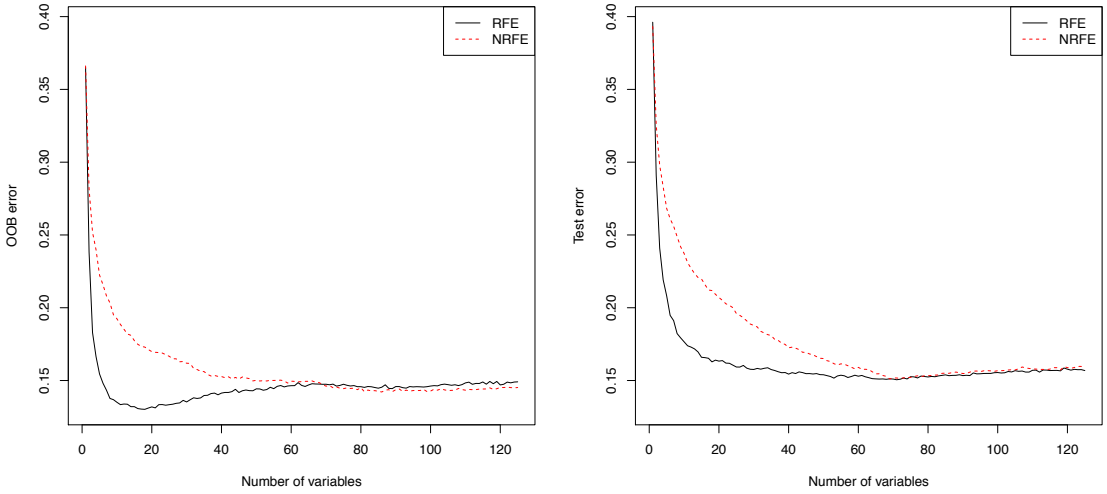


Figure 4.7 – Exp. 2 - Boxplots of the initial permutation importance measures. Only 10 variables of each group are displayed.



(a) (b)

Figure 4.8 – Exp. 3 - Out-of-bag error estimate (left) and test set estimate (right) versus the number of variables for RFE and NRFE algorithms. The curves are averaged over 100 runs of variable selections.

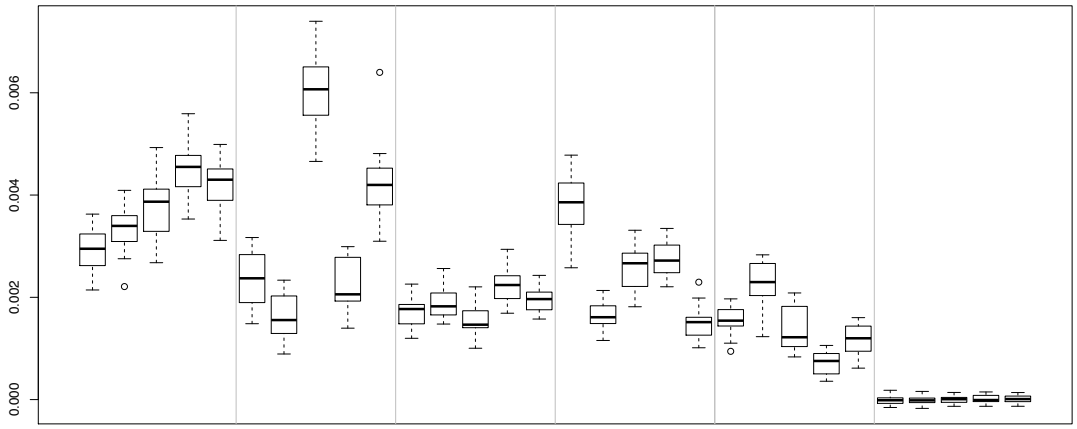
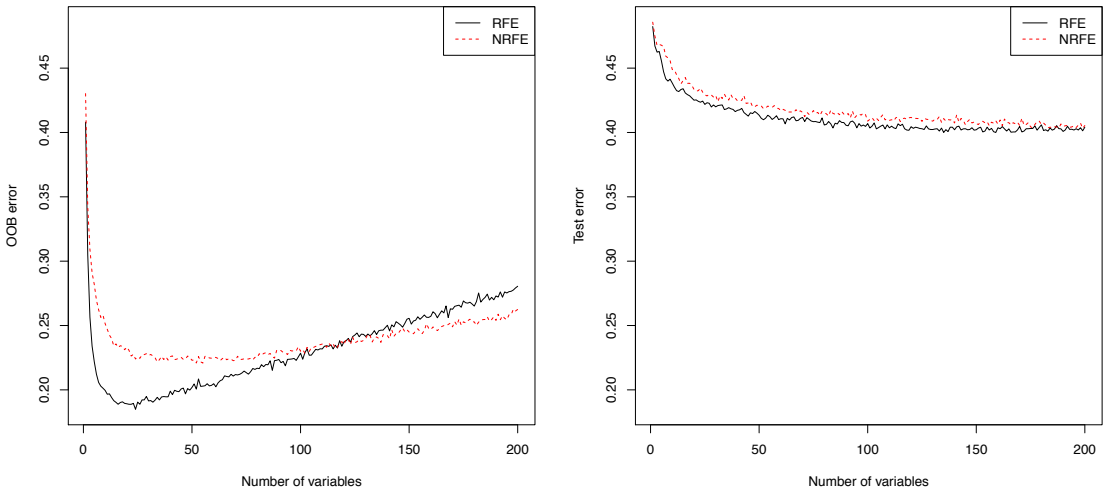


Figure 4.9 – Exp. 3 - Boxplots of the initial permutation importance measures. Only 5 variables of each group are displayed.



(a) (b)

Figure 4.10 – Exp. 4 - Out-of-bag error estimate (left) and test set estimate (right) versus the number of variables for RFE and NRFE algorithms. The curves are averaged over 100 runs of variable selections.

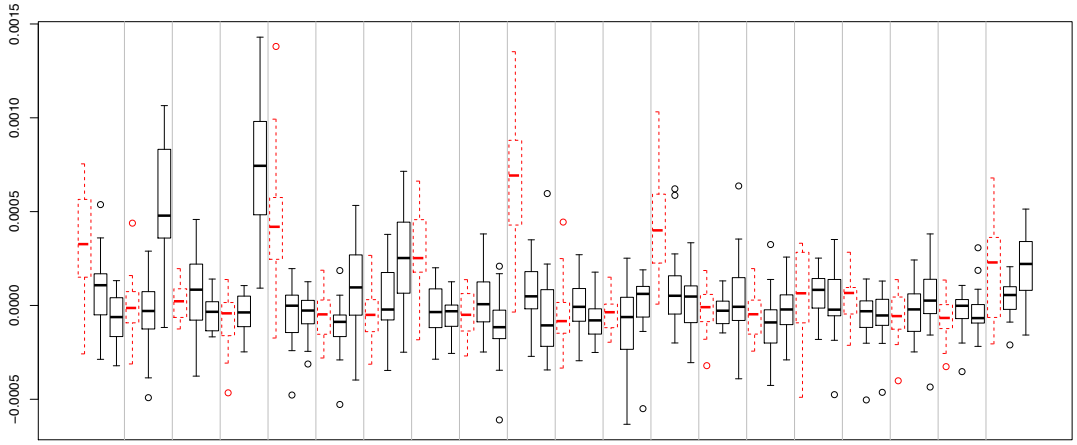
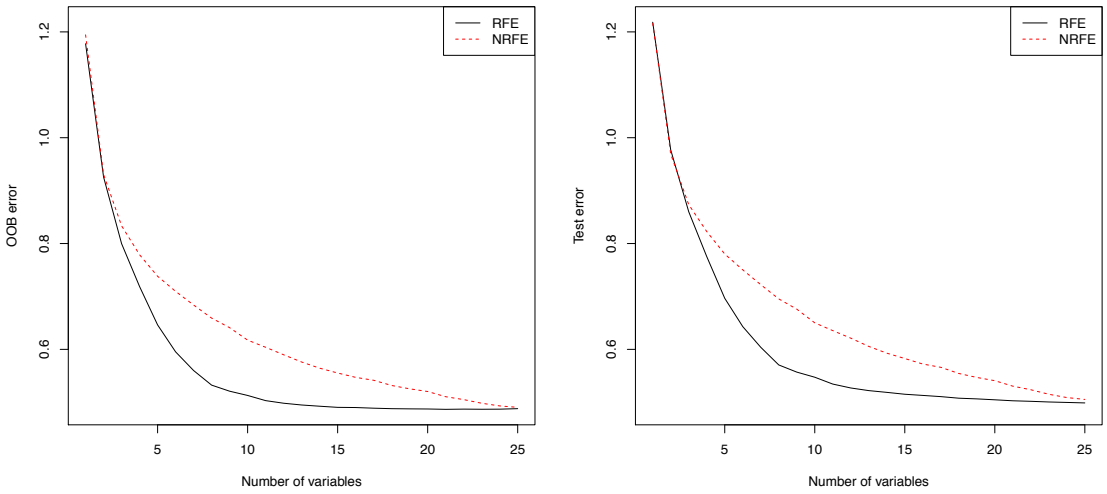


Figure 4.11 – Exp. 4 - Boxplots of the initial permutation importance measures. Only the first variables in each group (dashed lines) and two additional variables of the same group (solid lines) are displayed.



(a) (b)

Figure 4.12 – Exp. 5 - Out-of-bag MSE error (left) and test set (right) estimate versus the number of variables for RFE and NRFE algorithms. The curves are averaged over 100 runs of variable selections.

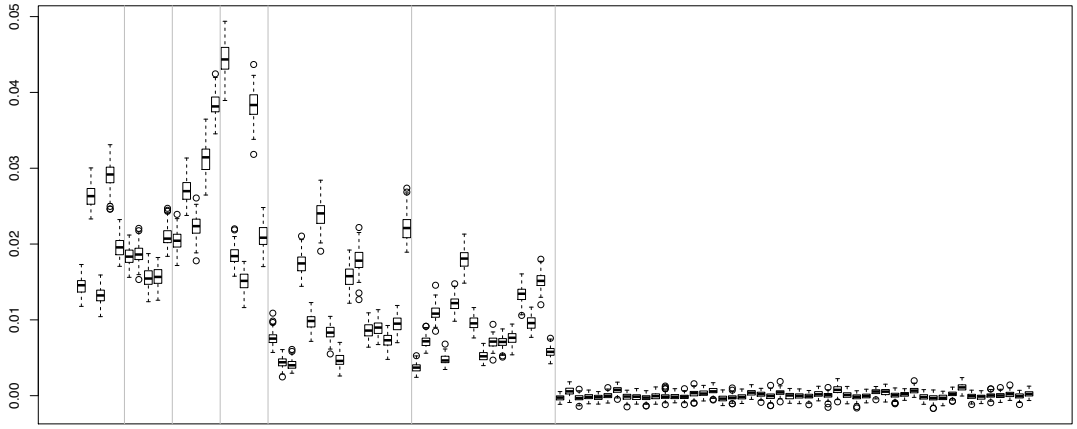


Figure 4.13 – Exp. 5 - Boxplots of the initial permutation importance measures. Four groups of 5 correlated variables and two groups of 15 correlated variables are simulated.

4.6 Application to flight data analysis

In this section the two approaches are applied to a real life problem coming from aviation safety. Airlines collect many informations during their flights using flight data recorders. Since several years, airlines have to use these data for flight safety purposes. A large number of flight parameters (up to 1000) are recorded each second as for instance the aircraft speed and accelerations, the heading, the position, several warnings. A flight provides a set of time series corresponding to these variables.

An important goal in this context is to perform variable selection in order to obtain a simple model showing good performances in term of prediction. This interpretable model for predicting a particular airline risk could be used for pilot training or for developments of new flight procedures.

We focus here on the risk of long landing. We propose to describe the risk with two levels: a flight is labelled by 1 if the landing distance exceeds 60 % of the runway length and by 0 otherwise. This naive approach does not catch the whole complexity of the phenomenon but it is better understood by safety experts.

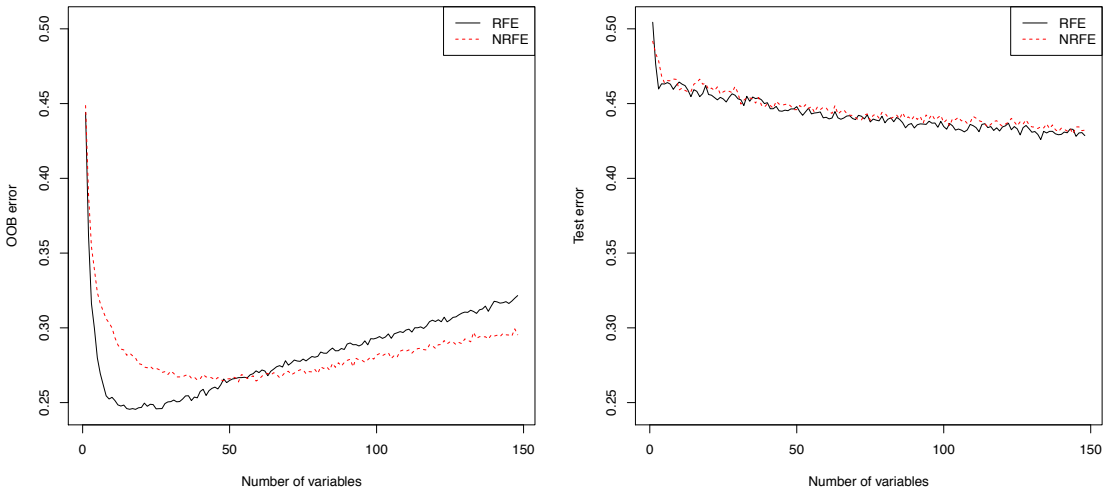
Following the recommendations of aviation experts, 22 numerical variables are preselected and the last ten minutes before touchdown are extracted from the recordings. A sample of 254 flights from the same airport and the same company is considered. The covariates obtained are highly correlated, due for instance to physical laws describing some relations between them. Some of these relations are linear but many of them are not. This problem is thus more complicated than those studied before in the paper.

The random forests algorithm can be adapted for longitudinal observations. Our approach is based on a projection of each time series in a Daubechies wavelet basis. All the variables are projected on a common wavelet basis. We consider the 8 first wavelets coefficients for each variable in order to speed up the algorithm. The obtained 176 wavelet coefficients are the input variables of the random forests algorithm.

For both RFE and NRFE, the OOB error and the classification error estimated using a test set are given in Figure 4.14. The test set is randomly chosen. It contains one third of the observations. The two algorithms provide similar results when the error is estimated using a test set (see Fig. 4.14b). As regards the OOB error, the results reinforce the simulation study. Indeed, the OOB errors for RFE algorithm are always lower for small size model (see Fig. 4.14a).

We now analyse the outputs of the RFE algorithm carried on with the test set approach. For improving the stability of the variable selection, we aggregate the procedure over 100 iterations using sub-sampling in the same spirit as the stability selection method from Meinshausen and Bühlmann (2010). By doing this, we compute the proportion of times each wavelet coefficient is selected. This procedure does not reveal the most relevant flight parameters for predicting long landings. One solution is to average for each flight parameter the selection frequencies of the eight coefficients. The resulted score provide the proportion of times at least one of the eight wavelet coefficients has been selected by the RFE algorithm. This indicator helps us to better understand the relationship between the flight parameters and the risk of long landing (see Fig. 4.15).

This procedure reveals four relevant flight parameters: the static air temperature (SAT), the altitude (ALT_STDC), the gross weight (GW_KG) and the wind speed (CASC_GSC). Figure 4.16 displays two profiles corresponding to a normal (solid lines) and an abnormal flight (dashed lines) for the variables ALT_STDC, CASC_GSC and SAT. The solid lines are consistent to normal operations. Indeed, aircraft have to level off at 4000 feet for stabilising before the final approach (see Fig. 4.16a). It is also necessary



(a) (b)

Figure 4.14 – Flight data - Out-of-bag error estimate (left) and test set estimate (right) versus the number of variables for RFE and NRFE algorithms. The curves are averaged over 100 runs of variable selections.

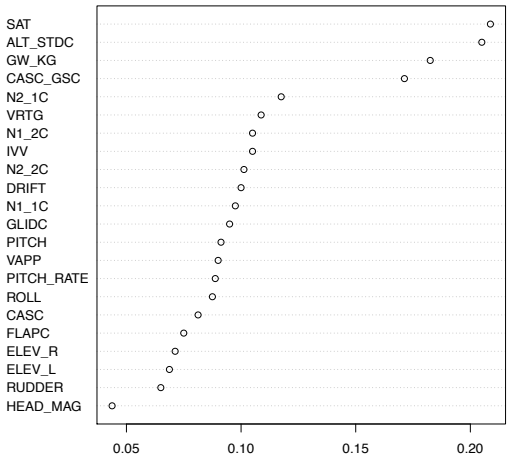
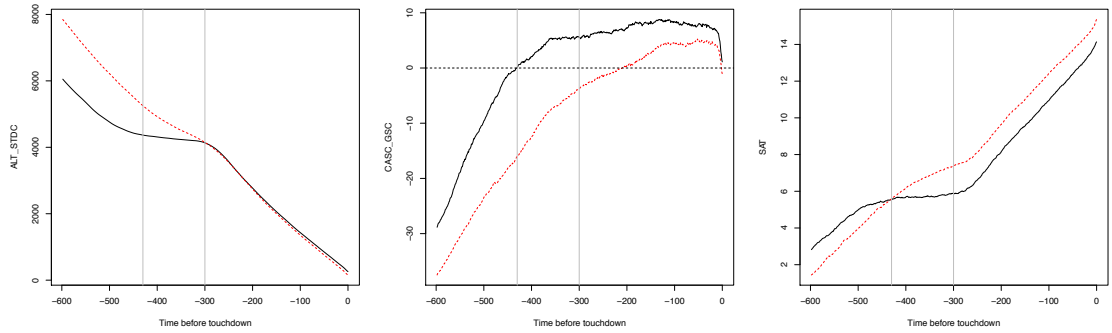


Figure 4.15 – Flight data - Selection percentage of the flight parameters



(a) Altitude (b) Wind speed (c) Static air temperature

Figure 4.16 – Flight data - Normal (solid lines) and abnormal (dashed lines) profiles for the risk of long landing. The most relevant time sequence is between the two vertical lines.

to have head wind for landing (positive values of the wind speed) in order to help the aircraft to reduce the airspeed. The abnormal profile of ALT_STDC displayed in Figure 4.16a does not level off at 4000 feet. The abnormal profile of CASC_GSC displayed in Figure 4.16b shows a tail wind which has a negative effect on the airspeed reduction. In addition, the profiles of static air temperature in Figure 4.16c reveal two different atmospheric conditions which are related to the wind speed. Finally, the analysis of the gross weight shows that the average of the weights are 165 tonnes for the abnormal flights and 160 tonnes for the regular flights. This has also a negative effect on the deceleration efficiency on the runway.

4.7 Conclusion

In this paper, we studied the problem of variable selection using the permutation importance measure from the random forests. Several simulation studies in the literature have shown an effect of the correlations between predictors on this criterion.

We first provided some theoretical insights on the effect of the correlations on the permutation importance measure. Considering an additive regression model, we obtained a very simple expression of this criterion which depends on the correlation between the covariates and on the number of correlated variables. Extending our results to a more general context is a challenging problem, this question should be investigated deeply for improving our knowledge of this widely used criterion. Moreover, the impact of the correlations on other importance measures (see van der Laan; 2006; Ishwaran; 2007) is a general question of great interest.

In a second step we focused on variable selection algorithm based on random forests analysis. A recursive and a non recursive approaches have been evaluated through an extensive simulation study on several classification and regression designs. As expected, the RFE algorithm provides better performances than the non recursive one in presence of correlated predictors: the prediction errors is always smaller with recursive strategy when small size models are selected. As a matter of fact RFE reduces the effect of the correlation on the importance measure.

In the RFE algorithm, updating the ranking is especially crucial at the last steps of the algorithm, when most of the irrelevant variables have been eliminated. In future works,

the algorithm could be adapted by combining a non recursive strategy at the first steps and a recursive strategy at the end of the algorithm.

4.8 Proofs

Proof of Proposition 3

The random variable X'_j and the vector $\mathbf{X}_{(j)}$ are defined as in Section 4.2:

$$\begin{aligned} I(X_j) &= \mathbb{E}[(Y - f(\mathbf{X}) + f(\mathbf{X}) - f(\mathbf{X}_{(j)}))^2] - \mathbb{E}[(Y - f(\mathbf{X}))^2] \\ &= \mathbb{E}[(f(\mathbf{X}) - f(\mathbf{X}_{(j)}))^2] + 2\mathbb{E}\left[\varepsilon(f(\mathbf{X}) - f(\mathbf{X}_{(j)}))\right] \\ &= \mathbb{E}[(f(\mathbf{X}) - f(\mathbf{X}_{(j)}))^2], \end{aligned}$$

since $\mathbb{E}[\varepsilon f(\mathbf{X})] = \mathbb{E}[f(\mathbf{X})\mathbb{E}[\varepsilon|\mathbf{X}]] = 0$ and $\mathbb{E}[\varepsilon f(\mathbf{X}_{(j)}] = \mathbb{E}(\varepsilon)\mathbb{E}[f(\mathbf{X}_{(j)}] = 0$. Since the model is additive, we have:

$$\begin{aligned} I(X_j) &= \mathbb{E}[(f_j(X_j) - f_j(X'_j))^2] \\ &= 2\mathbb{V}[f_j(X_j)], \end{aligned}$$

as X_j and X'_j are independent and identically distributed. For the second statement of the proposition, using the fact that $f_j(X_j)$ is centered we have:

$$\begin{aligned} \mathbb{C}[Y, f_j(X_j)] &= \mathbb{E}\left[f_j(X_j)\mathbb{E}[Y|\mathbf{X}]\right] = \mathbb{E}[f_j(X_j)\sum_{k=1}^{p} f_k(X_k)] \\ &= \mathbb{V}[f_j(X_j)] + \sum_{k\neq j} \mathbb{E}\left[f_j(X_j)f_k(X_k)\right] \\ &= \frac{I(X_j)}{2} + \sum_{k\neq j} \mathbb{C}\left[f_j(X_j), f_k(X_k)\right]. \end{aligned}$$

Proof of Proposition 4

This proposition is an application of Proposition 3 for a particular distribution. We only show that $\alpha = C^{-1}\boldsymbol{\tau}$ in that case.

Since $(\mathbf{X}, Y)$ is a normal multivariate vector, the conditional distribution of Y over $\mathbf{X}$ is also normal and the conditional mean $f(\mathbf{x}) = \mathbb{E}[Y|\mathbf{X} = \mathbf{x}]$ is a linear function: $f(\mathbf{x}) = \sum_{j=1}^{p} \alpha_j x_j$ (see for instance Rao; 1973, p. 522). Then, for any $j \in \{1, \ldots, p\}$,

$$\begin{aligned} \tau_j &= \mathbb{E}[X_j Y] \\ &= \mathbb{E}[\, X_j\mathbb{E}[Y|\mathbf{X}]\,] \\ &= \alpha_1\mathbb{E}[X_1X_j] + \cdots + \alpha_j\mathbb{E}[X_j^2] + \cdots + \alpha_p\mathbb{E}[X_pX_j] \\ &= \alpha_1 c_{1j} + \cdots + \alpha_j c_{jj} + \cdots + \alpha_p c_{pj}. \end{aligned}$$

The vector α is thus solution of the equation $\boldsymbol{\tau} = C\alpha$ and the expected result is proven since the covariance matrix C is invertible.

Proof of Proposition 5

The correlation matrix C is assumed to have the form $C = (1-c)I_p + c\mathbb{1}\mathbb{1}^\top$. We show that the invert of C can be decomposed in the same way. Let $M = aI_p + b\mathbb{1}\mathbb{1}^\top$ where a and b are real numbers to be chosen later. Then

$$\begin{aligned} CM &= ((1-c)I_p + c\mathbb{1}\mathbb{1}^\top)(aI_p + b\mathbb{1}\mathbb{1}^\top) \\ &= a(1-c)I_p + b(1-c)\mathbb{1}\mathbb{1}^\top + ac\mathbb{1}\mathbb{1}^\top + bc\mathbb{1}\mathbb{1}^\top\mathbb{1}\mathbb{1}^\top \\ &= a(1-c)I_p + (b(1-c) + ac + pbc)\mathbb{1}\mathbb{1}^\top, \end{aligned}$$

since $\mathbb{1}^\top\mathbb{1} = p$. Thus, $CM = I_d$ if and only if

$$\begin{cases} a(1-c) = 1 \\ b(1-c) + ac + pbc = 0, \end{cases}$$

which is equivalent to

$$\begin{cases} a = \dfrac{1}{(1-c)} \\ b = \dfrac{-c}{(1-c)(1-c+pc)}. \end{cases}$$

Consequently, $M^{-1}_{jk} = C^{-1}_{jk} = b$ if $j \neq k$ and $M^{-1}_{jk} = C^{-1}_{jj} = a + b$. Finally we find that for any $j \in \{1 \dots p\}$:

$$\begin{aligned} [C^{-1}\boldsymbol{\tau}]_j &= \tau_0(a+b) + \tau_0 b(p-1) \\ &= \tau_0(a+pb) \\ &= \tau_0\left(\frac{1}{(1-c)} - \frac{pc}{(1-c)(1-c+pc)}\right) \\ &= \frac{\tau_0}{1-c+pc}. \end{aligned}$$

The second point derives from Proposition 4.

Chapter 5

Mesure d'importance groupée dans les forêts aléatoires et application à l'analyse de données fonctionnelles multivariées.

Abstract. In this paper, we study the selection of grouped variables using the random forests algorithm. We first propose a new importance measure adapted for groups of variables. Theoretical insights of this criterion are given for additive regression models. The second contribution of this paper is an original method for selecting functional variables based on the grouped variable importance measure. Using a wavelet basis, we propose to regroup all of the wavelet coefficients for a given functional variable and use a wrapper selection algorithm with these groups. Various other groupings which take advantage of the frequency and time localisation of the wavelet basis are proposed. An extensive simulation study is performed to illustrate the use of the grouped importance measure in this context. The method is applied to a real life problem coming from aviation safety.

Ce chapitre fait l'objet d'un article écrit en collaboration avec Bertrand Michel et Philippe Saint Pierre, maîtres de conférence à l'Université Pierre et Marie Curie (Gregorutti et al.; 2014b).

Contents

5.1 Introduction

In high dimensional setting, the identification of the most relevant variables has been the subject of much research during the last two decades (Guyon and Elisseeff; 2003). For linear regression, the Lasso method (Tibshirani; 1996) is widely used. Many variable selection procedures have also been proposed for non linear methods. In the context of random forests (Breiman; 2001), it has been shown that the permutation importance measure introduced by Breiman, is an efficient tool for selecting variables (Díaz-Uriarte and Alvarez de Andrés; 2006; Genuer et al.; 2010; Gregorutti et al.; 2014a).

In many situations, as in medical studies and genetics, groups of variables can be clearly identified and it is of interest to select groups of variables rather than to select variables individually (He and Yu; 2010). Indeed, the interpretation of the model may be improved as well as the prediction accuracy by grouping the variables according to an a priori knowledge on the data. In the end, grouping the variables can be seen as a solution to stabilize variable selection methods. In linear settings, and more particularly for the linear regression, the group Lasso has been developed to deal with groups of variables, see for instance (Yuan and Lin; 2006b). Group variable selection have also been proposed for kernel methods (Zhang et al.; 2008) and neural networks (Chakraborty and Pal; 2008). As far as we know, this problem has not been studied for the random forests algorithm introduced by Breiman (2001). In this paper, we adapt the permutation importance measure for groups of variables in order to select groups of variables in the context of random forests.

The first contribution of this paper is a theoretical analysis of the grouped variable importance measure. Generally speaking, the grouped variable importance does not reduce to the sum of the individual importances and it can be hardly related to these last. However, in more particular models such as additive regression models, we derive exact decompositions of the grouped variable importance measure.

The second contribution of this work is an original method for selecting functional variables based on the grouped variable importance measure. Functional Data Analysis (FDA) is a field in statistics that analyzes data indexed by time (Ramsay and Silverman; 2005; Ferraty and Vieu; 2006; Ferraty; 2011). One standard approach in FDA consists in projecting the functional variables on a finite dimensional space spanned by a functional basis. Classical bases in this context are Splines, wavelets, Karhunen-Loève expansion for instance. Most of the papers about regression and classification methods for functional data consider only one functional predictor. Some references are Cardot et al. (1999, 2003); Rossi et al. (2006); Cai and Hall (2006) for linear regression methods, Amato et al. (2006); Araki et al. (2009) for logistic regression methods, Biau et al. (2005); Fromont and Tuleau (2006) for k-NN algorithms and Rossi and Villa (2006, 2008) for SVM classification. The multivariate FDA problem, where p functional variables are observed, has been less studied. Recently, Matsui and Konishi (2011); Fan and James (2013) have proposed answers to the linear regression problem with Lasso-like penalties. The logistic regression has been studied by Matsui (2014). Classification based on multivariate functional variables has

also been considered using CART algorithm (Poggi and Tuleau; 2006) and SVM (Yang et al.; 2005; Yoon and Shahabi; 2006).

We propose a new approach of multivariate FDA using random forest and the grouped variable importance measure. Indeed, various groups of basis coefficients can be proposed for a given functional decomposition. For instance, one can choose to regroup all the coefficients of a given functional variable. In this case, the selection of a group of coefficients corresponds to the selection of a functional variable. Various other groupings are proposed for a wavelet decomposition. For a given family of groups, we adapt the recursive feature elimination algorithm (Guyon et al.; 2002) which is particularly efficient when the predictors are strongly correlated (Gregorutti et al.; 2014a). In the context of random forests, this backward-like selection algorithm is guided by the grouped variable importance. Note that by regrouping the coefficients, the computational cost of the algorithm is drastically reduced compared to a backward strategy that would eliminate only one coefficient at each step.

An extensive simulation study illustrates the applications of the grouped importance measure for FDA. The method is finally applied to a real life problem coming from aviation safety. The aim of this study is to explain and predict landing distances. We select the most relevant flight parameters regarding the risk of long landings, which is a major issue for airlines. In order to speed up the algorithm, the dimension of the flight data is reduced in a preprocessing step. In Appendix 5.6, we propose a modified version of the well-known shrinkage method Donoho and Johnstone (1994) that simultaneously shrinks to zero the coefficients of the observed curves of a functional variable.

The group permutation importance measure is introduced in Section 5.2. Section 5.3 deals with multivariate FDA using random forests and the grouped variable importance measure. The application to flight data analysis is developed in Section 5.4.

5.2 Grouped variable importance measure

Let $u^\top$ denote the transpose of the vector $u \in \mathbb{R}^p$. Let Y be a random variable in $\mathbb{R}$ and $\mathbf{X} = (X_1, \ldots, X_p)$ a random vector in $\mathbb{R}^p$. We denote by $f(\mathbf{x}) = \mathbb{E}[Y|\mathbf{X} = \mathbf{x}]$ the regression function. Let $\mathrm{Var}(\mathbf{X})$ and $\mathrm{Cov}(\mathbf{X})$ denote the variance and the variance-covariance matrix of $\mathbf{X}$.

5.2.1 Importance measure of a group of variables

The permutation importance introduced by Breiman (2001) measures the accuracy of each variable X_j for predicting Y. It is based on the elementary property that the quadratic risk $\mathbb{E}\left[(Y - f(\mathbf{X}))^2\right]$ is the minimum error for predicting Y knowing $\mathbf{X}$. The formal definition of the variable importance measure of X_j is:

$$\mathcal{I}(X_j) := \mathbb{E}\left[\left(Y - f(\mathbf{X}_{(j)})\right)^2\right] - \mathbb{E}\left[(Y - f(\mathbf{X}))^2\right], \tag{5.2.1}$$

where $\mathbf{X}_{(j)} = (X_1, \ldots, X'_j, \ldots, X_p)^\top$ is a random vector such that X'_j is an independent replication of X_j which is also independent of Y and of all of the other predictors. Such criterion evaluates the increase of the prediction error after breaking the link between the variable X_j and the outcome Y, see Zhu et al. (2012) for instance.

In this paper, we extent the permutation importance for a group of variables. Let $J = (j_1, \ldots j_k)$ be a k-tuple of increasing indexes in $\{1, \ldots, p\}$, with $k \leq p$. We define the

permutation importance of the sub-vector $\mathbf{X}_J = (X_{j_1}, X_{j_2}, \ldots, X_{j_k})^\top$ of predictors by

$$\mathcal{I}(\mathbf{X}_J) := \mathbb{E}\left[\left(Y - f(\mathbf{X}_{(J)})\right)^2\right] - \mathbb{E}\left[(Y - f(\mathbf{X}))^2\right],$$

where $\mathbf{X}_{(J)} = (X_1, \ldots, X'_{j_1}, X_{j_1+1}, \ldots, X'_{j_2}, X_{j_2+1}, \ldots, X'_{j_\ell}, X_{j_\ell+1}, \ldots, X_p)^\top$ is a random vector such that $\mathbf{X}'_J = (X'_{j_1}, X'_{j_2}, \ldots, X'_{j_k})^\top$ is an independent replication of $\mathbf{X}_J$, which is also independent of Y and of all of the other predictors. We call this quantity the grouped variable importance since it only depends on which variables appear in $\mathbf{X}_J$. By abusing the notation and ignoring the ranking, we may also refer to $\mathbf{X}_J$ as a group of variables.

5.2.2 Decomposition of the grouped variable importance

Let $\mathbf{X}_J$ be a subgroup of variables from the random vector $\mathbf{X}$. Let $\mathbf{X}_{\bar{J}}$ denote the group of variables that does not appear in $\mathbf{X}_J$. Assume that we observe Y and $\mathbf{X}$ in the following additive regression model:

$$\begin{aligned} Y &= f(\mathbf{X}) + \varepsilon \\ &= f_J(\mathbf{X}_J) + f_{\bar{J}}(\mathbf{X}_{\bar{J}}) + \varepsilon, \end{aligned} \tag{5.2.2}$$

where the f_J and $f_{\bar{J}}$ are two measurable functions, and ε is a random variable such that $\mathbb{E}[\varepsilon|\mathbf{X}] = 0$ and $\mathbb{E}[\varepsilon^2|\mathbf{X}]$ is finite. The results of Gregorutti et al. (2014a) about the permutation importance of individual variables can be extended to the case of a group of variables.

Proposition 6. *Under model (5.2.2), the importance of the group J satisfies*

$$\mathcal{I}(\mathbf{X}_J) = 2\,\mathrm{Var}\left[f_J(\mathbf{X}_J)\right].$$

The proof is left to the reader, it follows the lines of Proposition 1 in Gregorutti et al. (2014a). Next Proposition gives the grouped variable importance for more specific models. It can be easily deduced from Proposition 6.

Corollary 1. *Assume that we observe Y and $\mathbf{X}$ in the model (5.2.2) with*

$$f_J(\mathbf{x}_J) = \sum_{j \in J} f_j(x_j), \tag{5.2.3}$$

where the f_j's are measurable functions and $\mathbf{x}_J = (x_j)_{j \in J}$.

1. *If the random variables $(X_j)_{j \in J}$ are independent, then*

$$\mathcal{I}(\mathbf{X}_J) = 2\sum_{j \in J} \mathrm{Var}\left(f_j(X_j)\right) = \sum_{j \in J} \mathcal{I}\left(X_j\right).$$

2. *If for any $j \in J$, f_j is a linear function such that $f_j(x_j) = \alpha_j x_j$, then*

$$\mathcal{I}(\mathbf{X}_J) = 2\alpha_J^\top \mathrm{Cov}(\mathbf{X}_J)\alpha_J, \tag{5.2.4}$$

where $\alpha_J = (\alpha_j)_{j \in J}$.

If f is additive and if the variables of the group are independent, the grouped variable importance is nothing more than the sum of the individual importances. As shown by the

second point of Corollary 1, this property is lost as soon as the variables in the group are correlated. Section 5.5 in appendix allows us to compare the grouped variable importance with the individual importances in various models. To sum up, these experiments suggest that the grouped variable importance cannot be compared with the sum of the individual importances in general settings. This is not surprising since the grouped variable importance is a more accurate measure of the importance of a group of variables than a simple sum of the individual importances.

Corollary 1 also tells us that the importance may increase with the number of variables in the group. This remark motivates the introduction of the renormalised version of the grouped variable importance:

$$\mathcal{I}_{\text{nor}}(\mathbf{X}_J) := \frac{1}{|J|}\mathcal{I}(\mathbf{X}_J).$$

In Section 5.3, we propose a variable selection algorithm based on the grouped variable importance. This algorithm used the normalised version to take into account the size of the groups in the selection process. More generally, we will rather prefer the normalised version when comparing groups of variables of different sizes.

5.2.3 Grouped variable importances and Random Forests

Classification and regression trees are competitive techniques for estimating f. The most popular method in this field is the CART algorithm due to Breiman et al. (1984). Though efficient, tree methods are also known to be unstable insofar as a small perturbation of the training sample may change radically the predictions. For answering this issue, Breiman (2001) introduced the random forests as a substantial improvement of the decision trees. The permutation importance measure was also introduced in this seminal paper. We now recall how individual permutation importances can be estimated with random forests before giving the natural extension to the estimation of grouped variable importances.

Assume that we observe n i.i.d. replications $\mathcal{D}_n = \{(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)\}$ of $(\mathbf{X}, Y)$. Random forests algorithm consists in aggregating a collection of random trees, in the same way as the bagging method also proposed by Breiman (1996): the trees are built over M bootstrap samples $\mathcal{D}_n^1, \ldots, \mathcal{D}_n^M$ of the training data $\mathcal{D}_n$. Instead of CART algorithm, a subset of variables is randomly chosen for the splitting rule at each node. Each tree is then fully grown or until each node is pure. The trees are not pruned. The resulting learning rule is the aggregation of all of the tree-based estimators denoted by $\hat{f}_1, \ldots, \hat{f}_M$. In the regression setting, the aggregation is based on the average of the predictions.

For any $m \in \{1, \ldots, M\}$, let $\bar{\mathcal{D}}_n^m := \mathcal{D}_n \setminus \mathcal{D}_n^m$ be the corresponding out-of-bag sample. The risk of $\hat{f}_m$ is estimated on the out-of-bag sample as follows:

$$\hat{R}(\hat{f}_m, \bar{\mathcal{D}}_n^m) = \frac{1}{|\bar{\mathcal{D}}_n^m|} \sum_{i:(\mathbf{X}_i, Y_i) \in \bar{\mathcal{D}}_n^m} (Y_i - \hat{f}_m(\mathbf{X}_i))^2.$$

Let $\bar{\mathcal{D}}_n^{mj}$ be the permuted version of $\bar{\mathcal{D}}_n^m$ obtained by randomly permuting the variable X_j in each out-of-bag sample $\bar{\mathcal{D}}_n^m$. The estimation of the permutation importance measure of the variable X_j is finally obtained by

$$\hat{\mathcal{I}}(X_j) = \frac{1}{M} \sum_{m=1}^{M} \left[\hat{R}(\hat{f}_m, \bar{\mathcal{D}}_n^{mj}) - \hat{R}(\hat{f}_m, \bar{\mathcal{D}}_n^m) \right].$$

This random permutation mimics the replacement of X_j by X'_j in (5.2.1) and breaks the link between X_j and Y and the other predictors.

We now extend the method for estimating the permutation importance of a group of variables $\mathbf{X}_J$. For any $m \in \{1, \ldots, M\}$, let $\bar{\mathcal{D}}_n^{mJ}$ be the permuted version of $\bar{\mathcal{D}}_n^m$ obtained by randomly permuting the group $\mathbf{X}_J$ in each out-of-bag sample $\bar{\mathcal{D}}_n^{mj}$. Note that the same random permutation is used for each variable X_j of the group. By this way the (empirical) joint law of $\mathbf{X}_J$ is left unchanged by the permutation whereas the link between $\mathbf{X}_J$ and Y and the other predictors is broken. The importance of $\mathbf{X}_J$ can be estimated by

$$\hat{\mathcal{I}}(\mathbf{X}_J) = \frac{1}{M} \sum_{m=1}^{M} \left[\hat{R}(\hat{f}_m, \bar{\mathcal{D}}_n^{mJ}) - \hat{R}(\hat{f}_m, \bar{\mathcal{D}}_n^{m}) \right].$$

Let $\hat{\mathcal{I}}_{\text{nor}}(\mathbf{X}_J)$ be the normalized version of this estimation.

In the next section, we use the grouped variable importance as a criterion for selecting features in the context of multivariate functional data.

5.3 Multivariate functional data analysis using the grouped variable importance

In this section, we consider an application of the grouped variable selection for multivariate functional regression with scalar response Y. Each covariate $X^1, \ldots, X^p$ takes its values in the Hilbert space $L^2([0,1])$ equipped with the inner product

$$\langle f, g \rangle_{L^2} = \int f(t)g(t)dt,$$

for $f, g \in L^2([0,1])$. One common approach of functional data analysis is to project the variables on a finite dimensional subspace of $L^2([0,1])$ and to use the basis coefficients in a learning algorithm.

5.3.1 Functional representation using wavelets

The wavelet transform is widely used in signal processing and for non parametric function estimation (see for instance Antoniadis et al.; 2001). Unlike Fourier basis or Splines, the wavelets are localised both in frequency and time.

For $j \geq 0$ and $k = 0, \ldots, 2^j - 1$, define a sequence of functions ϕ_{jk} (resp. ψ_{jk}), obtained by translations and dilatations of a compactly supported function ϕ (resp. ψ), called scaling function (resp. wavelet function). For any $j_0 \geq 0$, the collection

$$\mathcal{B} = \{\phi_{j_0 k}, k = 0, \ldots, 2^{j_0} - 1\} \cup \{\psi_{jk}, j \geq j_0, k = 0, \ldots, 2^j - 1\}$$

forms an orthonormal basis of $L^2([0,1])$, see for instance Percival and Walden (2000). Then a function $s \in L^2([0,1])$ can be decomposed into

$$s(t) = \sum_{k=0}^{2^{j_0}-1} \langle s, \phi_{j_0 k} \rangle_{L^2} \phi_{j_0 k}(t) + \sum_{j \geq j_0} \sum_{k=0}^{2^j - 1} \langle s, \psi_{jk} \rangle_{L^2} \psi_{jk}(t). \tag{5.3.1}$$

The first term in Equation (5.3.1) is the smooth approximation of s at level j_0 while the second term is the detail part of the wavelet representation.

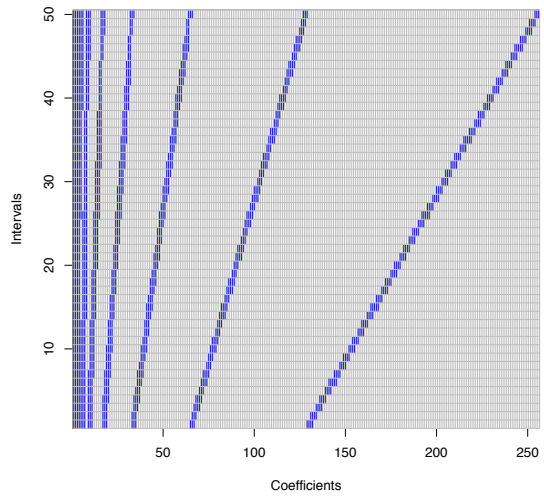


Figure 5.1 – Correspondence between the time domain and the wavelet functions for a Daubechies wavelet basis with two vanishing moments. For a time t, the colored points correspond to the wavelet functions which are not null at time t.

We assume that each covariate X is observed on a fine sampling grid $t_1, \ldots, t_N$ with $t_\ell = \frac{\ell}{N}$. A wavelet decomposition of X can be given, in a similar form as in (5.3.1). For $j_0 = 0$, we have

$$X(t_\ell) = \zeta \phi_{00}(t_\ell) + \sum_{j=0}^{J-1} \sum_{k=0}^{2^j-1} \xi_{jk} \psi_{jk}(t_\ell), \tag{5.3.2}$$

where $J := \log_2(N)$ is the maximal number of wavelet levels and ζ and ξ_{jk} are respectively the scale and the wavelet coefficients of the discretized curve X at the position k for the resolution level j. These empirical coefficients can be efficiently computed using the discrete wavelet transform algorithm described in Percival and Walden (2000, Chap. 4).

For a given wavelet basis, we introduce the *wavelet support at time* t as the set of all the indices of wavelet functions that are not null at t:

$$\mathcal{S}(t) = \{(j,k) : \psi_{jk}(t) \neq 0\}.$$

Figure 5.1 displays the matrix giving the correspondence between a time location and the associated wavelet functions, for a Daubechies wavelet basis with two vanishing moments. In a similar way but for an interval $\mathcal{T}$, we define the *wavelet support of the time interval* $\mathcal{T}$ by

$$\begin{aligned} \mathcal{S}(\mathcal{T}) &= \{(j,k) : \psi_{jk}(t) \neq 0, \forall t \in \mathcal{T}\} \\ &= \bigcap_{t \in \mathcal{T}} \mathcal{S}(t). \end{aligned}$$

This set corresponds to the wavelet functions localised on the interval $\mathcal{T}$.

5.3.2 Grouped variable importance for functional variables

In this section, we show how the grouped variable importance can be fruitfully used for comparing the importances of wavelet coefficients in the context of functional predictors. Remember that p functional covariates $X^1, \dots, X^p$ are observed together with a scalar response Y. For the sake of simplicity, the covariates are decomposed on the same wavelet basis $\mathcal{B}$ but the methodology presented above could be also adapted with a particular basis for each covariable. For any $u \in \{1, \dots, p\}$, let $\mathbf{W}^u = (\zeta^u, \xi^u_{jk})_{jk}$ be the random vector composed of the wavelet coefficients of the functional variable X^u.

Groups of wavelet coefficients

The wavelet coefficients are characterised by their frequency, their time location and the functional variables they describe. Consequently, they can be grouped in many ways. We give below a non exhaustive list of groups for which we are interested in computing the importance:

- **A group related to a variable.** The vector $\mathbf{W}^u$ defines the group $G(u)$.
- **A group related to a frequency level of a variable.** For a fixed variable X^u, the group is composed of the wavelet coefficients of frequency level j:

$$G(j, u) := \{\xi^u_{j,1}, \dots, \xi^u_{j,2^j-1}\}.$$

- **A group related to a frequency level.** The group is composed of all the wavelet coefficients of frequency level j for all the variables:

$$G(j) := \bigcup_{u=1,\dots,p} G(j, u).$$

- **A group related to a given time.** Define the group of "active" wavelet coefficients associated to a given time t by

$$G(t) := \bigcup_{u=1,\dots,p} \{\zeta^u\} \cup \bigcup_{u=1,\dots,p,\,(j,k)\in\mathcal{S}(t)} \{\xi^u_{jk}\}.$$

  Depending of the size of the support of ϕ and ψ, the group $G(t)$ may be very large. For instance with a Daubechies wavelet basis with two vanishing moments, on Figure 5.1 the group $G(t)$ is composed of the colored points of the row corresponding to time t.

- **A group related to a time interval.** Let $[a, b]$ be a time interval. The group of "active" wavelet coefficients associated to $[a, b]$ is

$$G([a, b]) := \bigcup_{t\in[a,b]} G(t).$$

Many other groupings could be proposed. For instance, one could regroup two correlated variables, or consider a group composed of the wavelet coefficients taken in a interval of frequencies, a group related to a given time and a fixed variable, etc...

By computing the importances of such groups, one directly obtain a rough detection of the most important groups of coefficients for predicting Y. When grouping by frequency

levels or by time locations, all the groups do not have equal sizes. As explained in Section 5.2, it is preferable to use the normalised version of the grouped variable importance in order to compensate the effect of group size in the grouped variable importance measure.

Grouped variable selection

We now propose a more elaborated method for selecting groups of coefficients. The selection procedure is based on the Recursive Feature Elimination (RFE) algorithm proposed by Guyon et al. (2002) in the context of support vector machines. In this paper, we propose a random forests version of the RFE algorithm which is guided by the grouped variable importance. The procedure can be summarised in Algorithm 3. This backward grouped elimination approach produces a collection of nested subsets of groups. The selected groups are obtained by minimising the validation error computed in step 2.

Algorithm 3 Grouped Variable Selection

1: Train a random forest model
2: Compute the error using a validation sample
3: Compute the grouped variable importance measure
4: Eliminate the less important group of variables
5: Repeat 1–4 until no further groups remain

This algorithm is motivated by the results from our previous work (Gregorutti et al.; 2014a) about variable selection using the permutation importance measure from the random forests. Strong correlations between predictors have a strong impact on the permutation importance measure. It was also shown in this previous paper that, when the predictors are srongly corrolated, the RFE algorithm provides better performances than the "non-recursive" strategy (NRFE) that computes the grouped variables importance just once and does not recompute the importance at each step of the algorithm. In the present paper, we continue this study by adapting the RFE algorithm for the grouped variable importance measure. We give below two applications of this algorithm.

- **Selection of functional variables.** Each vector $\mathbf{W}^u$ defines a group $G(u)$ and the goal is to perform a grouped variable selection over the groups $G(1), \ldots, G(p)$. The selection allows us to identify the most relevant functional variables.

- **Selection of the wavelet levels.** This problem is about the selection of the wavelet levels for a given functional variable. For a fixed u, we make a selection over the groups $G(j, u)$ to identify the frequency levels which yield predictive information.

Remark. *Algorithm 3 for grouped variable selection is appropriate for groups defining a partition over the wavelet coefficients. This is not the case for groups related to time locations. The algorithm can be hardly adapted with these groups because most of the wavelet coefficient belong to several groups and the elimination of a whole group might be a non efficient strategy. For instance the coefficient ζ^u, which approximates the smooth part of the curves and which is usually a good predictor, is common to all times t.*

5.3.3 Numerical experiments

In this section, we present various numerical experiments for illustrating the interest of the grouped variable importance for analysing functional data. We first describe the simulation designs.

Presentation of the general simulation design

The experiments presented below consider one or several functional covariates for predicting an outcome variable Y. Except for the second simulation of Experiment 1 which is presented in details in the next section, the functional covariates are defined in function of Y.

First, a n-sample of the outcome variable Y is simulated from a given distribution specified for each experiment. The realisation of a functional covariate X (denoted by X^u when there are several functional covariates) is a n-sample of independent discrete time random processes $X_i = (X_i(t_\ell))_{\ell=1,\dots,N}$, for all $i \in \{1,\dots,n\}$, according to a model of the form

$$X_i(t_\ell) = s(t_\ell, Z_i) + \sigma\varepsilon_{i,\ell},\ \ell = 1,\dots,N,$$

where the $\varepsilon_{i,\ell}$'s are i.i.d standard Gaussian random variables and $t_\ell = \frac{\ell}{N}$. The random variable Z_i is correlated to Y_i, it will be specified for each experiment. It is equal to the outcome variable Y_i for most of the experiments. The functional covariates are actually simulated in the wavelet domain from the following model: for $i = 1,\dots,n$, $j = 0,\dots,J-1$ and $k = 0,\dots,2^j-1$,

$$\zeta_i = \omega_0 + h_\zeta(Z_i) + \sigma\eta_{i\zeta}. \tag{5.3.3}$$

and

$$\xi_{ijk} = \begin{cases} \omega_{jk} + h_{jk}(Z_i) + \sigma\eta_{ijk} & \text{if } j \le j^\star,\ k \in \{0,\dots,2^j-1\}, \\ 0 & \text{if } j^\star < j \le J-1, \end{cases} \tag{5.3.4}$$

where $j^\star$ is the highest wavelet level of the signal. The random variables η_{ijk} and $\eta_{i\zeta}$ are i.i.d. standard Gaussian variables. The "signal" part of Equation (5.3.4) is the sum of a random coefficient ω_{jk} whose realisation is the same for all i, and a link function h_{jk}. The coefficients ω_0 and ω_{jk} in (5.3.3) and (5.3.4) are simulated as follows:

$$\begin{cases} \omega_{jk} & \sim & \mathcal{N}_1(0,\tau_j^2), & \text{if } j \le j^\star,\ k \in \{0,\dots,2^j-1\}, \\ \omega_0 & \sim & \mathcal{N}_1(3,1), \end{cases}$$

where $\tau_j = e^{-(j-1)}$. Note that the standard deviation τ_j decreases with j and thus less noise is added to the first wavelet levels. The link function h_{jk} describes the link between the wavelet coefficients ξ_{ijk} and the variable Z (or with the outcome variable Y). Two different link functions are considered in the experiments:

- a linear link $h_{jk}(z) = \theta_{jk}z$,
- a logistic link $h_{jk}(z) = \dfrac{\theta_{jk}}{1+e^{-z}}$.

where the coefficients θ_{jk} parametrise the strength of the relation between Z and the wavelet coefficients. The n discrete processes $X_1,\dots,X_i,\dots X_n$ are simulated according to Equations (5.3.3) and (5.3.4) before applying the inverse wavelet transform.

We choose a Daubechies wavelet filter with four vanishing moments to simulate the observations. We use the same basis for the projection of the functional observations.

Experiment 1: detection of important time intervals

In this first experiment, we illustrate the use of the grouped variable importance for the detection of the most relevant time intervals. We simply estimate the importance of time intervals without applying Algorithm 3 (see the remark in Section 5.3.2). We only

consider one functional covariate X since it will be sufficient to illustrate the method. Let $\mathcal{T}^\star = [t_{50}, t_{55}]$, we propose two simulation designs for which the outcome Y is correlated to the signal X on the interval $\mathcal{T}^\star$.

- **Simulation 1.** For this first simulation, we follow the general simulation design presented before by considering linear link functions h for all the wavelet coefficients belonging to the wavelet support $\mathcal{S}(\mathcal{T}^\star)$. The outcome variable Y is simulated from a Gaussian distribution $\mathcal{N}_1(0, 3)$. We simulate the wavelet coefficients as in (5.3.4) and the scaling coefficients as in (5.3.3) with $Z = Y$. We take linear link functions. We take $n = 1000$, $\sigma = 0.01$, $N = 2^8$ and thus $J = 8$. We take $j^\star = 7$ which means that even the wavelet coefficients of highest level $j = 7$ are not Dirac distributions at zero. The wavelet coefficients and the scaling coefficient are generated as follows: for any $j \in \{0, \dots, J-1\}$ and any $k \in \{0, \dots, 2^j - 1\}$,

$$\xi_{ijk} = \begin{cases} \omega_{jk} + Y_i + \sigma\eta_{ijk}, & \text{if } (j,k) \in \mathcal{S}(\mathcal{T}^\star) \\ \omega_{jk} + \sigma\eta_{ijk}, & \text{otherwise,} \end{cases} \tag{5.3.5}$$

  and

$$\zeta_i = \omega_0 + Y_i + \sigma\eta_{i\zeta}, \tag{5.3.6}$$

  for $i = 1, \dots, n$.

- **Simulation 2.** Contrary to the previous simulation, we first simulate the functional variable X and then we simulate the outcome variable Y in function of X. The functional variable is simulated in the wavelet domain according to Equations (5.3.3) and (5.3.4) with $h_{jk} = h_\zeta = 0$ for all j, k. We also take $n = 1000$, $\sigma = 0.01$, $N = 2^8$ and $j^\star = 7$. By applying the wavelet inverse transform for any i, we obtain an n-sample of discrete time random processes $X_i = (X_i(t_\ell))_{\ell=1,\dots,N}$. Figure 5.2 displays a set of 10 of these processes.

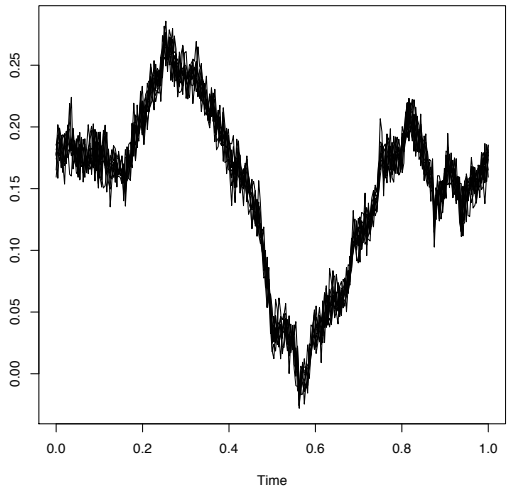


Figure 5.2 – Experiment 1 – Example of 10 processes drawn from the protocol used for Simulation 2.

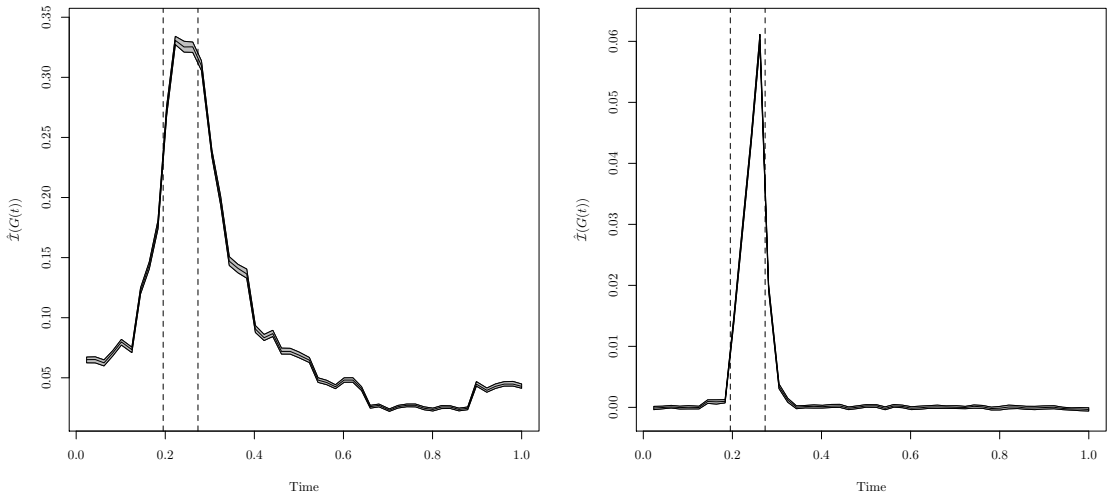
The outcome variable Y is finally obtained by the relation

$$Y_i = \frac{1000}{|\mathcal{T}^\star|} \sum_{t_\ell \in \mathcal{T}^\star} |X_i(t_\ell) - X_i(t_{\ell-1})|.$$

Thus, Y_i is a measure of the oscillations of the curve X_i over the interval $\mathcal{T}^\star$.

The aim is to detect $\mathcal{T}^\star$ using the grouped variable importance. In both cases, the grouped variable importance $\mathcal{I}(G(t))$ is evaluated at 50 equally spaced time points. Figure 5.3 displays the importance of the time points, averaged over 100 iterations. The first and third quartiles are also represented for highlighting the estimation variability. In the two cases, the importance estimation makes it possible to detect $\mathcal{T}^\star$.

Note that the detection problem is tricky in the second case because the link between Y and the wavelet coefficients is complex in this simulation. Consequently the estimated importances are low and the important intervals are difficult to detect.



(a) Case 1 (b) Case 2

Figure 5.3 – Experiment 1 – Averaged time importances, first and third quartiles over 100 iterations. The time interval $\mathcal{T}^\star$ is located between the two vertical lines.

Experiment 2: Selection of wavelet levels

This simulation is about the selection of wavelet levels for one functional variable. We follow the general simulation design presented before. The outcome variable Y is simulated from a Gaussian distribution $\mathcal{N}_1(0, 3)$. We simulate the wavelet coefficients as in (5.3.4) and the scaling coefficients as in (5.3.3) with $Z = Y$. We make two simulations.

- In the first case we use linear link functions:

$$h_\zeta(y) = 0.1y$$

and

$$h_{jk}(y) = \begin{cases} \theta_j y & \text{if } j \leq 3,\ k \in \{0, \dots, 2^j - 1\}, \\ 0 & \text{otherwise,} \end{cases}$$

where the θ_j's decrease linearly from 0.1 to 0.01.

- For the second simulation, we use logistic link functions:

$$h_\zeta(y) = \frac{0.1}{1 + e^{-y}}$$

and

$$h_{jk}(y) = \begin{cases} \dfrac{\theta_{jk}}{1 + e^{-y}} & \text{if } j \leq 3,\ k \in \{0, \dots, 2^j - 1\}, \\ 0 & \text{otherwise,} \end{cases}$$

where the θ_j's decrease linearly from 0.1 to 0.01.

We take $n = 1000$, $\sigma = 0.05$, $N = 2^8$ (thus $J = 8$) and $j^\star = 7$ in the two cases.

The aim is to identify the most relevant wavelet levels for the prediction of Y, using the grouped importance. We regroup the wavelet coefficients by wavelet levels: for $j \in \{0, \dots, J-1\}$,

$$G(j) = \left\{\xi_{jk}, k \in \{1, \dots, 2^j - 1\}\right\}$$

and

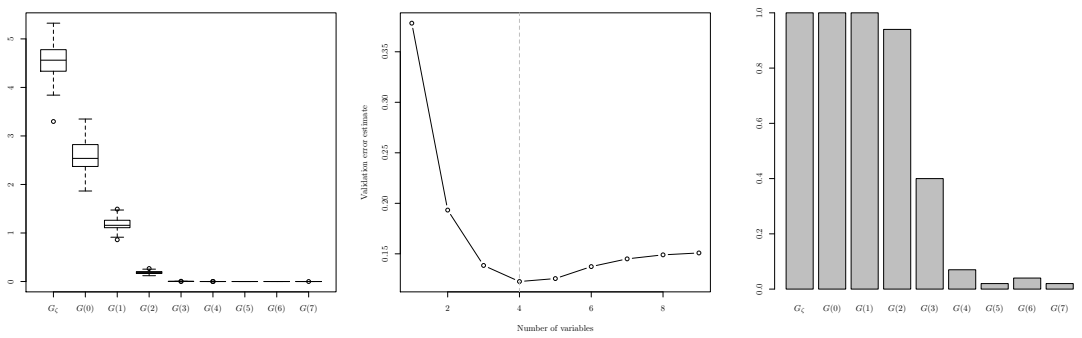
$$G_\zeta = \{\zeta\}.$$

We apply Algorithm 3 with these groups. As the group sizes are of different, the normalised grouped importance criterion given in Section 5.2.3 is used.

The experiences are both repeated 100 times. Figure 5.4 and 5.5 respectively give the results for the linear link and the logistic link. We start with the experience with linear link. The boxplots of the grouped permutation importances at the first step of the algorithm over the 100 experiences are given on Figure 5.4a. The fifth group $G(3)$ being not strongly correlated with Y, its importance is close to zero. It is selected 40 times out of the 100 simulations (Fig. 5.4c) whereas G_ζ, $G(0)$, $G(1)$ and $G(2)$ are almost always selected. The other groups are not correlated with Y and are almost never selected. For each experience, the mean squared error (MSE) is computed in function of the number of variables in the model. Figure 5.4b shows the average of the errors over the 100 simulations. On average, the model selected by minimising the MSE includes four groups but the model with five groups also has an error close to the minimum.

The experience with the logistic link gives similar results. However the fifth group $G(3)$ is more frequently selected (Fig. 5.5c). The minimisation of the MSE leads to select five groups as shown in Figure 5.5b. Note that this approach based on the random forests and grouped variable importance performs well even with a non linear link.

In both experiences, the grouped variable importances obtained at the first step of the algorithm are ranked in the same order as the θ_j's. Indeed the impact of the correlation between predictors is not too strong in the two cases. In this context, the backward Algorithm 3 does not provide additional information compared to the "non-recursive" strategy (see the disussion following Algorithm 3).



(a) (b) (c)

Figure 5.4 – Experiment 2, linear links – Selection of the wavelet levels. From the left to the right: (a) Boxplots of the grouped variable importances, (b) MSE error versus the number of groups and (c) Selection frequencies.

Experiment 3: Selection of functional variables in presence of strong correlation

This simulation illustrates the interest of Algorithm 3 for selecting functional variables in presence of correlation.

First, we simulate $n = 1000$ i.i.d. realisations of $p = 20$ functional variables $X^1, \ldots, X^p$ according to the general simulation design detailed before. For all $i \in \{1, \ldots, n\}$, let $Z_i^1, \ldots, Z_i^p$ be some latent variables drawn from a standard Gaussian distribution. The outcome variable Y is defined as:

$$Y_i = 3.5\, Z_i^1 + 3\, Z_i^2 + 2.5\, Z_i^3 + 2.5\, Z_i^4.$$

Then for $u \in \{1, \ldots, p\}$, the wavelet coefficients are simulated according to (5.3.4) and (5.3.3) with a linear link:

$$\xi_{ijk}^u = \omega_{jk}^u + Z_i^u + \sigma \eta_{ijk}^u \quad \text{if } j \leq j^\star,\ k \in \{0, \ldots, 2^j - 1\}$$

and

$$\zeta_i^u = \omega_0^u + Z_i^u + \sigma \eta_{i\zeta}^u,$$

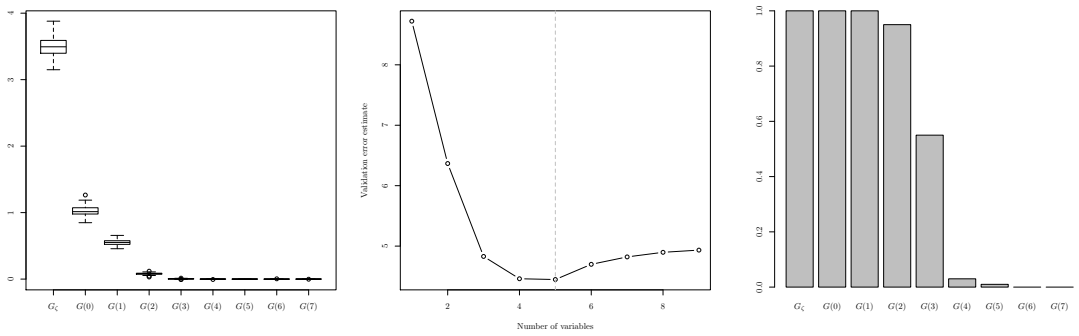
with $\sigma = 0.1$, $N = 2^9$ and thus $J = 9$. We take $j^\star = 3$ in order to make the functional variables smooth enough. Among the 20 variables X^u, the first four variables have a decreasing predictive power whereas the others are independent of Y. Next, we add $q = 10$ i.i.d. variables $X^{1,1}, \ldots, X^{1,q}$ which are strongly correlated with X^1: for any $v \in \{1, \ldots, q\}$, any $i \in \{1, \ldots, n\}$,

$$\zeta_i^{1,v} = \zeta_i^1 + \tilde{\sigma} \eta_{i\zeta}^{1,v}$$

and

$$\xi_{ijk}^{1,v} = \begin{cases} \xi_{ijk}^1 + \tilde{\sigma} \eta_{ijk}^{1,v} & \text{if } j \leq j^\star,\ k \in \{0, \ldots, 2^j - 1\} \\ 0 & \text{otherwise,} \end{cases}$$

where $\tilde{\sigma} = 0.05$. The $\eta_{ijk}^{1,v}$'s and the $\eta_{i\zeta}^{1,v}$'s are i.i.d. standard Gaussian random variables.



(a) (b) (c)

Figure 5.5 – Experiment 2, logistic links – Selection of the wavelet levels. From the left to the right: (a) Boxplots of the grouped variable importances, (b) MSE error versus the number of groups and (c) Selection frequencies.

The discrete processes $X_i^{1,v}$ are obtained using the inverse wavelet transform. In the same way, we add $q = 10$ i.i.d. variables $X^{2,1}, \ldots, X^{2,q}$ which are strongly correlated with X^2. To sum

up, the vector of predictors is composed of 40 variables:

$$X^1, X^{1,1}, \ldots, X^{1,q}, X^2, X^{2,1}, \ldots, X^{2,q}, X^3, X^4, \ldots, X^p.$$

The aim is to identify the most relevant functional variables for the prediction of Y. For each functional variable, we regroup all the wavelet coefficients and we apply Algorithm 3. The experience is repeated 100 times.

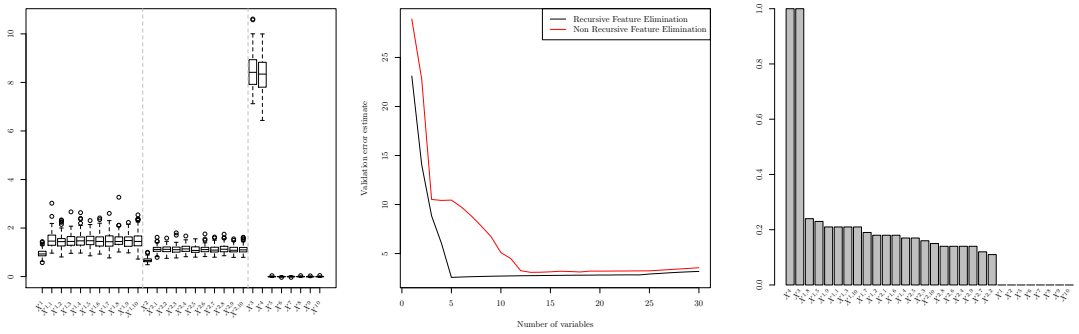
The boxplots of the group permutation importances at the first step of the algorithm, over the 100 experiences and for each functional variable, are given on Figure 5.6a. We see that the importances of the variables X^1 and X^2 and their noisy replications are much lower than the importances of the variables X^3 and X^4. This is due to the strong correlations between the two first variables and their noisy replications and it confirms the results obtained in Gregorutti et al. (2014a) for the individual importance measure. Indeed, it is shown in this last paper that the importance measure decreases when the correlation or the number of correlated variables increase. Note that the importances of X^1 and X^2 are slightly lower than the ones of their noisy replications. This can be explained by the fact that the correlation between X^1 and their noisy replications is higher than, for instance, the correlation of $X^{1,1}$ with $X^1, X^{1,2}, \ldots, X^{1,q}$.

Figure 5.6b is a comparison of the performances of Algorithm 3 and the "non-recursive" strategy (NRFE). Algorithm 3 clearly shows better prediction performances. In particular, Algorithm 3 reaches a minimum error faster than the NRFE: only five variables for Algorithm 3 whereas NRFE needs about twelve variables. This observation is consistent with the conclusion of Gregorutti et al. (2014a): the RFE procedure is more efficient than the NRFE when the predictors are highly correlated.

Additional informations are displayed in Figure 5.6c. The selection frequencies using Algorithm 3 show that the variables X^3 and X^4 are always selected. Indeed, these two variables have predictive power and they are not correlated to the other predictors. Note that the variables X^1 and X^2 are less selected then their replications, even if they are

more correlated with Y than their replications are. This also comes from the fact that the correlation between X^1 and their replications is higher than the correlation of $X^{1,1}$ with $X^1, X^{1,2}, \ldots, X^{1,q}$. We observe that X^1 and X^2 are eliminated in the first steps of the backward procedure, but this has no consequences on the prediction performances of Algorithm 3.

These results motivate the use of Algorithm 3 in practice. It reduces the effect of the correlation between predictors on the selection process and it provides better prediction performances.



(a) (b) (c)

Figure 5.6 – Experiment 3 – Selection of functional variables. From the left to the right: (a) Boxplots of the grouped variable importances, (b) MSE error versus the number of groups and (c) Selection frequencies using Algorithm 3.

5.4 A case study: variable selection for aviation safety

In this section, we study a real problem coming from aviation safety. Airlines collect many informations during their flights using flight data recorders. Since several years, airlines have to use these data for flight safety purposes. A large number of flight parameters (up to 1000) are recorded each second as for instance the aircraft speed and accelerations, the heading, the position, several warnings. One flight provides a multivariate time series corresponding to these family of functional variables.

We focus here on the risk of long landing. A sequence of $N = 512$ seconds before touchdown is observed for predicting the landing distance. The evaluation of the risk the long landings is crucial for safety managers to avoid runway excursions and more generally to keep a high level of safety. One answer to this problem is to select the flight parameters that better explain the risk of long landings. By this way, we attempt to find a sparse model showing good predictive performances. Down the road, the analysis of the flight data could be used for pilot training or for developments of new flight procedures during approach.

Following the aviation experts, 23 variables are preselected and a sample of 1868 fights from the same airport and the same company is considered. The functional variables are projected on a Daubechies wavelet basis with four vanishing moments using the discrete wavelet transform algorithm as in Section 5.3. The choice of the wavelet basis is conducted by the nature of the flight data. Indeed, the data contains informations on the time scale as well as the frequency scale and it is important to well retrieve it.

Preliminary dimension reduction

The design matrix formed by the wavelet coefficients for all of the flight parameters has dimension $23 \times 512 = 11\ 776$. Selecting the variables directly from the whole coefficients is prohibitive, we first need to reduce significantly the dimension. The naive method that shrinks the n curves independently according to Donoho and Johnstone (1994) and then brings the non-zero coefficients together in a second step would lead to consider a large block of coefficients with many zero values. This first solution is not relevant in our context. We propose an alternative method which consists in shrinking the wavelet coefficients of the n curves simultaneously. More precisely, this method is adapted from Donoho and Johnstone (1994) for the particular context of n independent (but non necessary identically distributed) discrete random processes. The shrinkage is done on the norm of the n-dimensional vector containing the wavelet coefficients. The complete method is described in Appendix 5.6.

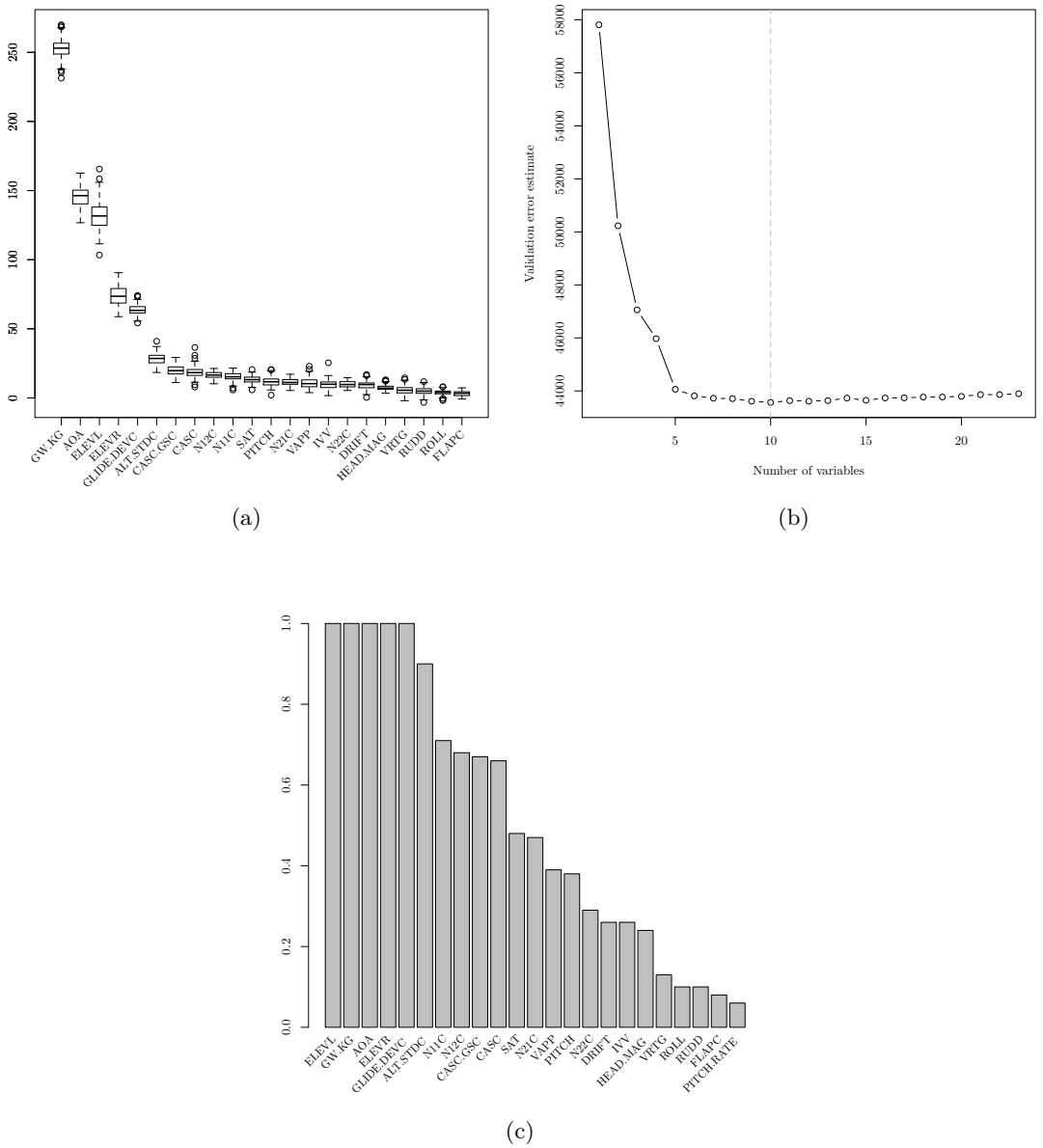
Selection of flight parameters

We obtain a selection of the functional parameters by grouping together the wavelet coefficients of each flight parameter and applying Algorithm 3 with these groups. At each iteration, we randomly split the dataset into a training set containing 90 % of the data and a validation set containing the remaining 10 %. In the backward algorithm the grouped variable importance is computed on the training set and the validation set is only used to compute the MSE errors. The selection procedure is repeated 100 times to reduce the variability of the selection. The final model is chosen by minimising the averaged prediction error. Figure 5.7a represents the boxplots of the grouped variable importance values computed on the 100 runs of the selection algorithm. According to this ranking, five variables are found significantly relevant. Looking at the averaged MSE estimate on Figure 5.7b, we see that the averaged number of selected variables is ten but taking only five variables is sufficient to get a risk close to the minimum.

Figure 5.7c gives additional informations by displaying the proportion of times each flight parameter is selected. Firstly, it confirms the previous remarks: five variables are always selected by the algorithm and the ten first are selected more than 60 times over the 100 runs. Secondly, it shows that the flight parameters related to the aircraft trajectory during the approach are among the most relevant variables for predicting the long landing. Indeed, the elevators (ELEVL, ELEVR) are used by the pilots to control the pitch of the aircraft. It has an effect on the angle of attack (AOA) and consequently on the landing. The variable GLIDE.DEVC is the glide slope deviation, that is the deviation between the aircraft trajectory and a glide path of approximatively three degrees above horizontal. It indicates how the aircraft approaches the airport. Another significant variables related to the airspeed reduction are the gross weight (GW.KG) which has an effect on the deceleration efficiency, the airspeed (CASC) and the engine rating (N11C, N12C).

It should be noted that the ranking due to the selection frequency is close to the direct ranking given by the importance measures when all the variables are taken in the model. This suggests that for this data the correlation between the predictors are not strong enough to influence their permutation importance measures. Moreover, if we regroup several variables as for example the flight parameters N11C and N12C, N21C and N22C and ELEVR and ELEVL into three new variables N1, N2 and ELEV, Figure 5.8 shows that the ranking is kept unchanged.

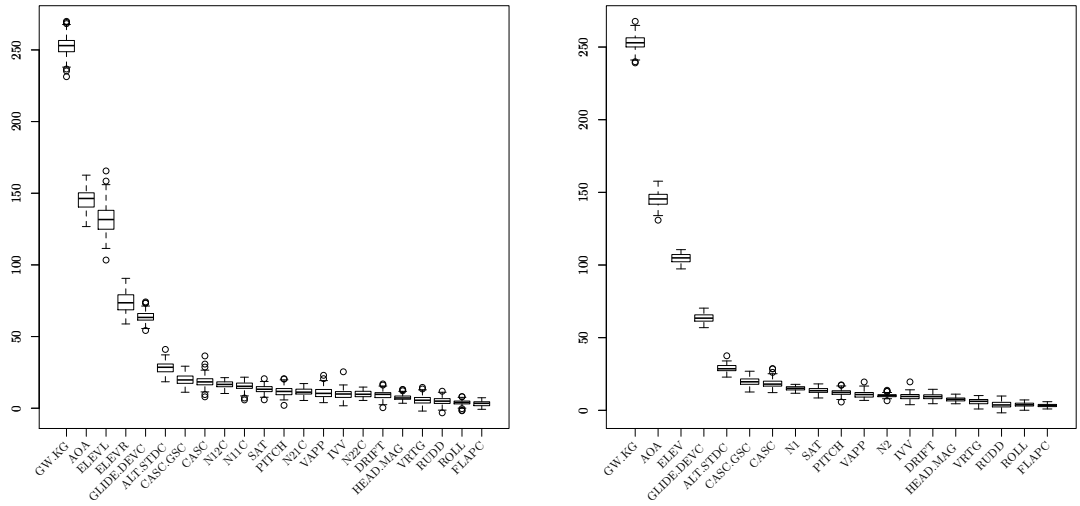
A similar study (Gregorutti et al.; 2014a) used the classification version of the Ran-



(a) (b)

(c)

Figure 5.7 – Application to long landing – From the left to the right: (a) Boxplots of the grouped variable importance (b) MSE error versus the number of groups and (c) selection frequencies.



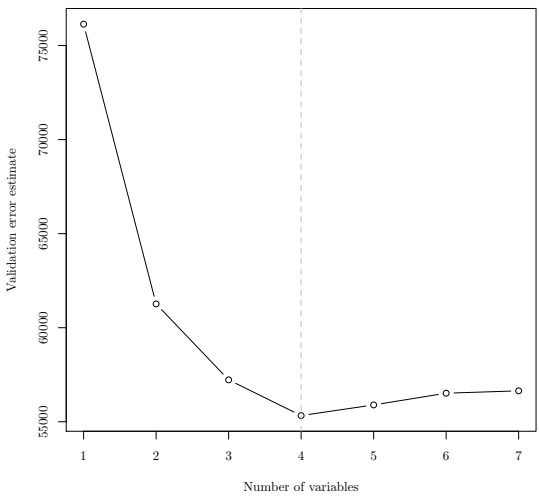
(a) (b)

Figure 5.8 – Application to long landing – Grouped variable importance measure before and after grouping the correlated flight parameters N11C and N12C, N21C and N22C, ELEVR and ELEVL into N1, N2 and ELEV.

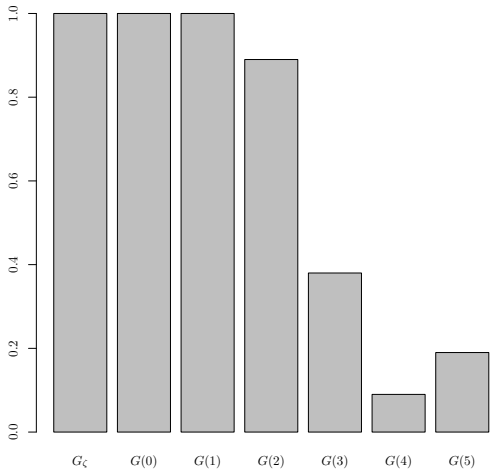
dom Forests Algorithm and (individual) permutation importance measures for variable selection. The database in Gregorutti et al. (2014a) corresponds to a set of flights from a different company and a different aircraft type. Nevertheless we found in this previous study that the gross weight, the altitude (ALT.STDC) and the wind speed (CASC.GSC) were the most relevant for predicting the landing classes. More precisely, a too high altitude combined to a tail wind was decisive during the final approach and can lead to a long landing. The two analysis are thus consistent. The variable selection procedure used for the previous study is the "non grouped" version of Algorithm 3. It consists in directly selecting the wavelet coefficients (without grouping them) and then aggregating the selected coefficients in a second step to obtain a selection of the flight parameters. Such an approach is computationally demanding. Moreover, aggregating the selection in the second step is not obvious. For these reasons, Algorithm 3 is much efficient and satisfactory for analysing this kind of multivariate functional data.

Selection of wavelet levels

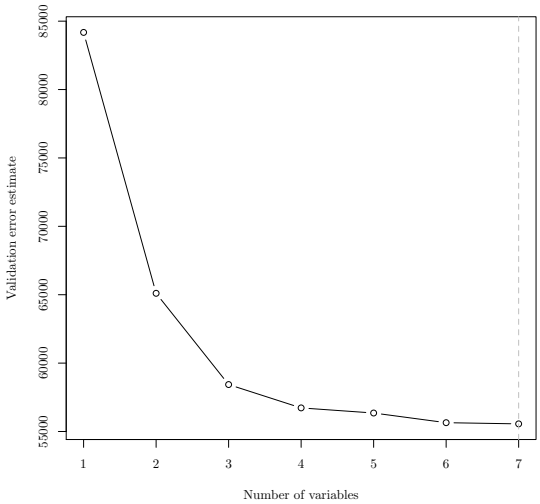
We now determine for a flight parameter which wavelet levels are the most able to predict the risk of long landing. The selection of the wavelet levels is done independently for the gross weight (GW.KG) and the angle of attack (AOA), which are among the most selected flight parameters (Fig. 5.9). Figures 5.9a and 5.9c show the averaged number of selected levels for GW.KG is less than for AOA. Indeed, the selection frequencies in Figure 5.9b indicate that for GW.KG, the first approximation levels are selected at each run (groups $\zeta, G(0)$ and $G(1)$) whereas the last levels are selected less than 40 times over 100. The situation is quite different for AOA: all the levels are selected more than 50 times over the 100 runs. The predictive power of this functional variable is shared by both the high levels of approximation and in the details of the wavelet decomposition (Fig. 5.9d).



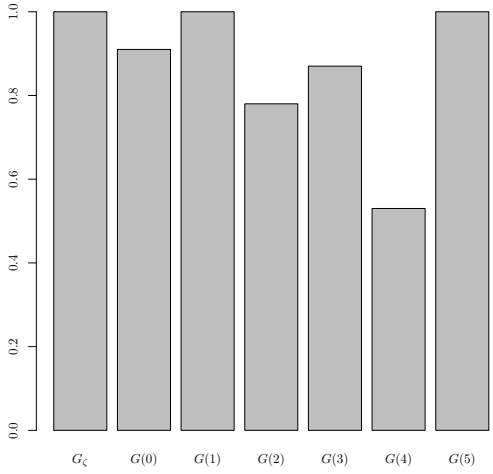
(a) Gross weight – MSE versus the number of groups



(b) Gross weight – Selection frequencies

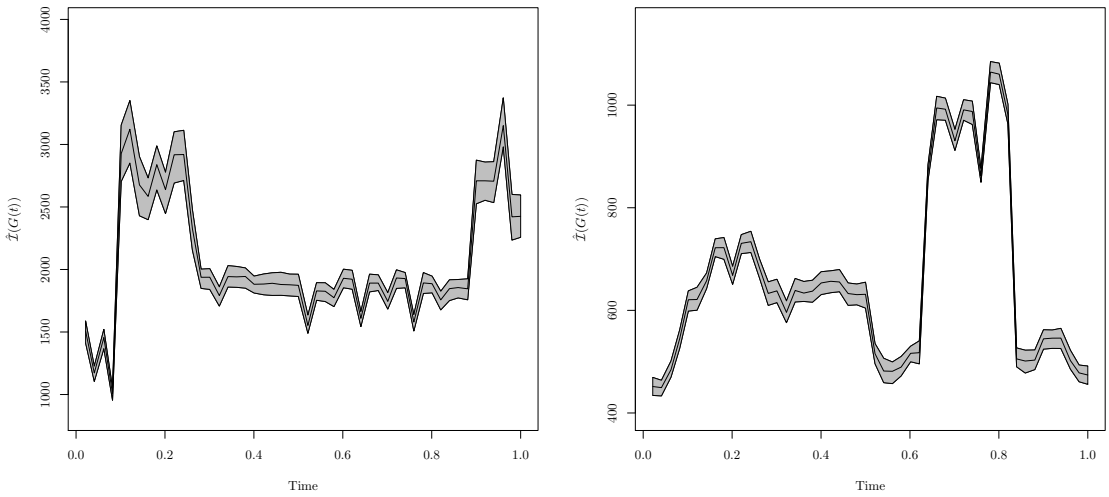


(c) Angle of attack – MSE versus the number of groups



(d) Angle of attack – Selection frequencies

Figure 5.9 – Application to long landing – Selection of the wavelet levels for the gross weight and the angle of attack.



(a) Altitude (b) Angle of attack

Figure 5.10 – Application to long landing – Averaged time importance, first and third quartiles for 100 iterations.

Detection of important time intervals

We now compute the importance of time intervals for the altitude (ALT.STDC) and for the angle of attack (AOA). Figure 5.10 displays the averaged grouped importance $G(t)$ evaluated on 50 equally spaced times points (renormalised in $[0, 1]$). The time $t = 1$ stands for the touchdown of the aircraft and $t = 0$ corresponds to 512 seconds before touchdown. Two intervals are detected with high predictive power for the altitude. These results are consistent with the view of aviation safety experts. Indeed, during the interval $[0.1, 0.25]$, the aircraft has to level off for stabilising before the final approach. A too high altitude at this moment can induce a long landing. During the interval $[0.9, 1]$, few seconds before touchdown, a too high altitude can also induce a long landing (Gregorutti et al.; 2014a).

The interval detected for the angle of attack is $[0.6, 0.8]$. This make sense because the pilots have to reduce the airspeed few seconds before touchdown.

5.5 Additional experiments about the Grouped Variable Importance

In this section, we investigate the properties of the permutation importance measure of groups of variables with numerical experiments, in addition to the theoretical results given before. In particular, we compare this quantity with a sum of individual importances in various models. We also study how this quantity behaves in "sparse situations" where only a small number of variables in the group are relevant for the prediction the outcome.

The general framework of the experiments is the following. For a fixed $p \geq 1$, let $\mathbf{X}^\top := (\mathbf{W}^\top, \mathbf{Z}^\top)$ where $\mathbf{W}$ and $\mathbf{Z}$ are two random vectors both of length p. Some of the components of $\mathbf{W}$ are correlated with Y whereas those in $\mathbf{Z}$ are all independent of Y. Let C_w be the variance-covariance matrix of $\mathbf{W}$. By incorporating the group $\mathbf{Z}$ in the model,

we present a realistic framework where not all the X_j's have a link with Y. For each experiment, we simulate $n = 1000$ samples of Y and $\mathbf{X}$ and we compute the importance $\mathcal{I}(\mathbf{W})$ of the group $\mathbf{W}$, the normalised grouped variable importance $\mathcal{I}(\mathbf{W})$ and the sum of the individual importances of the variables in $\mathbf{W}$. We repeat each experiment 500 times. The boxplots of the importances over the 500 repetitions are drawn on Figures 5.11 to 5.14 with values p between 1 and 16.

Let 0_p and I_p denote the null vector and the identity matrix of $\mathbb{R}^p$. Let $\mathbb{1}_p$ the vector of $\mathbb{R}^p$ with all coordinates equal to one and let $0_{p,q}$ denote the null matrix of dimension $p \times q$.

Experiment 1: linear link function.

We simulate $\mathbf{X}$ and Y from a multivariate Gaussian distribution. More precisely, we simulate samples from to the joint distribution

$$\begin{pmatrix} \mathbf{X} \\ Y \end{pmatrix} = \begin{pmatrix} \mathbf{W} \\ \mathbf{Z} \\ Y \end{pmatrix} \sim \mathcal{N}_{2p+1} \left(0_{2p+1} , \begin{pmatrix} C_w & 0_{p,p} & \tau \\ 0_{p,p} & I_p & 0_p \\ \tau^\top & 0_p^\top & 1 \end{pmatrix} \right)$$

where $\boldsymbol{\tau}$ is the vector of the covariances between $\mathbf{W}$ and Y. In this context, the conditional distribution of Y over $\mathbf{X}$ is normal and the conditional mean f is a linear function: $f(\mathbf{x}) = \sum_{j=1}^{p+q} \alpha_j x_j$ with $\alpha = (\alpha_1, \dots, \alpha_p, 0, \dots 0)^\top$ a sequence of deterministic coefficients (see for instance Rao (1973), p. 522 and Section 3 in Gregorutti et al. (2014a)).

- **Exp. 1a: independent predictors.** We take $\boldsymbol{\tau} = 0.9\, \mathbb{1}_p$ and $C_w = I_p$. All the variables of $\mathbf{W}$ are independent and correlated with Y.
- **Exp. 1b: correlated predictors.** We take $\boldsymbol{\tau} = 0.9\, \mathbb{1}_p$ and $C_w = (1 - 0.9) I_p + 0.9 \mathbb{1}_p \mathbb{1}_p^\top$. The variables of $\mathbf{W}$ are correlated. They are also correlated with Y.
- **Exp. 1c: independent predictors, sparse case**. We take $\boldsymbol{\tau} = (0.9, 0, \dots, 0)^\top$ and $C_w = I_p$. Only the first variable in the group $\mathbf{W}$ is correlated with Y.
- **Exp. 1d: correlated predictors, sparse case**. We take $\boldsymbol{\tau} = (0.9, 0, \dots, 0)^\top$ and $C_w = (1 - 0.9) I_p + 0.9 \mathbb{1}_p \mathbb{1}_p^\top$. The variables of $\mathbf{W}$ are correlated. Only the first variable in the group $\mathbf{W}$ is correlated with Y.

Experiment 2: additive link function.

We simulate $\mathbf{X}$ from a multivariate Gaussian distribution:

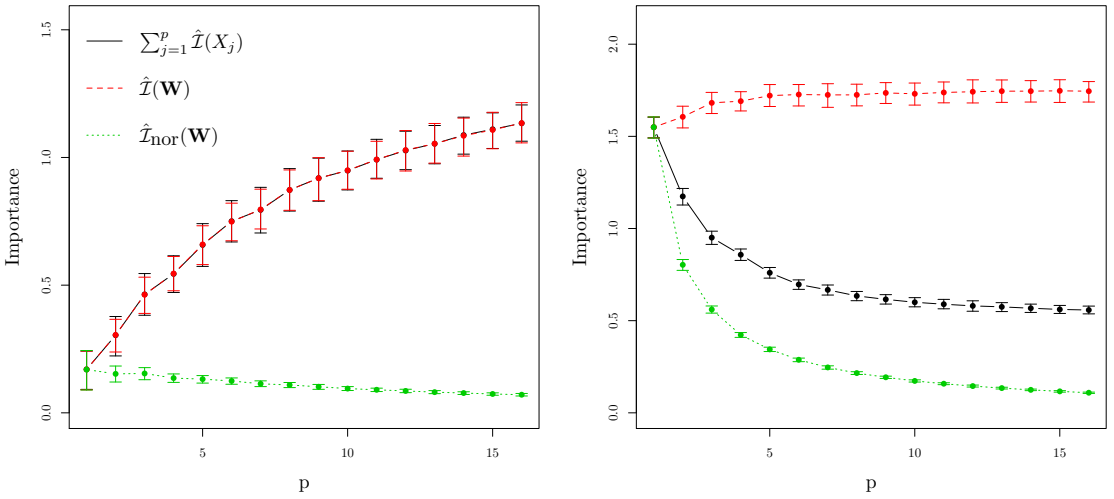
$$\mathbf{W} = \begin{pmatrix} \mathbf{W} \\ \mathbf{Z} \end{pmatrix} \sim \mathcal{N}_{2p} \left(0_{2p} , \begin{pmatrix} C_w & 0_{p,p} \\ 0_{p,p} & I_p \end{pmatrix} \right).$$

and the conditional distribution of Y is

$$(Y|\mathbf{X}) \sim \mathcal{N} \left(\sum_{j=1}^{p} f_j(X_j), 1 \right),$$

where $f_j(x) = \sin(2x) + j$ for $j < p/2$ and $f_j(x) = \cos(2x) + j$ for $j \geq p/2$.

- **Experiment 2a: independent predictors.** We take $C_w = I_p$. All the variables of $\mathbf{W}$ are independent and correlated with Y.



(a) Experiments 1a

(b) Experiments 1b

Figure 5.11 – Boxplots of the importance measures for Experiments 1a and 1b. The number of variables in $\mathbf{W}$ varies from 1 to 16. For Experiment 1a, the sum of the individual importances and $\hat{\mathcal{I}}(\mathbf{W})$ overlap.

- **Experiment 2b: correlated predictors.** We take $C_w = (1 - 0.9)I_p + 0.9\mathbb{1}_p\mathbb{1}_p^\top$. The variables of $\mathbf{W}$ are correlated and also correlated with Y.

Experiment 3: link function with interactions.

We simulate $\mathbf{X}$ from a multivariate Gaussian distribution:

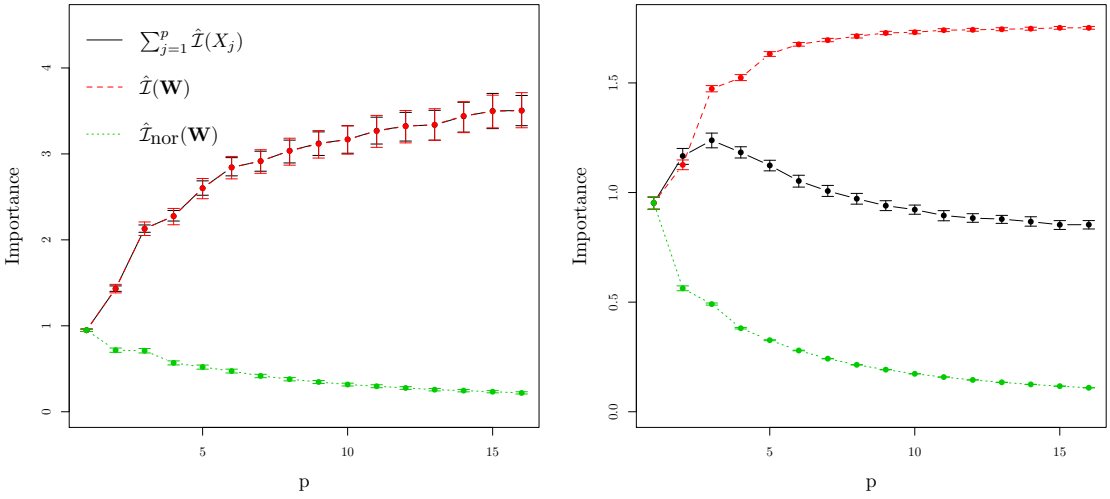
$$\mathbf{W} = \begin{pmatrix} \mathbf{W} \\ \mathbf{Z} \end{pmatrix} \sim \mathcal{N}_{2p}\left(0_{2p}\, I_{2p}\right).$$

and the conditional distribution of Y is

$$(Y|\mathbf{X}) \sim \mathcal{N}\left(\sum_{j=1}^{p} X_j + X_p X_1 + \sum_{j=1}^{p-1} X_j X_{j+1}\,,\, 1\right).$$

Results

Experiments 1a-b and 2a-b illustrate the results of Corollary 1 (see Figure 5.11 and 5.12). Indeed the regression function of both cases satisfies the additive property (5.2.3) for these experiments. In Experiments 1a and 2a, the variables of the group $\mathbf{W}$ are independent and the grouped variable importance is nothing more than the sum of the individual importances in this case. In Experiments 1b and 2b, the variables of the group $\mathbf{W}$ are positively correlated. In these situations, the grouped variable importance is larger than the sum of the individual importances, which agrees with Equation (5.2.4). Note that the grouped variable importance increases with p, which is natural because the amount of information for predicting Y increases with the group size in these models. On the other



(a) Experiments 2a

(b) Experiments 2b

Figure 5.12 – Boxplots of the importance measures for Experiments 2a and 2b. The number of variables in $\mathbf{W}$ varies from 1 to 16. For Experiment 2a, the sum of the individual importances and $\hat{\mathcal{I}}(\mathbf{W})$ overlap.

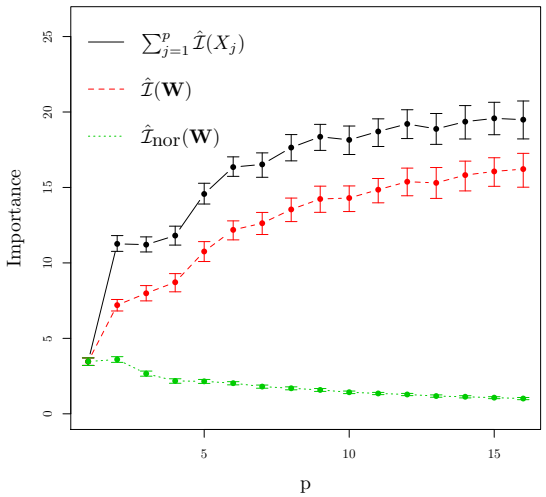
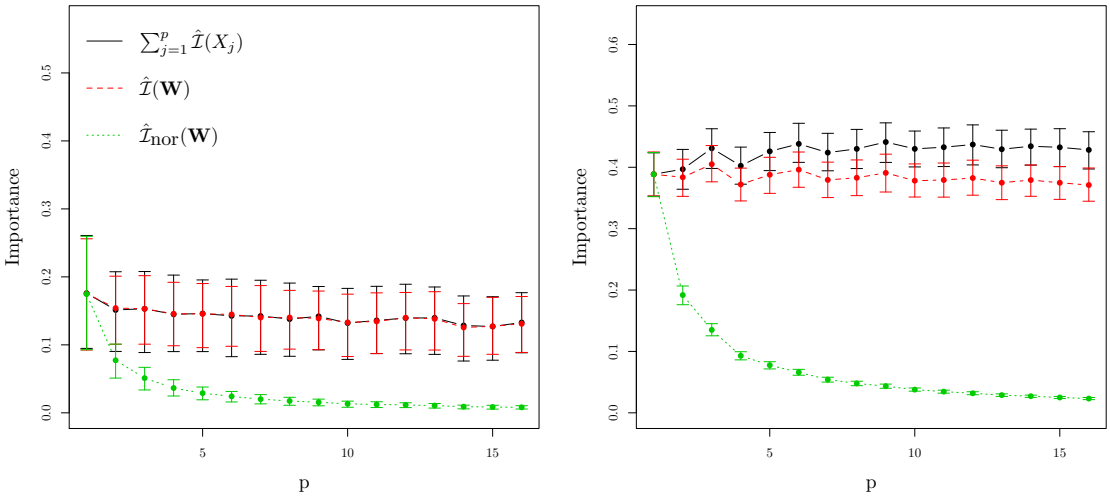


Figure 5.13 – Boxplots of the importance measures for Experiment 3. The number of variables in $\mathbf{W}$ varies from 1 to 16.



(a) Experiment 1c

(b) Experiment 1d

Figure 5.14 – Boxplots of the importance measures for Experiments 1c and 1d. The number of variables in $\mathbf{W}$ varies from 1 to 16. For Experiment 2a, the sum of the individual importances and $\hat{\mathcal{I}}(\mathbf{W})$ overlap.

hand, it was shown in Gregorutti et al. (2014a) that individual importances decrease with correlation between the predictors. Indeed we observe that the sum of the individual importances decreases with p in the correlated cases.

The regression function of Experiments 3 does not satisfy the additive form (5.2.3). Although the variables in the group are independent, the grouped variable importance is not equal to the sum of the individual importances (Figure 5.13). In a general setting, it appears that these two quantities differ.

We now comment the results of the sparse Experiments 1c and 1d (Figure 5.14). It is clear that $\mathcal{I}(\mathbf{W}) = \mathcal{I}(X_1) = \sum_{j=1\dots p} \mathcal{I}(X_j)$ for these two experiments (see Proposition 6 for instance). Regarding Experiment 1c, the boxplots of the estimated values $\hat{\mathcal{I}}(\mathbf{W})$ and $\sum_{j=1\dots p} \hat{\mathcal{I}}(X_j)$ agree with this equality. On the other hand, in Experiment 1d, we observe that $\sum_{j=1\dots p} \hat{\mathcal{I}}(X_j)$ is significantly higher than $\hat{\mathcal{I}}(\mathbf{W})$. Indeed, it has been noticed by Nicodemus et al. (2010) that the individual importances of predictors that are not associated with the outcome tend to be overestimated by the random forests, when there is correlation between predictors. In contrast, the estimator $\hat{\mathcal{I}}(\mathbf{W})$ seems to correctly estimate $\mathcal{I}(\mathbf{W})$ even for large p. Indeed, for both experiments 1c and 1d, the importance $\hat{\mathcal{I}}(\mathbf{W})$ is unchanged when the size of the group p varies from 2 to 16. Note that for variable selection, we may prefer to consider the normalized importance $\hat{\mathcal{I}}_{\text{nor}}$ to select in priority small group of variables.

5.6 Curve dimension reduction with wavelets

The analysis of flights data in Section 5.4 required, for computational reasons, to preliminary reduce the dimension of the wavelet decomposition of the flight parameters. We need to adapt the famous wavelet shrinkage method to the context of independent ran-

dom processes. Using the notations of Section 5.3.1, we first recall the hard-thresholding estimator introduced by Donoho and Johnstone (1994) in the case of one random signal. This approach is then extended to deal with n independent random signals.

5.6.1 Signal denoising via wavelet shrinkage

The problem of signal denoising can be summarised as follows. Suppose that we observe N noisy samples $X(t_1), \dots, X(t_N)$ of a deterministic function s (the signal):

$$X(t_\ell) = s(t_\ell) + \sigma\varepsilon_\ell, \quad \ell = 1, \dots, N \tag{5.6.1}$$

where the ε_ℓ's are independent standard Gaussian random variables. We assume that s belongs to $L^2([0,1])$. The goal is to recover the underlying function s from the noisy data $\{X(t_\ell), \ell \in \{1, \dots, N\}\}$ with small error. Using the discrete wavelet transform, this model can be rewritten in the wavelet domain as

$$\xi_{jk} = \omega_{jk} + \sigma\eta_{jk}, \quad \forall j \in \{0, \dots, J-1\}, \forall k \in \{0, \dots, 2^j - 1\},$$

and the scaling domain as

$$\zeta = \omega_0 + \sigma\eta_0,$$

where ξ_{jk} and ζ are the empirical wavelet and scaling coefficients of $X(t_\ell)$ as in Equation (5.3.2). The random variables η_{jk} and η_0 are i.i.d. random variables from the distribution $\mathcal{N}_1(0,1)$.

A natural approach for estimating ω_{jk} is to shrink the coefficients ξ_{jk} to zero. An estimator of s in this context has the form

$$\hat{s}(t_\ell) = \hat{\omega}_0\phi(t_\ell) + \sum_{j=0}^{J-1}\sum_{k=0}^{2^j-1}\hat{\omega}_{jk}\psi_{jk}(t_\ell)$$

with $\hat{\omega}_0 = \zeta$ and

$$\hat{\omega}_{jk} = \begin{cases} \xi_{jk} & \text{if } |\xi_{jk}| > \delta_N \\ 0 & \text{otherwise.} \end{cases}$$

This method refers as the *hard-thresholding estimator* in the literature. Donoho and Johnstone (1994) propose the universal threshold $\delta_N = \sigma\sqrt{2\log(N)}$. In addition, the standard deviation σ can be estimated by the median absolute deviation (MAD) estimate of the wavelet coefficients at the finest levels, i.e.

$$\hat{\sigma} = \frac{\text{Med}(|\xi_{jk} - \text{Med}(\xi_{jk})| : j = J-1, k = 0, \dots, 2^{J-1} - 1)}{0.6745},$$

where the normalisation factor 0.6745 comes from the normality assumption in (5.6.1). This estimator is known to be a robust and consistent estimator of σ. The underlying idea is that the variability of the wavelet coefficients is essentially concentrated at the finest level.

5.6.2 Consistent wavelet thresholding for independent random signals

This section presents a natural extension of the hard-thresholding method when n independent random processes $X_1, \dots, X_n$ are observed. The aim is to reduce the dimension of the problem by shrinking the wavelet coefficients of the n signals. One simple solution

consists in applying independently the hard-thresholding rule to each wavelet decomposition. By doing so, the n shrunk decompositions have no reason to be identical: the wavelet coefficients set to zero will be different for the n signals. In some situations (as for Section 5.4 in this paper), it is preferable to shrink the same coefficients of the n observed signals. We adapt the hard-thresholding method to answer this problem.

Identically distributed case

We start by assuming that the observations come from the same distribution: for any $i \in \{1, \dots, n\}$,

$$X_i(t_\ell) = s(t_\ell) + \sigma\varepsilon_{i,\ell}, \quad \ell = 1, \dots, N, \tag{5.6.2}$$

where the $\varepsilon_{i,\ell}$'s are independent standard Gaussian random variables. The wavelet coefficients of s can be easily deduced from the mean signal $\bar{X} := \frac{1}{n}\sum_{i=1}^n X_i$ which satisfies

$$\bar{X}(t_\ell) = s(t_\ell) + \frac{\sigma}{\sqrt{n}}\varepsilon_\ell, \quad \ell = 1, \dots, N,$$

where the ε_ℓ's are independent standard Gaussian random variables. By applying the hard-thresholding rule to this signal, we obtain the following estimation of the wavelet parameters of s: $\hat{\omega}_0 = \zeta$ and

$$\hat{\omega}_{jk} = \begin{cases} \bar{\xi}_{jk} & \text{if } |\bar{\xi}_{jk}| > \bar{\delta}_N \\ 0 & \text{otherwise,} \end{cases}$$

where $\bar{\xi}_{jk}$ is the wavelet coefficient of level (j,k) of $\bar{X}$. Here the threshold is $\bar{\delta}_N = \frac{\sigma}{\sqrt{n}}\sqrt{2\log(N)}$.

Non identically distributed case

In many real life situations, assuming that the n signals are identically distributed is not a realistic assumption. For the study presented in Section 5.4 for instance, the flight parameters have no reason to follow the same distribution in safe and unsafe conditions. We propose a generalisation of the model (5.6.2) by introducing a latent random variable Z taking it value in a set $\mathcal{Z}$. Roughly speaking, the variable Z represents all the phenomena that have an effect on the mean signal. Conditionally to $Z_i = z_i$, the distribution of the process X_i is now defined, for any $i \in \{1, \dots, n\}$, by

$$X_i(t_\ell) = s(t_\ell, z_i) + \sigma\varepsilon_{i,\ell}\,, \quad \ell = 1, \dots, N,$$

where the $\varepsilon_{i,\ell}$'s are independent standard Gaussian random variables. This regression model allows us to consider various situations of interest arising in Functional Data Analysis. In supervised settings where a variable Y has to be predicted using X, one reasonable modeling is taking $Z = Y$. We now propose an hard-thresholding method which simultaneously shrinks the wavelet decomposition of the n signals.

Let $\|\cdot\|_n$ denote the ℓ_2-norm in $\mathbb{R}^n$: $\|\boldsymbol{u}\|_n := \sqrt{\sum_{i=1}^n u_i^2}$ for any $\boldsymbol{u} \in \mathbb{R}^n$. Let $\boldsymbol{\xi}_{jk}$ be the vector $(\xi_{1jk}, \dots, \xi_{ijk}, \dots, \xi_{njk})^\top$ where ξ_{ijk} is the coefficient of level (j,k) in the wavelet decomposition of the signal X_i.

For any $z \in \mathcal{Z}$, let $\omega_{jk}(z)$ be the wavelet coefficient of level (j,k) of $s(\cdot, z)$, and $\boldsymbol{\omega}_{jk} := (\omega_{jk}(Z_1), \dots, \omega_{jk}(Z_n))^\top$. We define the common wavelet support of s by

$$L := \{(j,k)\,|\,\omega_{jk}(Z) = 0\ a.s\}\,.$$

If $(j,k) \in L$, then $\boldsymbol{\omega}_{jk} = (0,\dots,0)^\top$ almost surely and $\|\boldsymbol{\xi}_{jk}\|_n^2$ has a centered chi-square distribution with n degrees of freedom. Otherwise, $\boldsymbol{\omega}_{jk}$ can be not null and in this case $\|\boldsymbol{\xi}_{jk}\|_n^2$ has the distribution of a sum of n independent uncentered chi-square distributions. We thus propose a thresholding rule for the statistic $\|\boldsymbol{\xi}_{jk}\|_n$. For any $j \in \{0,\dots,J-1\}$ and any $k \in \{0,\dots,2^j-1\}$, let

$$\hat{\boldsymbol{\omega}}_{jk} = \begin{cases} \boldsymbol{\xi}_{jk} & \text{if } \|\boldsymbol{\xi}_{jk}\|_n > \delta_{N,n} \\ (0,\dots,0)^\top & \text{otherwise,} \end{cases}$$

where the threshold $\delta_{N,n}$ depends on N, n and σ.

Proving adaptive results in the spirit of Donoho et al. (1995) for this method is beyond the scope of the paper. However, an elementary consistent result can be proved. We would like $\hat{\boldsymbol{\omega}}_{jk}$ to be a zero vector with high probability when $(j,k) \in L$. For some $x \geq 0$, take $\delta_{N,n}^2 = \delta_{N,n}^2(x) = \sigma^2(2x + 2\sqrt{nx} + n)$, then

$$\begin{aligned} \mathbb{P}\left[\bigcup_{(j,k)\in L} \left\{\hat{\boldsymbol{\omega}}_{jk} \neq (0,\dots,0)^\top\right\}\right] &\leq \sum_{(j,k)\in L} \mathbb{P}\left[\|\boldsymbol{\xi}_{jk}\|_n^2 \geq \delta_{N,n}^2(x)\right] \\ &= \sum_{(j,k)\in L} \mathbb{P}\left[\frac{\|\boldsymbol{\xi}_{jk}\|_n^2}{\sigma^2} - n \geq 2x + 2\sqrt{nx}\right] \\ &\leq |\bar{L}|e^{-x} \leq Ne^{-x} \end{aligned} \tag{5.6.3}$$

where we have used a deviation bound for central chi-square distributions from Laurent and Massart (2000, p. 1325). If the signal is exactly zero, it can be recovered with high probability by taking $x \gg \log(N)$. In particular, if we choose $x = 2\log(N)$, the threshold is $\delta_{N,n}^2 = \sigma^2(4\log(N) + 2\sqrt{2n\log(N)} + n)$ and the convergence rate in (5.6.3) is of order $O(\frac{1}{N})$. In practice, σ can be estimated by a MAD estimator computed on the coefficients of the highest level of all the n wavelet decompositions. Next, x and $\delta_{N,n}$ can be chosen such that (5.6.3) is lower than a given probability q. Letting $Ne^{-x} = q$, we obtain the threshold

$$\delta_{N,n} = \hat{\sigma}\left(2\log\left(\frac{N}{q}\right) + 2\sqrt{n\log\left(\frac{N}{q}\right)} + n\right)^{\frac{1}{2}}.$$

Assuming that $\omega_{j,k}(Z) = 0$ almost surely for some level (j,k) is a strong assumption that can be hardly met in practice. Hopefully, this method still works if the wavelet support of $s(\cdot,z)$ does not vary too much with z. In particular, it may be applied if there exists a common set S of indexes (j,k) such that, for any z, the projection of $s(\cdot,z)$ on $\mathrm{Vect}(\psi_{jk} \,|\, (j,k) \in S)$ is not too far from $s(\cdot,z)$ for the L^2 norm.

Chapitre 6

Comparaison de méthodes fonctionnelles appliquées aux données de vol

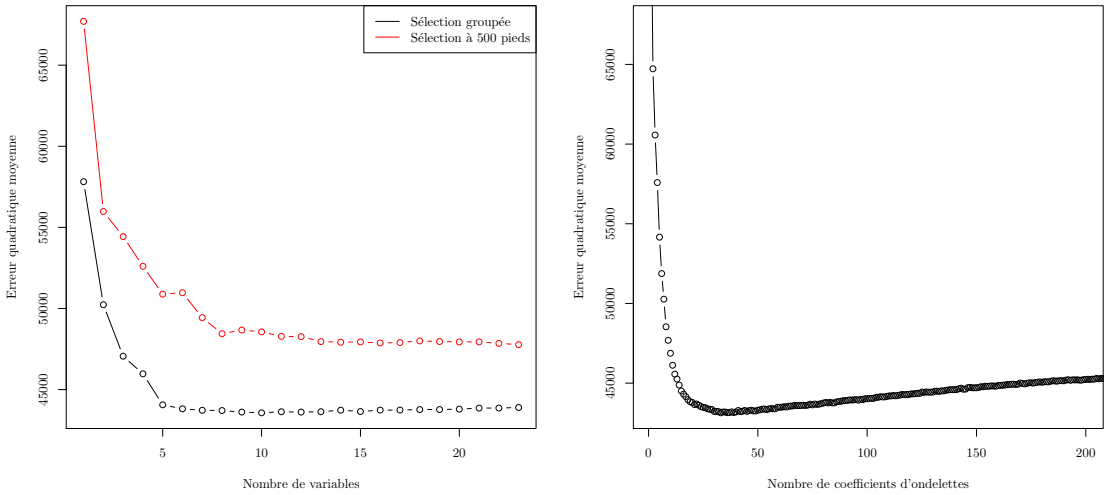
Résumé. Dans ce chapitre, nous étudions différentes méthodes de sélection de variables pour la prédiction des distances d'atterrissage. Nous comparons, tout d'abord les trois approches employées dans ce mémoire, la sélection à 500 pieds (Chapitre 2), la sélection des coefficients d'ondelettes (Chapitre 4) et la sélection groupée (Chapitre 5). Nous confrontons ensuite l'algorithme de sélection groupée des forêts aléatoires avec le *Group Lasso*, méthode bien connue en régression linéaire.

Sommaire

6.1 Comparaison des Chapitres 2, 4 et 5

Dans ce mémoire, nous avons utilisé différentes approches pour étudier le risque d'atterrissage long. Dans le Chapitre 2, les données étaient observées à l'altitude de 500 pieds. Cette analyse a donné une première réponse satisfaisante au problème de la classification des atterrissages longs. Néanmoins, se restreindre à une altitude fixée *a priori* induit une perte d'information car des incidents peuvent survenir à n'importe quel moment du vol et conduire à un atterrissage long. C'est pourquoi, des méthodes d'analyse de données fonctionnelles ont été employées par la suite au moyen d'algorithmes de sélection avec les forêts aléatoires. La démarche utilisée dans le Chapitre 4 est une méthode non groupée où les coefficients d'ondelettes sont directement sélectionnés. Les paramètres de vol choisis sont ceux dont les coefficients d'ondelettes sont les plus fréquemment sélectionnés. L'approche employée dans le Chapitre 5 considère les paramètres de vol comme des groupes de coefficients. L'algorithme de sélection est basé sur la mesure d'importance groupée que l'on a proposé.

Nous comparons donc ces trois approches pour la régression des distances d'atterrissage (données `Reg1`). Dans chaque cas, nous répétons la procédure de sélection 100 fois.



(a) Sélection à 500 pieds et sélection groupée

(b) Sélection non groupée

FIGURE 6.1 – (a) Erreurs quadratiques en fonction du nombre de paramètres de vol et (b) erreurs quadratiques en fonction du nombre de coefficients d'ondelettes.

La Figure 6.1 donne les erreurs de prédiction moyennes en fonction du nombre de paramètres de vol pour la sélection à 500 pieds et pour la sélection groupée (Figure 6.1a). Pour la sélection non groupée, nous donnons les erreurs en fonction du nombre de coefficients d'ondelettes (Figure 6.1b). Nous observons tout d'abord que la sélection à 500 pieds est moins performante que les deux autres approches. Ce résultat confirme le fait que l'information prédictive ne peut être complètement captée en observant uniquement les données à 500 pieds. Les deux autres approches donnent des résultats similaires en terme d'erreur de prédiction. L'erreur minimale est en effet de 43 136 pour la sélection non groupée (Figure 6.1b) alors qu'elle est de 43 566 pour la sélection groupée (Figure 6.1a). La sélection groupée est donc légèrement moins performante que la sélection non groupée. Cela s'explique par le fait que, pour un paramètre de vol, certains coefficients d'ondelettes peuvent être moins prédictifs que d'autres. Ils peuvent donc être éliminés par la procédure non groupée mais être sélectionnés par l'approche groupée.

Nous comparons également les fréquences de sélection pour chaque paramètre de vol. Pour la sélection non groupée, nous calculons la moyenne des fréquences de sélection de ces coefficients d'ondelettes. Au regard des variables les plus fréquemment sélectionnées, nous notons d'une part que la sélection non groupée (Figure 6.2b) et la sélection groupée (Figure 6.2c) donnent des résultats relativement similaires. En effet, les cinq paramètres de vol les plus fréquemment sélectionnés dans les deux cas sont l'altitude barométrique (ALT.STDC), l'angle d'attaque (AOA) puis la position des gouvernes (ELEVL), la vitesse air (CASC) et l'écart au plan de descente (GLIDE.DEVC).

Par ailleurs, nous identifions des différences entre la sélection à 500 pieds (Figure 6.2a) et les sélections fonctionnelles (Figures 6.2b et 6.2c). En effet, la sélection à 500 pieds fait apparaître la masse (GW.KG) ainsi que des paramètres relatifs à la vitesse (vitesse du vent CASC.GSC, vitesse air et vitesse d'approche VAPP) et à la puissance délivrée par les moteurs (N22C, N21C). La sélection groupée identifie la masse et surtout des paramètres relatifs au pilotage durant l'approche : position des gouvernes (ELEVR, ELEVL), angle

d'attaque, écart au plan de descente et altitude. Nous avons ainsi des informations plus complètes sur les trajectoires et sur le pilotage en phase d'approche. Il est naturel d'avoir la masse de l'avion commune aux deux sélections puisqu'elle varie peu au cours du vol. Notons de plus que la vitesse air et la vitesse d'approche (VAPP) sont très liées à l'altitude de 500 pieds. Cela provient d'une obligation des pilotes à respecter la vitesse d'approche à cette altitude.

Par ailleurs, il peut sembler contre-intuitif de voir le paramètre ALT.STDC être sélectionné dans plus de 70 % des cas pour des données observées à 500 pieds. Il s'agit ici de l'altitude barométrique, calculée par des différences de pressions atmosphériques par rapport à une pression de référence définie par les conditions atmosphériques standards du niveau de la mer (soit environ 1013 hPa). Ce paramètre varie naturellement en fonction des conditions météorologiques autour de l'avion et peut avoir une valeur très supérieure à zéro à l'atterrissage. Or les données observées à 500 pieds sont extraits selon la radio altitude, mesurée par des capteurs.

6.2 Comparaison des forêts aléatoires avec le *Group Lasso*

L'algorithme de sélection *backward* RFE, utilisé dans l'ensemble de nos travaux de thèse, est une alternative non linéaire aux techniques de régression de type Lasso (voir le Chapitre 2). C'est pourquoi nous comparons les résultats de la sélection groupée obtenus dans le Chapitre 5 avec le *Group Lasso* introduit par Yuan and Lin (2006a). Cette approche est utilisée, notamment par Fan and James (2013) et Oliva et al. (2014), pour la régression d'une variable Y réelle par p variables fonctionnelles $X_1, \ldots, X_p$. Le modèle linéaire fonctionnel

$$Y = \sum_{u=1}^{p} \langle X_u, \beta_u \rangle_{L^2} + \varepsilon,$$

se réécrit

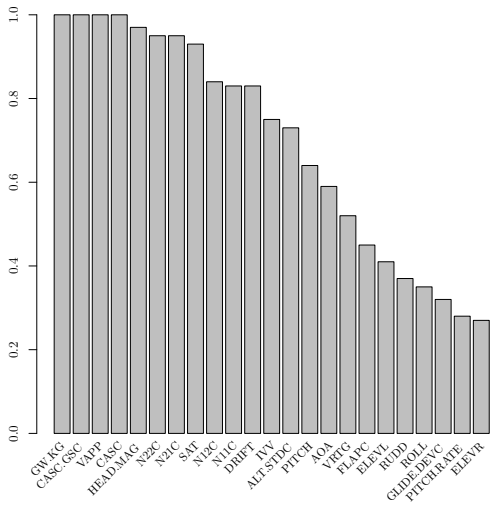
$$Y = \sum_{u=1}^{p} \mathbf{Z}_u^\top \theta_u + \epsilon,$$

où $\mathbf{Z}_u$ (resp. θ_u) est le vecteur d_u-dimensionnel des coefficients de base de la covariable X_u (resp. de la fonction β_u) et ϵ combine le bruit du modèle ε et les erreurs de l'approximation des éléments de $L^2([0,1])$ sur un sous-espace de dimension d_u (voir le Chapitre 3 pour plus de détails). Dans la suite, les groupes $\mathbf{Z}_u$ sont formés par les coefficients d'ondelettes retenus par la méthode du seuillage simultané (Section 3.3). Soit $\mathbf{Z} = (\mathbf{Z}_1, \ldots, \mathbf{Z}_p)$ le vecteur de l'ensemble des coefficients d'ondelettes et $\mathbf{D}_n$ l'échantillon de n couples indépendants et de même loi que $(\mathbf{Z}, Y)$.

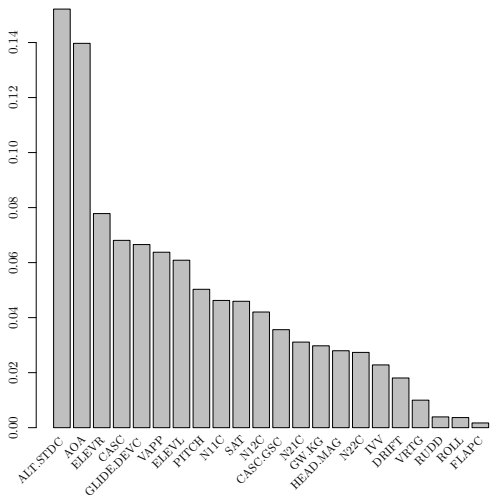
Pour estimer le vecteur des paramètres $\theta = (\theta_1, \ldots, \theta_p)$ de dimension $D = \sum_{u=1}^{p} d_u$, nous utilisons le critère proposé par Yuan and Lin (2006a) :

$$\hat{\theta} \in \underset{\theta \in \mathbb{R}^D}{\arg\min} \left\{ \|Y - \sum_{u=1}^{p} \mathbf{Z}_u^\top \theta_u\|_2^2 + \lambda \sum_{u=1}^{p} \sqrt{d_u} \|\theta_u\|_2 \right\}.$$

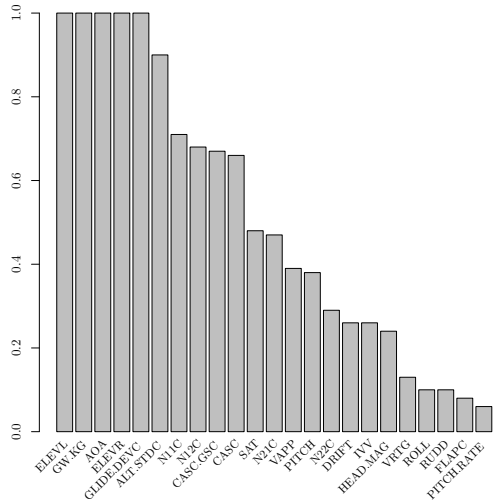
La procédure de sélection par le *Group Lasso* est la suivante. Tout d'abord, les données $\mathbf{D}_n$ sont découpées aléatoirement en trois échantillons indépendants, un échantillon de validation $\mathbf{D}_{val}$, un échantillon d'entrainement $\mathbf{D}_{tr}$ et un échantillon de test $\mathbf{D}_{te}$. L'ensemble $\mathbf{D}_{val}$ est utilisé pour l'optimisation du paramètre de régularisation λ par validation croisée 5-fold. Les ensembles $\mathbf{D}_{tr}$ et $\mathbf{D}_{te}$ sont respectivement utilisés pour estimer θ et pour calculer l'erreur de prédiction. Pour un $\lambda^\star$ optimisé par validation croisée, nous répétons



(a) Sélection à 500 pieds

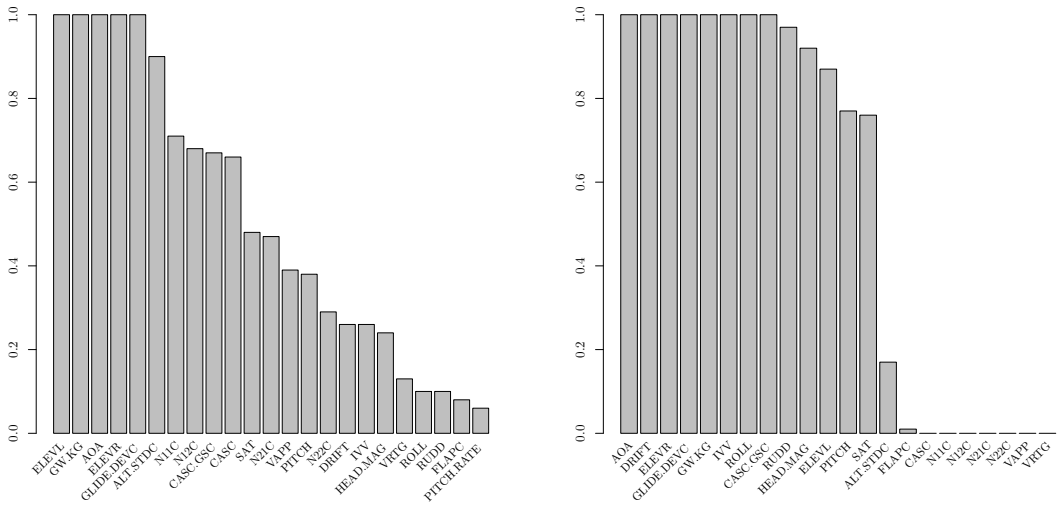


(b) Sélection non groupée



(c) Sélection groupée – Chapitre 5

FIGURE 6.2 – Fréquences de sélection de l'analyse à 500 pieds, de la méthode non groupée et de la méthode groupée



(a) Sélection groupée – Chapitre 5

(b) Group Lasso

Figure 6.3 – Comparaison de la sélection groupée avec les forêts aléatoires et du *Group Lasso*

la sélection sur 100 itérations en ré-échantillonnant les ensembles $\mathbf{D}_{tr}$ et $\mathbf{D}_{te}$.

La Table 6.1 donne les erreurs moyennes sur les 100 itérations ainsi que le nombre moyen de paramètres de vol sélectionnés par chaque méthode. Avec une erreur moyenne de 43 566, la sélection groupée est légèrement plus performante que le *Group Lasso* où l'erreur est de 44 893. Les forêts aléatoires sont également plus *sparses* en sélectionnant 10 variables en moyenne contre 12 pour le *Group Lasso*.

Méthode	**Nb de variables**	**Erreur de prédiction**
Sélection groupée	**10**	**43 566**
Group Lasso	12	44 893

Table 6.1 – Erreurs de prédiction et nombre moyen de paramètres de vol sélectionnés calculés sur 100 itérations.

Les fréquences de sélection des paramètres de vol sont représentées dans les Figures 6.3a et 6.3b. La première remarque que nous pouvons faire est que le *Group Lasso* est plus stable que la sélection groupée. En effet, la Figure 6.3a montre que toutes les variables ont été sélectionnées au moins 5 fois sur les 100 itérations par les forêts aléatoires à l'inverse du *Group Lasso* où les fréquences de sélections sont globalement plus élevées. De plus, les informations données par les deux méthodes sont cohérentes. En effet, certains paramètres sélectionnés à chaque itération par le *Group Lasso* se retrouvent parmi les plus fréquemment sélectionnés par les forêts aléatoires. Il s'agit de l'angle d'attaque (AOA), de la position des gouvernes (ELEVL, ELEVR), de l'écart au plan de descente (GLIDE.DEVC) et de la masse (GW.KG).

6.3 Conclusion du chapitre

Dans ce chapitre, nous avons étudié le risque d'atterrissage long à travers la prédiction des distances d'atterrissage (données `Reg1`). Nous avons comparé les trois approches de sélection des paramètres de vol employées dans ce mémoire : la sélection des données à 500 pieds (Chapitre 2), la sélection des coefficients d'ondelettes (Chapitre 4) et la sélection groupée (Chapitre 5). Nous observons tout d'abord que la sélection à 500 pieds est moins performante que les approches fonctionnelles. Ce résultat n'est pas surprenant : une sequence entière de vol contient plus d'information qu'un instant donné. Nous confirmons cette observation au regard des résultats donnés par les deux procédures fonctionnelles (sélection des coefficients d'ondelettes et sélection groupée). En effet, les paramètres de vol identifiés sont relatifs à la trajectoire de l'avion ainsi qu'au pilotage en phase d'approche : la position des gouvernes, l'angle d'attaque, l'écart au plan de descente et l'altitude. La sélection à 500 pieds a identifié la masse, la vitesse du vent ainsi que la vitesse air et la vitesse d'approche. Ces deux derniers paramètres sont liés à l'altitude de 500 pieds car les pilotes doivent respecter la vitesse d'approche à cet instant pour assurer une bonne stabilisation de l'avion.

Nous avons également comparé la sélection groupée des forêts aléatoires avec le *Group Lasso*, méthode de régression linéaire. Nous avons observé de meilleures performances prédictives avec les forêts aléatoires. Les deux méthodes donnent des résultats cohérents en terme de sélection des paramètres de vol : la plupart des paramètres de vol sélectionnés par le *Group Lasso* sont parmi les plus fréquemment sélectionnés par les forêts aléatoires.

Chapitre 7

Valorisation industrielle – Produit *FlightScanner*

Résumé. Ce chapitre a pour objectif d'illustrer la façon dont les outils proposés pendant la thèse sont valorisés au sein de la société Safety Line. Nous présentons le produit *FlightScanner* dans lequel s'intègrent nos travaux. Au sein de la plateforme *Operation Health Monitoring* dédiée à l'analyse des données aéronautiques, cette solution innovante dans l'exploitation aérienne permet à la fois le monitoring des risques et le suivi des facteurs qui les influencent. Dans un second temps, nous illustrons l'utilisation de ce logiciel pour l'étude de l'atterrissage dur.

Sommaire

7.1 Introduction

Safety Line propose à ses clients des solutions logicielles innovantes pour la gestion des risques opérationnels. En particulier, *Operation Health Monitoring* (OHM) est une plateforme d'analyse automatisée des données aéronautiques de nouvelle génération. Plusieurs services sont fournis par Safety Line à travers OHM afin de proposer aux compagnies aériennes et aux aéroports une gamme de produits dédiés à la sécurité dans le transport aérien.

Le service *Airport* analyse les données des radars d'approche pour déterminer l'état des pistes de manière autonome dans le but de minimiser les interruptions de services liées aux inspections et aux opérations de déneigement. Le service *Maintenance*, non développé à

ce jour, a pour objectif d'optimiser les opérations de maintenance fournissant, équipement par équipement, l'état exact de l'avion à l'issu de chaque vol.

À travers le logiciel *FlightScanner*, la société Safety Line propose une nouvelle approche de la gestion des risques basée sur les outils proposés dans ce manuscrit. Amélioration majeure du traitement des données des enregistreurs de vol, *FlightScanner* permet d'identifier les facteurs explicatifs des situations à risque de manière automatique et sur l'ensemble des vols. Il permet aux compagnies aériennes de détecter des combinaisons de facteurs qui n'étaient pas envisagées auparavant, compte tenu de la complexité des données. La société propose ainsi l'automatisation d'un processus jusqu'alors réalisé de manière exclusivement manuelle. Pour cela, Safety Line définit en collaboration avec les compagnies aériennes, les risques à surveiller (atterrissage long, dur, etc.) puis met en place des serveurs dédiés pour le traitement des données. Les résultats (facteurs de risque, différents indicateurs) sont finalement consultables via une interface accessible en ligne. Les utilisateurs ont alors la possibilité d'évaluer objectivement les risques et de les suivre dans le temps.

Afin de proposer un produit complet, Safety Line intègre également dans *FlightScanner* une version améliorée du *Flight Data Monitoring* (FDM) permettant de détecter des dépassements de seuils survenus en vol. Des fonctionnalités nouvelles sont apportées, comme par exemple la possibilité de fixer des seuils différents de ceux qui sont habituellement employés. Cela permet aux exploitants aériens de s'assurer du respect de la conformité réglementaire.

FlightScanner a été conçu à partir des outils développés dans ce manuscrit et constitue un aboutissement de ces travaux de thèse.

7.2 Processus de traitement des données de vol implémenté dans *FlightScanner*

Le processus de traitement des données de vol implémenté dans le produit *FlightScanner* procède en trois étapes : une phase de collecte des données, une phase d'apprentissage et une phase de prédiction (Figure 7.1). Les outils développés dans les phases d'apprentissage et de prédiction sont directement issus de nos travaux de thèse.

Dans cette section, nous ne détaillons pas les aspects techniques des algorithmes et du calcul des indicateurs de risque. Nous les préciserons plus formellement dans la Section 7.4 dans le cas du risque d'atterrissage dur.

Collecte des données. En sortie des enregistreurs de vol, les données sont stockées dans des fichiers binaires. Les compagnies aériennes clientes fournissent à Safety Line l'accès à ces fichiers. Cependant, ils ne sont pas directement utilisables par les algorithmes statistiques. C'est pourquoi l'équipe de développement déploie des algorithmes propriétaires permettant de décoder les données brutes associant une connaissance métier et une expertise informatique. Pour cela, il a été mis en place des serveurs distribués pour stocker et extraire les données au moyen de la technologie *Hadoop*[1] et d'algorithmes de type *MapReduce* (Dean and Ghemawat; 2008).

Phase d'apprentissage. La phase d'apprentissage vise à rechercher dans un historique d'exploitation les paramètres de vol ayant le plus d'influence sur un risque donné, l'atterrissage dur par exemple. Cette phase de calcul intensive est conduite à intervalles de temps réguliers (mensuel ou trimestriel) et se déroule de la façon suivante :

1. http ://hadoop.apache.org

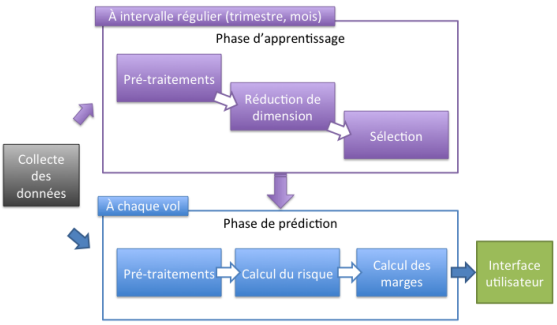


Figure 7.1 – Processus d'analyse des données de vol implémenté dans *FlightScanner*.

- **Pré-traitements.** Cette étape consiste, pour chaque risque, à extraire la séquence de vol à analyser (la phase d'approche par exemple) puis à pré-sélectionner les paramètres de vol. Les signaux sont ensuite recalés et normalisés. Cette étape est principalement guidée par la connaissance des experts aéronautiques.

- **Réduction de dimension.** L'étape de réduction de dimension vise à représenter, paramètre par paramètre, l'ensemble des vols sur une base de fonctions commune (Chapitre 3). Deux méthodes sont actuellement implémentées : la méthode du seuillage par ondelettes et l'Analyse en Composantes Principales Fonctionnelle (ACPF). Les ondelettes ont l'avantage d'être localisées à la fois en fréquence et en temps et permettent de calculer l'importance temporelle proposée dans le Chapitre 5. L'ACPF est également intéressante car elle fournit une représentation parcimonieuse des données tout en conservant une bonne qualité d'approximation.

- **Sélection.** Cette étape recherche parmi l'ensemble des paramètres de vol ceux qui sont les plus prédictifs pour le risque étudié. Le produit *FlightScanner* implémente l'algorithme de sélection groupée introduit dans le Chapitre 5. À partir des paramètres sélectionnés, le logiciel propose aux utilisateurs des profils-types définissant des situations de vol sures. Cet outil essentiel du produit permet de voir comment un vol se situe par rapport à ces situations. Plus on observe des déviations par rapport aux profils-types, plus le risque est important. Safety Line propose ainsi une représentation graphique pour aider les gestionnaires de la sécurité à mieux comprendre les causes potentielles des risques surveillés (atterrissages longs ou durs, par exemple). Les aspects formels de la construction des profils-types seront donnés dans la Section 7.4.

Phase de prédiction. La seconde phase de calcul consiste à évaluer le niveau de risque de chaque nouveau vol reçu à partir des résultats de la phase précédente. Elle permet aux utilisateurs d'avoir des mesures objectives sur le niveau de risque à l'échelle d'un vol, d'une flotte ou d'un aéroport et surtout de pouvoir suivre son évolution dans le temps. La phase de prédiction se compose de trois étapes :

- **Pré-traitements.** Les pré-traitements effectués sur les nouveaux vols sont les mêmes que la phase d'apprentissage.

- **Calcul du risque.** Cette étape consiste à mesurer le risque du nouveau vol sachant les paramètres de vol sélectionnés. La valeur prédite est un indicateur de risque naturel et permet de mesurer la proximité du vol à une situation dangereuse. Dans le cas d'un problème de classification, l'indicateur est donné par la probabilité *a posteriori* estimée (Chapitre 2).

- **Calcul des marges.** En complément du calcul du risque pour chaque nouveau vol, le logiciel offre la possibilité de calculer des écarts aux marges de sécurité. Plus précisément, pour chaque paramètre de vol sélectionné, il est possible d'évaluer les déviations des nouveaux vols par rapport aux profils-types construits lors de la phase de sélection. Un second indicateur de risque est alors donné par la déviation moyenne d'un vol au profil-type correspondant. Cette déviation peut être annonciatrice d'une situation risquée.

7.3 Interface utilisateur

L'interface de *FlightScanner* permet à l'utilisateur de consulter les résultats des analyses à plusieurs niveaux. Une fois connecté, l'utilisateur accède à un tableau de bord résumant les dernières analyses effectuées (Figures 7.2 et 7.3). Il peut en particulier consulter le nombre d'événements FDM détectés ("Findings" dans l'interface) durant la période analysée et comparer les résultats avec la période précédente. Par exemple, on peut observer une diminution du nombre d'événements "High speed at take off" pour une semaine donnée par rapport à la semaine précédente (encadré "Finding trends" dans la Figure 7.3). De la même façon, l'utilisateur peut consulter l'évolution du nombre de vols à risque par rapport à la période précédente (encadré "Risk trends" dans la Figure 7.3). En complément, la liste des derniers vols analysés est affichée pour avoir un aperçu rapide de ceux qu'il faut surveiller (encadré "Last flights" dans la Figure 7.3). Pour chacun de ces vols, les chiffres donnés correspondent au nombre de risques (atterrissage long, dur, etc.) pour lesquels le vol est considéré à risque. L'interface propose ainsi trois niveaux de risque pour chaque vol. Par exemple, le premier vol de la liste admet un seul risque de niveau 3. Le deuxième vol admet deux risques de niveau 2 et trois risques de niveau 1.

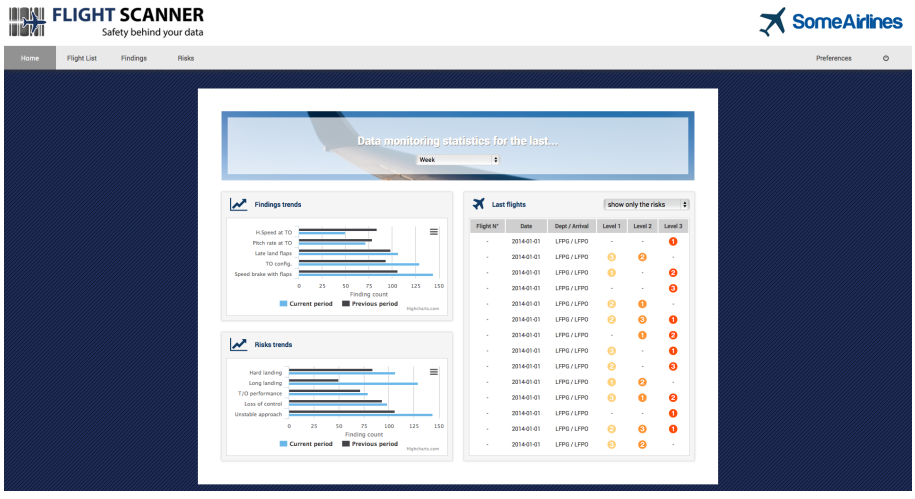


FIGURE 7.2 – Tableau de bord (données fictives).

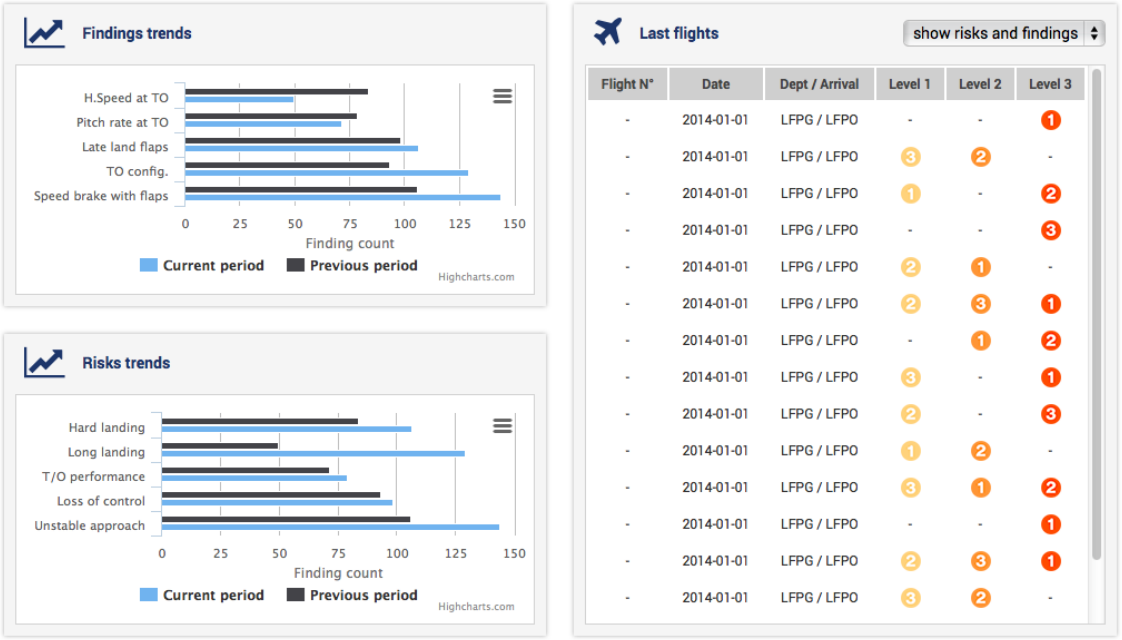


Figure 7.3 – Tableau de bord (données fictives).

L'onglet "Flight list" fournit une liste plus complète des vols analysés (Figure 7.4). L'utilisateur peut choisir de les filtrer notamment par la date, le type d'avion ou l'aéroport de destination. Ici encore, le logiciel propose à la fois le nombre de risques et le nombre d'événements détectés pour trois niveaux de détection. L'utilisateur a ensuite la possibilité d'afficher le détail d'un vol pour approfondir l'analyse en cherchant les facteurs explicatifs pour les différents risques.

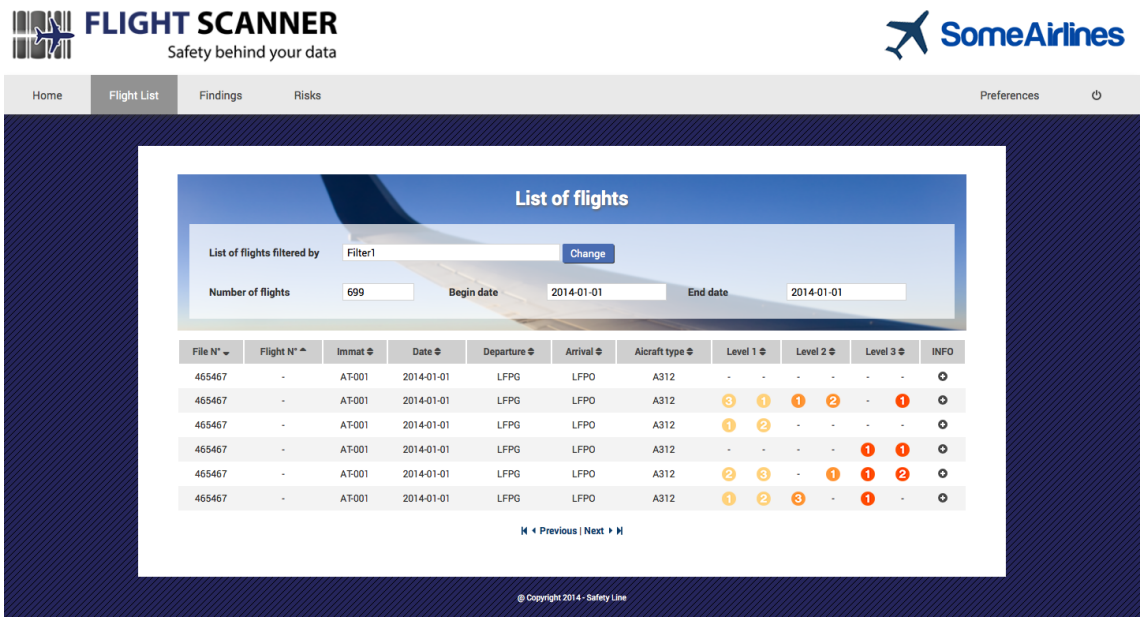


Figure 7.4 – Liste des derniers vols analysés (données fictives).

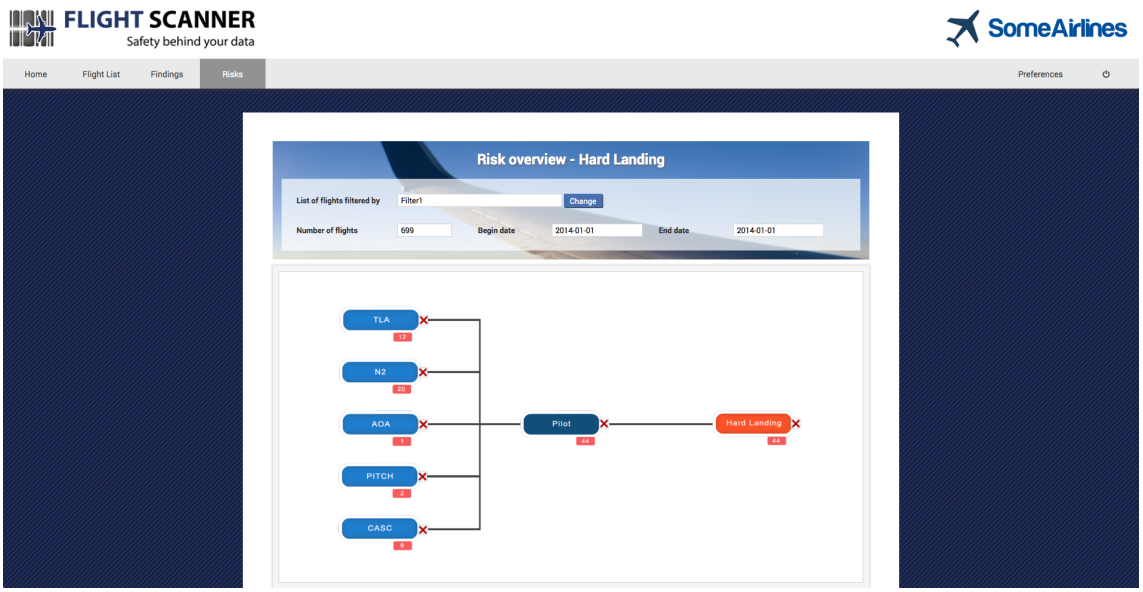


FIGURE 7.5 – Liste des paramètres de vols sélectionnés (données `Reg2`). Le chiffre donné sous un paramètre de vol correspond au nombre de vols qui sont jugés à risque pour ce paramètre. Ces données sont fictives.

L'onglet "Findings" donne le détail des événements détectés durant la période analysée et permet de suivre leur nombre dans le temps. Nous n'entrons pas plus dans les détails de cette fonctionnalité car elle utilise des algorithmes différents de ceux proposés dans ce manuscrit.

La page "Risk" présente les résultats de la phase d'apprentissage (Figure 7.5). Safety Line a choisi d'utiliser une représentation classique en modélisation des risques opérationnels pour présenter les résultats des analyses statistiques, notamment de sélection des paramètres de vol. Cette représentation, communément appelée *bow-tie*, lie un événement redouté avec les causes possibles et ses conséquences. Pour représenter les résultats de la sélection des paramètres de vol, il a été décidé d'afficher uniquement l'événement redouté et les causes possibles. Dans l'exemple, l'événement redouté est l'atterrissage dur et les causes, issues de la phase d'apprentissage, sont des paramètres de vol liés au pilotage (à gauche de la représentation). Le chiffre donné sous un paramètre de vol correspond au nombre de vols qui sont jugés à risque pour ce paramètre.

La Figure 7.6 présente le détail d'un vol. L'utilisateur peut alors examiner en détail les résultats des analyses pour le risque d'atterrissage dur par exemple. On peut voir les paramètres de vol pour lesquels ce vol a dévié du profil-type. En cliquant sur un des éléments du bow-tie, l'utilisateur affiche un graphe où le vol est superposé au profil moyen pour le paramètre correspondant. Par exemple, dans le cas de l'atterrissage dur, la Figure 7.7 affiche le profil-type de la vitesse air et le vol à surveiller (en rouge). L'utilisateur peut alors observer une déviation de ce vol à risque pour la vitesse air. Nous reviendrons à ce cas particulier dans la section suivante dédiée à l'étude du risque d'atterrissage dur.

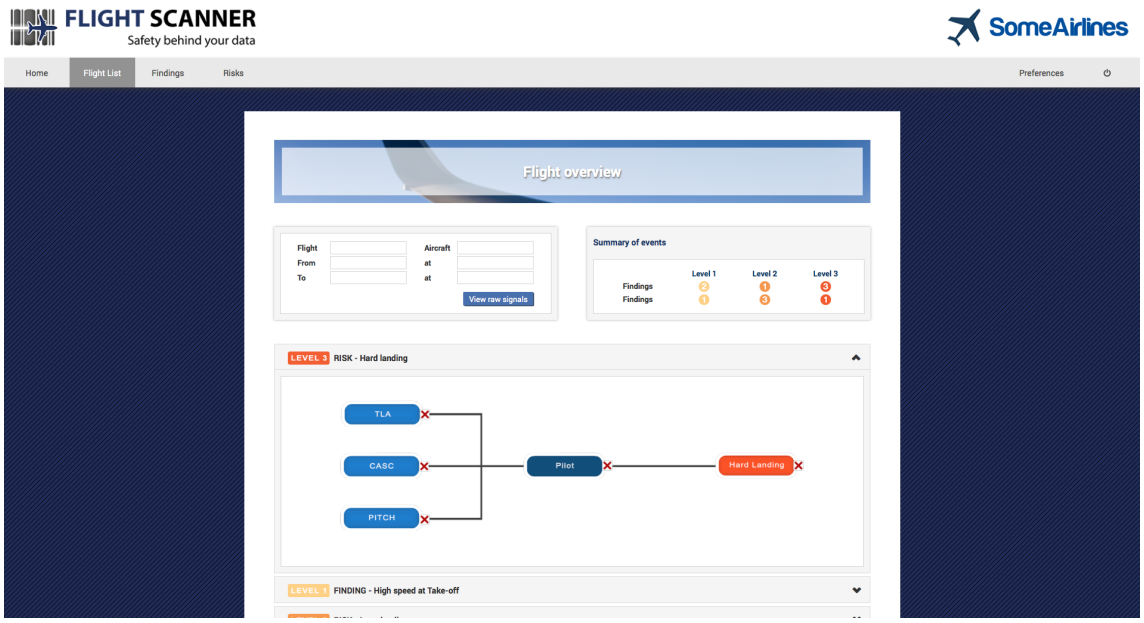


Figure 7.6 – Détail d'un vol (données `Reg2`).

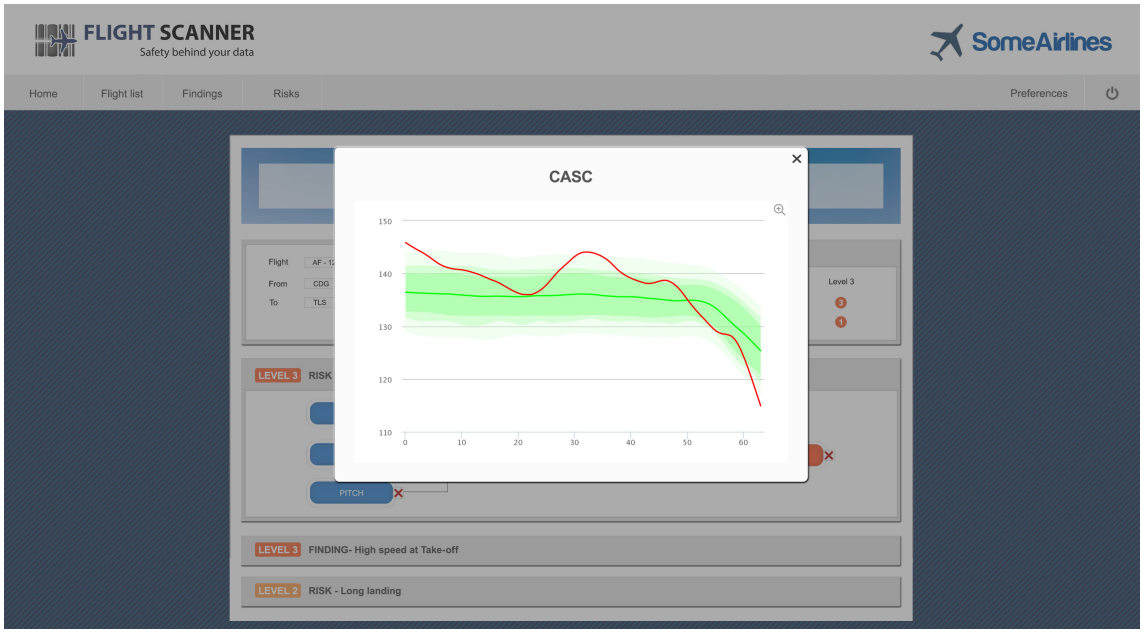


Figure 7.7 – Comparaison d'un vol avec le profil moyen de vitesse (données `Reg2`).

7.4 Étude de l'atterrissage dur

Dans cette section, nous présentons une utilisation opérationnelle des outils implémentés dans *FlightScanner* pour l'étude du risque d'atterrissage dur. Nous décrivons en particulier une interprétation des résultats fournis par les interfaces utilisateur.

7.4.1 Description du risque d'atterrissage dur

Les atterrissages durs sont des événements redoutés par les opérateurs aériens au même titre que les sorties de piste. Ils sont dus à une vitesse verticale excessive au moment du toucher des roues sur la piste traduisant une forte compression des trains. Des opérations de maintenance sont nécessaires dès lors que la vitesse verticale excède 600 pieds par minute ou si l'accélération verticale (VRTG) est supérieure à 2.6 g. En pratique, les compagnies aériennes analysent vol par vol l'intensité des atterrissages au moyen de différents seuils définis *a priori*. Les atterrissages sont classés en trois niveaux :

- Niveau 1 si VRTG $>$ 1.4 g ;
- Niveau 2 si VRTG $>$ 1.8 g ;
- Niveau 3 si VRTG $>$ 2.1 g.

La méthodologie choisie par Safety Line consiste à traiter tous les vols d'une flotte pour mesurer le risque d'atterrissage dur par les méthodes statistiques présentées dans ce manuscrit. Pour cela, il a été décidé, en accord avec les experts, de mesurer l'intensité de l'atterrissage par le maximum de l'accélération verticale qui à lieu au moment du toucher. L'analyse du risque d'atterrissage dur se base donc sur un problème de régression où l'accélération verticale au toucher est à prédire. En outre, l'expertise métier permet de supposer que les atterrissages les plus "fermes" (ayant une accélération verticale observée forte mais inférieure aux seuils de détection) ont des points communs avec les atterrissages effectivement durs. Autrement dit, les facteurs de risque pour les atterrissages "fermes" sont vus comme des causes potentielles d'atterrissages durs.

7.4.2 Données

Le jeu de données considéré dans ce chapitre contient 699 vols correspondant à un type d'avion long-courrier et à un aéroport de destination (données `Reg2`). Les 29 paramètres de vol pré-sélectionnés par les experts se répartissent en trois catégories :

- Commandes de pilotage : actions sur le manche et le palonnier, position de la manette des gaz et vitesse sélectionnée ;
- État de l'avion : angle d'attaque, assiette, roulis, position des surfaces de contrôle (volets, gouvernes), régimes moteur, vitesses observées et masse ;
- Environnement : vitesse du vent, cap magnétique, écart au plan de descente, angle de dérive, radio altitude et position GPS.

À la différence des jeux de données utilisés précédemment, quatre paramètres de vol ne sont observés qu'au moment du toucher car ils varient très peu pendant la phase d'atterrissage. Il s'agit de la masse, de la position des volets, de la latitude et de la longitude. Les autres paramètres sont observés chaque seconde durant la minute qui précède l'atterrissage et quatre secondes après afin de tenir compte de la décélération de l'avion et d'éventuels rebonds. Ces variables fonctionnelles sont donc échantillonnées sur une grille temporelle de 64 points à pas constant.

7.4.3 Identification des facteurs de risque pour l'atterrissage dur

Aspects formels

Afin de bien comprendre le fonctionnement de *FlightScanner*, nous précisons quelques éléments formels pour la réduction de dimension, la sélection des paramètres de vol et la construction de profils-types.

Réduction de dimension. Une des méthodes de réduction de dimension implémentée dans *FlightScanner* est l'Analyse en Composantes Principales Fonctionnelle (ACPF). En accord avec le Chapitre 3, la méthode choisie consiste à lisser préalablement les signaux puis à effectuer une ACP multivariée sur les coefficients de base (Ramsay and Silverman; 2005). Dans le logiciel, un lissage Spline à partir de 32 coefficients est utilisé. Le nombre final de composantes principales est déterminé de sorte à expliquer 90 % de la variance des signaux.

À l'issu de l'étape de réduction de dimension, nous concaténons l'ensemble des composantes principales pour chaque variable fonctionnelle ainsi que les variables réelles. Notons $\mathbf{D}_n$ le nouvel échantillon d'apprentissage qui sera utilisé dans l'étape de sélection.

Sélection des paramètres de vol. La méthode implémentée dans *FlightScanner* pour l'identification des paramètres de vol influents est la procédure de sélection groupée avec les forêts aléatoires proposée dans le Chapitre 5. Rappelons que cette procédure élimine itérativement les groupes de variables les moins importants au sens du critère d'importance groupée proposé dans le Chapitre 5. Les paramètres de vol fonctionnels sont, en effet, considérés comme des groupes formés par les premières composantes principales issues de l'étape de réduction de dimension. Les paramètres de vol observés uniquement à l'atterrissage sont vus comme des groupes de taille 1. Ces groupes étant de taille différentes, nous utilisons la version normalisée de la mesure d'importance groupée (voir le Chapitre 5).

La procédure de sélection des paramètres de vol développée par Safety Line est résumée dans l'algorithme 4. Tout d'abord, nous sous-échantillonnons aléatoirement les données d'apprentissage $\mathbf{D}_n$ en trois ensembles : un ensemble d'entrainement $\mathbf{D}_{tr}$, un ensemble de test $\mathbf{D}_{te}$ et un ensemble de validation $\mathbf{D}_{val}$ (étape 1). Les échantillons $\mathbf{D}_{tr}$ et $\mathbf{D}_{te}$ sont utilisés dans l'algorithme de sélection groupée pour respectivement construire des forêts aléatoires et calculer l'erreur (étape 2). Les paramètres sélectionnés sont ceux qui minimisent l'erreur de prédiction calculée à chaque étape de l'algorithme *backward*. Enfin, à partir des paramètres sélectionnés, une prédiction est effectuée pour chaque observation contenue dans l'ensemble $\mathbf{D}_{val}$ (étape 3).

Algorithm 4 Procédure de sélection implémentée dans *FlightScanner*

1: Sous-échantillonner $\mathbf{D}_n$ en trois ensembles $\mathbf{D}_{tr}$, $\mathbf{D}_{te}$ et $\mathbf{D}_{val}$
2: Sélectionner les paramètres de vol à partir de $\mathbf{D}_{tr}$ et $\mathbf{D}_{te}$
3: Prédire les observations de $\mathbf{D}_{val}$
4: Répéter 100 fois les étapes 1 à 3 pour stabiliser les résultats

La procédure est répétée 100 fois afin de stabiliser les résultats et de classer les variables selon leur fréquence de sélection. De plus, les prédictions obtenues pour les n vols de la base d'apprentissage serviront à la construction de profils-types.

Construction de profils-types. Comme nous l'avons écrit précédemment, les profils-types de vol constituent un des outils essentiels de *FlightScanner*. L'objectif est de voir

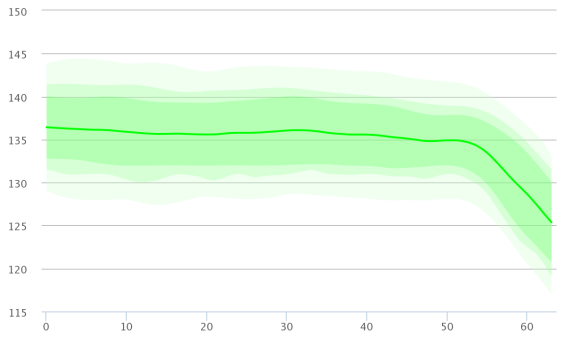


FIGURE 7.8 – Un exemple de profil-type.

comment un vol se place par rapport à une situation sure pour certains paramètres à surveiller. La méthode de construction de ces profils moyens utilise les prédictions de l'accélération verticale effectuées pour les n vols de la base d'apprentissage, notées $\hat{\mathbf{Y}} = (\hat{Y}_1, \dots, \hat{Y}_n)$. Pour un paramètre de vol fixé X_u, soit $X_{u1}, \dots, X_{un}$ les n observations de la base d'apprentissage. Un profil de vol "sûr" est défini comme la courbe moyenne parmi les vols dont l'accélération verticale prédite est inférieure au premier quartile de $\hat{\mathbf{Y}}$. Autrement dit, si $\mathcal{J} = \{i : \hat{Y}_i \leq \mathrm{Q}_1(\hat{\mathbf{Y}})\}$ est l'ensemble des indices correspondant à ces vols, alors le profil-type est défini par

$$\bar{X}_u(t) = \frac{1}{|\mathcal{J}|} \sum_{i \in \mathcal{J}} X_{ui}(t),$$

pour tout $t \in \{t_N, \dots, t_N\}$. Un exemple d'une telle courbe est donné par la Figure 7.8.

Les profils-types de vol ainsi construits permettent d'identifier des situations de vol sures. Des déviations par rapport à ces profils peuvent être annonciatrices de situations risquées. C'est pourquoi il est nécessaire de détecter les vols qui dévient de ces profils.

Sorties du logiciel

Dans l'interface utilisateur de *FlightScanner*, les résultats de la sélection des paramètres de vol sont représentés par un *bow-tie* (Figure 7.9). Les cinq paramètres de vol les plus influents pour le risque d'atterrissage dur sont les commandes de poussée des moteurs (TLA), les régimes moteur (N2), l'angle d'attaque (AOA), l'assiette (PITCH) et la vitesse air (CASC). Tous ces paramètres sont relatifs au pilotage.

Pour interpréter ces résultats, il faut mentionner que la poussée des moteurs et les régimes moteurs sont des paramètres de vol relatifs à la vitesse de l'avion. Les résultats montrent donc que des variations de vitesse durant la minute précédant l'atterrissage sont déterminantes. Il est d'ailleurs connu qu'une vitesse trop faible quelques mètres au dessus de la piste peut conduire à une perte de portance et donc à des atterrissages durs. L'angle d'attaque et l'assiette sont des paramètres liés à l'arrondi au moment de l'atterrissage où le pilote doit cabrer l'avion pour réduire la pente de descente et augmenter la portance. L'arrondi est donc identifié comme facteur de risque. En effet, la connaissance métier permet de dire qu'un arrondi mal maîtrisé est source d'incidents à l'atterrissage.

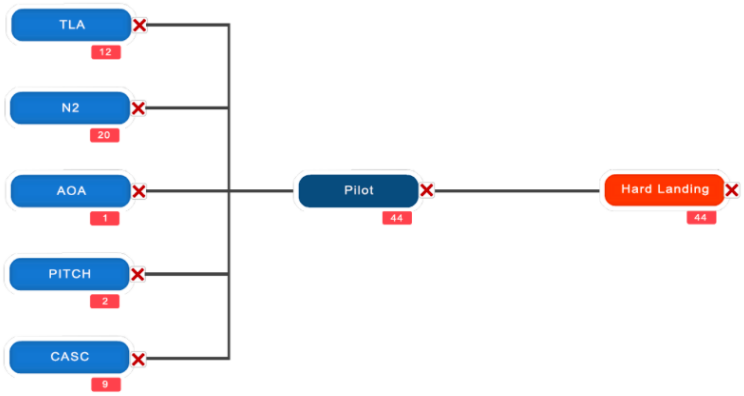


Figure 7.9 – Liste des paramètres de vols sélectionnés (données `Reg2`).

7.4.4 Prédiction et comparaison aux profils-types

La seconde étape du traitement des données de vol consiste à prédire chaque nouveau vol reçu. Dans le cas de l'atterrissage dur, nous prédisons la valeur de l'accélération verticale à l'atterrissage. Nous nous servons de ces prédictions comme indicateur de risque. Certains vols "à risque" sont comparés aux profils-types pour les paramètres sélectionnés. Cela permet aux utilisateurs d'identifier plus facilement les facteurs de risque.

Aspects formels

Afin de mesurer l'écart entre des vols particuliers et les profils-types, un indice de sévérité est défini. Plus formellement, pour un paramètre de vol X_u, définissons une suite de variables aléatoires i.i.d. $\{D_{i,t}, i \in \mathcal{J}\}$ par

$$D_{i,t} := (X_{ui}(t) - \bar{X}_u(t))^2,$$

pour tout $i \in \mathcal{J}$ et tout $t \in \{t_1, \ldots, t_N\}$. Soient F_t, la fonction de répartition des variables $\{D_{i,t}, i \in \mathcal{J}\}$ et $F_{n,t}$ la fonction de répartition empirique associée. Pour un nouveau vol $X_{u,n+1}$, $F_{n,t}(D_{n+1,t})$ mesure sa déviation à la courbe $\bar{X}_u$ en t avec

$$D_{n+1,t} = (X_{n+1}(t) - \bar{X}(t))^2.$$

L'indice de sévérité implémenté dans *FlightScanner* est donné par la déviation moyenne de $X_{u,n+1}$ à $\bar{X}_u(t)$:

$$\mathcal{S} = \frac{1}{N} \sum_{\ell=1}^{N} F_{n,t_\ell}(D_{n+1,t_\ell}).$$

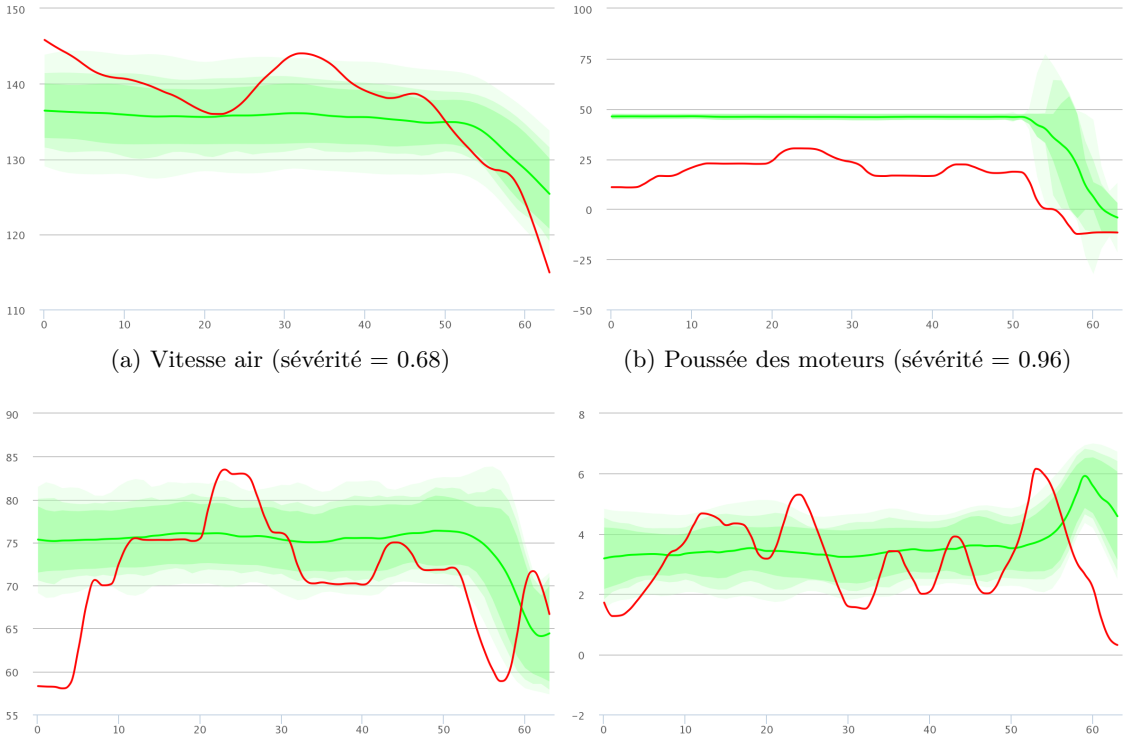
Plus cet indice est proche de 1, plus la distance ℓ_2 entre $X_{u,n+1}$ et $\bar{X}_u(t)$ est grande.

Sorties du logiciel

Le logiciel *FlightScanner* permet aux utilisateurs d'extraire certains vols à risque et de les comparer aux profil-types. Nous avons donc extrait un de ces vols à partir de l'interface utilisateur ainsi que les sorties graphiques pour quatre des paramètres de vol les plus

influents (Figure 7.10). Les gestionnaires de la sécurité peuvent observer les déviations de ce vol par rapport aux profils-types. C'est particulièrement le cas pour le paramètre TLA (poussée des moteurs) où le vol observé est très en dessous de la courbe moyenne.

En outre, l'observation jointe de ces quatre graphiques permet de préciser les facteurs de risque pour l'atterrissage dur. En effet, les Figures 7.10a, 7.10b et 7.10c confirment que la vitesse est déterminante durant la dernière minute : le vol à risque a une vitesse trop importante une minute avant l'atterrissage. L'action du pilote (Figure 7.10b) mène à la réduction prématurée de la vitesse. De plus, la courbe d'assiette montre que l'arrondi est effectué légèrement trop tôt, soit entre 10 et 5 secondes avant le toucher (Figure 7.10d). En comparaison, l'arrondi du vol moyen est observé dans les 5 dernières seconde avant le toucher.



(a) Vitesse air (sévérité = 0.68)

(b) Poussée des moteurs (sévérité = 0.96)

(c) Régime moteur (sévérité = 0.65)

(d) Assiette (sévérité = 0.72)

FIGURE 7.10 – Comparaison d'un vol à risque et des profils-types pour quatre paramètres de vol influents. Les graphiques sont issus de l'interface utilisateur de *FlightScanner* pour les données `Reg2`. Le point de toucher des trains est représenté à $t = 60$ secondes.

Chapitre 8

Conclusion

8.1 Résumé des travaux

De nouvelles réglementations imposent désormais aux compagnies aériennes d'établir une stratégie de gestion des risques pour réduire encore davantage le nombre d'accidents. Les données des enregistreurs de vol, très peu exploitées à ce jour, doivent être analysées de façon systématique pour identifier, mesurer et suivre l'évolution des risques.

L'objectif de cette thèse était de proposer un ensemble d'outils méthodologiques pour répondre à la problématique de l'analyse des données de vol. Nos travaux se sont articulés autour de deux thèmes statistiques : la sélection de variables en apprentissage supervisé d'une part et l'analyse des données fonctionnelles d'autre part. Les méthodes proposées ont été appliquées aux risques d'atterrissage long et d'atterrissage dur, deux questions importantes pour les compagnies aériennes.

Du fait de la complexité des données, des méthodes d'apprentissage non paramétriques ont été privilégiées. Nous avons utilisé en particulier l'algorithme des forêts aléatoires pour analyser les données de vol dans les Chapitres 2, 4, 5 et 6. Très performantes en pratique, les forêts aléatoires intègrent des mesures d'importance qui peuvent être utilisées dans des procédures de sélection de variables.

La mesure d'importance par permutation a été étudiée dans le Chapitre 4 dans un contexte où les variables sont corrélées. Ce critère nous a permis de proposer une première méthode de sélection des paramètres de vol dans le cadre fonctionnel. Dans le Chapitre 5, nous avons étendu cette approche en introduisant un nouveau critère d'importance pour des groupes de variables. Sélectionner les paramètres de vol revient alors à sélectionner les groupes formés par les coefficients de base correspondants (les ondelettes par exemple). Afin de réduire la dimension des données et projeter les courbes sur une base commune, il a été nécessaire d'adapter la méthode de seuillage par ondelettes pour n processus indépendants (Chapitres 3 et 5).

Enfin, le Chapitre 7 a décrit l'intégration des outils proposés dans ce manuscrit dans le logiciel *FlightScanner* développé par Safety Line. Destiné aux gestionnaires de la sécurité des compagnies aériennes, cette nouvelle approche de la gestion des risques constitue un aboutissement de ces travaux de thèse. Les interfaces utilisateurs sont présentées ainsi que les différents indicateurs et représentations graphiques permettant aux opérationnels de mieux suivre les risques en exploitation. Une application au problème de l'atterrissage dur a finalement été proposée pour illustrer l'utilisation du logiciel.

8.2 Perspectives

Dans le Chapitre 2, nous avons établi un lien entre la mesure d'importance par permutation des forêts aléatoires et les indices de sensibilité. Ce résultat nous permet de considérer les forêts aléatoires comme une nouvelle méthode d'estimation des indices de Sobol. Pour compléter ce travail, plusieurs axes de travail sont envisagés. Tout d'abord, une étude de simulation est nécessaire pour comparer les forêts aléatoires et les méthodes d'estimation classiques par Monte Carlo. Deux cas pourront notamment être considérés :

1. Lorsque le code de calcul est coûteux, notre objectif est de valider le fait que les forêts aléatoires sont plus rapides que les méthodes de Monte Carlo.

2. Dans le cas inverse, nous souhaitons montrer que les méthodes classiques (Monte Carlo) fournissent une meilleure estimation des indices de Sobol.

Nous envisageons également de généraliser notre résultat au cas où les paramètres d'entrée ne sont pas indépendants en nous basant sur les travaux de Da Veiga et al. (2009); Mara and Tarantola (2012); Chastaing et al. (2012). L'objectif serait d'étudier l'effet de la corrélation sur les indices de Sobol comme nous l'avons fait pour la mesure d'importance par permutation dans le Chapitre 4.

Un deuxième axe de recherche concerne la méthode de réduction de dimension proposée dans les chapitres 3 et 5. Nous avons proposé une méthode de seuillage par ondelettes pour n processus indépendants ainsi qu'un résultat élémentaire de consistance. Or il est connu que dans le cadre gaussien, seuiller les coefficients d'ondelettes par la méthode de seuillage dur (Donoho and Johnstone; 1994) est équivalent à les sélectionner par un critère pénalisé de type ℓ_0 (Massart; 2003). Un travail futur serait donc de reconsidérer notre méthode de réduction de dimension avec ce second point de vue. Il serait en effet possible d'appliquer les résultats généraux obtenus par Birgé and Massart (2001) pour sélectionner de façon simultanée les coefficients d'ondelettes par une procédure pénalisée ℓ_0. Cette approche aurait un double avantage. Sur le plan théorique, une telle procédure de sélection serait garantie par une inégalité de type "oracle". D'un point de vue computationnel, plutôt que d'estimer σ par la méthode MAD, nous utiliserions l'heuristique de pente (Birgé and Massart; 2006; Baudry et al.; 2012) pour calibrer la constante de pénalité. Ces travaux futurs pourront alors fournir une méthode automatique de réduction de dimension des données de vol.

Par ailleurs, les travaux ont soulevé plusieurs points. Tout d'abord, la procédure de sélection *backward* utilisée avec les forêts aléatoire est assez instable. Nous avons donc répété la procédure et avons agrégé les différentes sélections suivant les idées de Meinshausen and Bühlmann (2010). Les paramètres de vol finalement choisis sont ceux dont la fréquence de sélection est la plus élevée. D'un point de vue explicatif, choisir les variables par leur fréquence de sélection pose plusieurs difficultés méthodologiques. D'une part, le choix du nombre de variables à sélectionner est fait *a priori* et n'est pas optimal au sens de l'erreur de prédiction. D'autre part, la façon dont les variables apparaissent dans les différentes sélections n'est pas prise en compte. Un axe de recherche consisterait à améliorer l'agrégation des différentes sélections afin d'obtenir une liste de paramètres de vol stable en nous inspirant des nombreux travaux existants, par exemple ceux de Kalousis et al. (2007); He and Yu (2010); Haury et al. (2011); Shah and Samworth (2013). Nous pourrions également améliorer la stabilité et l'interprétation de la sélection des paramètres de vol en injectant les connaissances que l'on a sur les données. Cette approche est notamment étudiée par Haury et al. (2010).

Les temps de calculs ont également été un problème rencontré couramment dans le traitement des données de vol. Dans le contexte de la thèse, nous avons étudié des échantillons de taille raisonnable pour pallier cette difficulté. Le volume des données reçues par Safety Line est cependant très important (plusieurs mois d'exploitation). Un objectif futur sera donc d'adapter les outils que nous avons proposé à un contexte de calcul distribué.

Annexes A

Glossaire

A.1 Glossaire des termes aéronautiques

Conformité : État de ce qui est conforme à un ensemble de normes.

CVR : Cockpit Voice Recorder.

FDM : Flight Data Monitoring ou *Analyse des vols*.

FDR : Flight Data Recorder ou enregistreurs de paramètres.

IATA : Association internationale du transport aérien.

Incident : événement autre qu'un accident, lié à l'utilisation d'un aéronef qui compromet ou pourrait compromettre la sécurité de l'exploitation.

Incident grave : incident dont les circonstances indiquent qu'un accident a failli se produire.

OACI : Organisation de l'Aviation Civile Internationale.

Palonnier : dispositif destiné à actionner la gouverne de direction de l'avion.

Plan de descente (glide slope) : pente nominale d'approche de 3 degrés.

SGS : Système de Gestion de la Sécurité.

Tube Pitot : instrument mesurant la pression totale de l'air pour, notamment, calculer la vitesse de l'avion.

A.2 Glossaire des unités de mesure

g : unité de mesure d'accélération. Un g est égal à 9.81 $m.s^{-2}$.

Nœud : unité de mesure de vitesse. Un nœud est égal à 0.514 $m.s^{-1}$.

Pied : unité de mesure d'altitude. Un pied est égal à 0.3048 mètres.

A.3 Glossaire des paramètres de vol

AILL, AILR : position des ailerons.

ALT_STDC : altitude barométrique en pieds.

AOA : angle d'attaque (Angle Of Attack).

CASC : vitesse air de l'avion (Calibrated Airspeed) en nœuds.

CASC_GSC : vitesse du vent en nœuds. Elle est calculée par la différence entre la vitesse air et la vitesse sol.

DRIFT : angle de dérive, écart entre le cap suivi par l'avion et le cap initialement fixé.

ELEVL, ELEVR : position des gouvernes de profondeur.

FLAPC : position des volets.

FQTYC : quantité de fuel (Total Fuel Quantity).

GLIDE_DEVC : écart au plan de descente.

GSC : vitesse de déplacement par rapport au référentiel terrestre (Ground Speed). L'unité est le nœud.

GW_KG : masse de l'avion en kilogrammes (Gross Weight).

HEAD_MAG : cap magnétique, angle entre le nord magnétique et l'axe longitudinal de l'avion (Magnetic Heading).

HEIGHT : hauteur en pieds.

IAS : vitesse indiquée en nœuds (Indicated airspeed).

IVV : vitesse verticale en pieds par minute (Inertial vertical speed).

LDG : toucher des trains (Landing Gear).

LATG : accélération latérale en g.

LATP : lattitude en degré décimal.

LDGL, LDGR : indicateur de sortie des trains d'atterrissage.

LONGG : accélération longitudinale en g.

LONP : longitude en degré décimal.

N1, N2 : régimes moteur en pourcentage de vitesse de rotation.

PITCH : assiette, rotation de l'avion autour de l'axe latéral. L'unité est le degré.

RALTC : radio altitude en pieds.

ROLL : roulis, rotation de l'avion autour de l'axe longitudinal. L'unité est le degré.

RUDDER : position de la gouverne de direction.

SAT : température statique de l'air (Static Air Temperature).

SLATS : position des becs.

TLA : commande de poussée des moteurs (Throttle Lever Angle).

VAPP : vitesse d'approche en nœuds.

VRTG : accélération verticale en g.

WIN_SHR_WAR : alarme de cisaillement de vent (Wind Shear Warning).

Références bibliographiques

Akaike, H. (1974). A new look at the statistical model identification, *IEEE Transactions on Automatic Control* **19** : 716–723.

Alonso, A. M., Casado, D. and Romo, J. (2012). Supervised classification for functional data : A weighted distance approach, *Computational Statistics and Data Analysis* **56**(7) : 2334–2346.

Altmann, A., Toloşi, L., Sander, O. and Lengauer, T. (2010). Permutation importance : a corrected feature importance measure, *Bioinformatics* **26**(10) : 1340–1347.

Amato, U., Antoniadis, A. and De Feis, I. (2006). Dimension reduction in functional regression with applications, *Computational Statistics and Data Analysis* **50** : 2422–2446.

Ambroise, C. and McLachlan, G. J. (2002). Selection bias in gene extraction on the basis of microarray gene-expression data, *Proceedings of the National Academy of Sciences* **99** : 6562–6566.

Aminghafari, M., Cheze, N. and Poggi, J.-M. (2006). Multivariate denoising using wavelets and principal component analysis, *Computational Statistics and Data Analysis* **50** : 2381–2398.

Amini, A. A., Levina, E. and Shedden, K. A. (2013). Structured functional regression models for high-dimensional spatial spectroscopy data. arXiv :1311.0416.

Andrieu, C. (2013). *Modélisation fonctionnelle de profils de vitesse en lien avec l'infrastructure et méthodologie de construction d'un profil agrégé*, PhD thesis, Université Toulouse III Paul Sabatier.

Antoniadis, A., Bigot, J. and Sapatinas, T. (2001). Wavelet estimators in nonparametric regression : A comparative simulation study, *Journal of Statistical Software* pp. 1–83.

Antoniadis, A., Brossat, X., Cugliari, J. and Poggi, J.-M. (2013). Clustering functional data using wavelets, *International Journal of Wavelets, Multiresolution and Information Processing* **11**.

Antoniadis, A., Brossat, X., Cugliari, J. and Poggi, J.-M. (2014). Une approche fonctionnelle pour la prévision non-paramétrique de la consommation d'électricité, *Journal de la Société Française de Statistique* **155**(2) : 202–219.

Araki, Y., Konishi, S., Kawano, S. and Matsui, H. (2009). Functional logistic discrimination via regularized basis expansions, *Communication in Statistics, Theory and Methods* **38** : 2944–2957.

Archer, K. J. and Kimes, R. V. (2008). Empirical characterization of random forest variable importance measures, *Computational Statistics and Data Analysis* **52**(4) : 2249–2260.

Aronszajn, N. (1950). Theory of reproducing kernels, *Transactions of the American Mathematical Society* **68**(3) : 337–404.

Auder, B., de Crecy, A., Iooss, B. and Marquès, M. (2012). Screening and metamodeling of computer experiments with functional outputs. application to thermal-hydraulic computations, *Reliability Engineering and System Safety* **107** : 122–131.

Auret, L. and Aldrich, C. (2011). Empirical comparison of tree ensemble variable importance measures, *Chemometrics and Intelligent Laboratory Systems* **105** : 157–170.

Bach, F. R. (2008). Bolasso : model consistent lasso estimation through the bootstrap, *Proceedings of the 25th international conference on Machine learning*, ICML '08, pp. 33–40.

Bahlmann, C., Haasdonk, B. and Burkhardt, H. (2002). On-line handwriting recognition with support vector machines - a kernel approach, *In Proceeding of the 8th IWFHR*.

Baudry, J.-P., Maugis, C. and Michel, B. (2012). Slope heuristics : overview and implementation., *Statistics and Computing* **22**(1) : 455–470.

Berlinet, A., Biau, G. and Rouvière, L. (2008). Functional supervised classification with wavelets, *Annales de l'ISUP* **52** : 61–80.

Besse, P. and Cardot, H. (2003). Modélisation statistique de données fonctionnelles, *in* G. Govaert (ed.), *Analyse des données*, Hermes Science-Lavoisier, pp. 167–198.

Bi, J., Bennett, K. P., Embrechts, M., Brenemanand, C. M. and Song, M. (2003). Dimensionality reduction via sparse support vector machines, *Journal of Machine Learning Research* **3** : 1229–1243.

Biau, G. (2012). Analysis of a random forests model, *Journal of Machine Learning Research* **13** : 1063–1095.

Biau, G., Bunea, F. and Wegkamp, M. (2005). Functional classification in hilbert spaces, *IEEE Transactions on Information Theory* **51** : 2163–2172.

Biau, G., Devroye, L. and Lugosi, G. (2008). Consistency of random forests and other averaging classifiers, *Journal of Machine Learning Research* **9** : 2015–2033.

Birgé, L. and Massart, P. (2001). Gaussian model selection, *Journal of the European Mathematical Society* **3** : 203–268.

Birgé, L. and Massart, P. (2006). Minimal penalties for gaussian model selection, *Probability Theory and Related Fields* **138**(1-2) : 33–73.

Blum, A. L. and Langley, P. (1997). Selection of relevant features and examples in machine learning, *Artificial Intelligence* **97** : 245–271.

Boser, B. E., Guyon, I. and Vapnik, V. (1992). A training algorithm for optimal margin classifiers, *Proceedings of the fifth annual workshop on Computational learning theory*, COLT '92, pp. 144–152.

Boyd, S. and Vandenberghe, L. (2004). *Convex optimization*, Cambridge Univ Pr.

Breiman, L. (1996). Bagging predictors, *Machine Learning* **24** : 123–140.

Breiman, L. (1997). Out-of-bag estimation, *Technical report*, Department of Statistics, University of Berkeley.

Breiman, L. (2001). Random forests, *Machine Learning* **45**(1) : 5–32.

Breiman, L. (2004). Consistency for a simple model of random forests, *Technical report*, Department of Statistics, University of Berkeley.

Breiman, L., Friedman, J. H., Olshen, R. A. and Stone, C. J. (1984). *Classification and regression trees*, Wadsworth Advanced Books and Software.

Bühlmann, P., Rütimann, P., van de Geer, S. and Zhang, C.-H. (2013). Correlated variables in regression : clustering and sparse estimation, *Journal of Statistical Planning and Inference* **143**(11) : 1835–1858.

Cai, T. and Hall, P. (2006). Prediction in functional linear regression, *The Annals of Statistics* **34**(5) : 2159–2179.

Cai, T. and Zhou, H. (2009). A data-driven block thresholding approach to wavelet estimation, *The Annals of Statistics* **37** : 569–595.

Cardot, H., Ferraty, F. and Sarda, P. (1999). Functional linear model, *Statistics and Probability Letters* **45** : 11–22.

Cardot, H., Ferraty, F. and Sarda, P. (2003). Spline estimators for the functional linear model, *Statistica Sinica* **13** : 571–592.

Cardot, H. and Sarda, P. (2005). Estimation in generalized linear models for functional data via penalized likelihood, *Journal of Multivariate Analysis* **92**(1) : 24–41.

Chakraborty, D. and Pal, N. R. (2008). Selecting useful groups of features in a connectionist framework, *IEEE Transactions on Neural Networks* **19**(3) : 381–396.

Chastaing, G. (2013). *Indices de Sobol généralisés pour variables dépendantes*, PhD thesis, Université de Grenoble.

Chastaing, G., Gamboa, F. and Prieur, C. (2012). Generalized hoeffding-sobol decomposition for dependent variables – application to sensitivity analysis, *Electronic Journal of Statistics* **6** : 2420–2448.

Chawla, N. V., Bowyer, K. W., Hall, L. O. and Kegelmeyer, W. P. (2002). Smote : Synthetic minority over-sampling technique, *Journal of Artificial Intelligence Research* **16** : 321–357.

Chen, C., Liaw, A. and Breiman, L. (2004). Using random forest to learn imbalanced data, *Technical report*, Department of Statistics, University of Berkeley.

Chen, S. S., Donoho, D. L. and Saunders, M. A. (1998). Atomic decomposition by basis pursuit, *SIAM Journal on Scientific Computing* **20**(1) : 33–61.

Cortes, C. and Vapnik, V. (1995). Support-vector networks, *Machine Learning*, pp. 273–297.

Cristianini, N. and Shawe-Taylor, J. (2000). *An introduction to support Vector Machines : and other kernel-based learning methods*, Cambridge University Press.

Cuevas, A., Febrero, M. and Fraiman (2007). Robust estimation and classification for functional data via projection-based depth notions, *Computational Statistics* **22**(3) : 481–496.

Da Veiga, S., Wahl, F. and Gamboa, F. (2009). Local polynomial estimation for sensitivity analysis on models with correlated inputs, *Technometrics* **51**(4) : 452–463.

Daubechies, I. (1992). *Ten Lectures on Wavelets*, number 61 in *CBMS/NSF Series in Applied Mathematics*, siam.

Dauxois, J. and Pousse, A. (1976). Les analyses factorielles en calcul des probabilités et en statistique : essai d'étude synthétique. Thèse d'état, Université Paul Sabatier, Toulouse.

Dauxois, J., Pousse, A. and Romain, Y. (1982). Asymptotic theory for the principal component analysis of a vector random function : Some applications to statistical inference, *Journal of Multivariate Analysis* **12** : 136–154.

de Rocquigny, E. (2006). La maîtrise des incertitudes dans un contexte industriel – 1re partie : une approche méthodologique globale basée sur les exemples, *Journal de la Société Française de Statistique* **147**(3) : 33–71.

de Rocquigny, E., Devictor, N. and Tarantola, S. (2008). *Uncertainty in Industrial Practice : A Guide to Quantitative Uncertainty Management*, Wiley.

Dean, J. and Ghemawat, S. (2008). Mapreduce : simplified data processing on large clusters, *Communications of the ACM* **51**(1) : 107–113.

Deville, J.-C. (1974). Méthodes statistiques et numériques de l'analyse harmonique, *Annales de l'insee* (15) : 5–101.

Devroye, L., Györfi, L. and Lugosi, G. (1996). *A probabilistic theory of pattern recognition*, Springer.

Díaz-Uriarte, R. and Alvarez de Andrés, S. (2006). Gene selection and classification of microarray data using random forest, *BMC Bioinformatics* **7** : 3.

Donoho, D. L. and Johnstone, I. M. (1994). Ideal spatial adaptation by wavelet shrinkage, *Biometrika* **81** : 425–455.

Donoho, D. L. and Johnstone, I. M. (1995). Adapting to unknown smoothness via wavelet shrinkage, *Journal of the American Statistical Association* **90**(432) : 1200–1224.

Donoho, D. L. and Johnstone, I. M. (1998). Minimax estimation via wavelet shrinkage, *The Annals of Statistics* **26** : 879–921.

Donoho, D. L., Johnstone, I. M., Kerkyacharian, G. and Picard, D. (1995). Wavelet shrinkage : asymptopia, *Journal of the Royal Statistical Society, Series B* **57** : 301–369.

Donoho, D. L., Johnstone, I. M., Kerkyacharian, G. and Picard, D. (1996). Density estimation by wavelet thresholding, *The Annals of Statistics* **24** : 508–539.

Drucker, H., Burges, C., Kaufman, L., Smola, A. and Vapnik, V. (1996). Support vector regression machines, *Advances in Neural Information Processing Systems*.

Duan, K.-B., Rajapakse, J., Wang, H. and Azuaje, F. (2005). Multiple svm-rfe for gene selection in cancer classification with expression data, *NanoBioscience, IEEE Transactions on* **4** : 228–234.

Evgeniou, T., Pontil, M. and Poggio, T. (2000). Regularization networks and support vector machines, *Advances in Computational Mathematics*, MIT Press, pp. 1–50.

Fan, Y. and James, G. (2013). Functional additive regression. Preprint.

Ferraty, F. (ed.) (2011). *Recent Advances in Functional Data Analysis and Related Topics*, Springer-Verlag.

Ferraty, F. and Vieu, P. (2002). The functional nonparametric model and application to spectrometric data, *Computational Statistics and Data Analysis* **17** : 545–564.

Ferraty, F. and Vieu, P. (2003). Curves discrimination : a nonparametric functional approach, *Computational Statistics and Data Analysis* **44** : 161–173.

Ferraty, F. and Vieu, P. (2006). *Nonparametric Functional Data Analysis : Theory and Practice (Springer Series in Statistics)*, Springer-Verlag New York, Inc.

Ferraty, F. and Vieu, P. (2009). Additive prediction and boosting for functional data, *Computational Statistics and Data Analysis* **53** : 1400–1413.

Fix, E. and Hodges, J. L. (1951). Discriminatory analysis, nonparametric discrimination : Consistency properties, *US Air Force School of Aviation Medicine* **Technical Report 4** : 477+.

Friedman, J., Hastie, T. and Tibshirani, R. (2010). A note on the group lasso and a sparse group lasso. arXiv :1001.0736.

Fromont, M. and Tuleau, C. (2006). Functional classification with margin conditions, *19th Annual Conference on Learning Theory*.

Genuer, R. (2012). Variance reduction in purely random forests, *Journal of Nonparametric Statistics* **24** : 543–562.

Genuer, R., Poggi, J.-M. and Tuleau-Malot, C. (2010). Variable selection using random forests, *Pattern Recognition Letters* **31** : 2225–2236.

Gertheiss, J., Maity, A. and Staicu, A.-M. (2013). Variable selection in generalized functional linear models, *Stat* **2**(1) : 86–101.

Gey, S. (2002). *Bornes de risque, détection de ruptures, boosting : trois thèmes autour de CART en régression*, PhD thesis, Université Paris Sud.

Gey, S. and Nédélec, E. (2005). Model selection for cart regression trees., *IEEE Transactions on Information Theory* **51** : 658–670.

González-Mantegna, W. and Martínez-Calvo, A. (2011). Bootstrap in functional linear regression, *Journal of Statistical Planning and Inference* **141** : 453–461.

Green, P. J. and Silverman, B. W. (1994). *Nonparametric regression and generalized linear models : a roughness penalty approach*, Chapman and Hall.

Gregorutti, B., Michel, B. and Saint Pierre, P. (2014a). Correlation and variable importance in random forests. arXiv :1310.5726.

Gregorutti, B., Michel, B. and Saint Pierre, P. (2014b). Grouped variable importance with random forests and applications to multivariate functional data analysis. arXiv :1411.4170.

Grömping, U. (2009). Variable importance assessment in regression : linear regression versus random forest, *The American Statistician* **63** : 308–319.

Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection, *Journal of Machine Learning Research* **3** : 1157–1182.

Guyon, I., Weston, J., Barnhill, S. and Vapnik, V. (2002). Gene selection for cancer classification using support vector machines, *Machine Learning* **46**(1-3) : 389–422.

Hall, P., Kerkyacharian, G. and Picard, D. (1999). On the minimax optimality of block thresholded wavelet estimators, *Statistica Sinica* **9** : 33–49.

Hapfelmeier, A. and Ulm, K. (2013). A new variable selection approach using random forests, *Computational Statistics and Data Analysis* **60** : 50–69.

Hastie, T. and Tibshirani, R. (1986). Generalized additive models, *Statistical Science* **1**(3) : 297–318.

Hastie, T. and Tibshirani, R. (1998). Classification by pairwise coupling, *The Annals of Statistics* **26**(2) : 451–471.

Hastie, T., Tibshirani, R. and Friedman, J. (2001). *The Elements of Statistical Learning*, Springer New York Inc.

Haury, A.-C., Gestraud, P. and Vert, J.-P. (2011). The influence of feature selection methods on accuracy, stability and interpretability of molecular signatures, *PLoS ONE* **6** : 1–12.

Haury, A.-C., Jacob, L. and Vert, J.-P. (2010). Improving stability and interpretability of gene expression signatures, *Technical report*, arXiv.

He, Z. and Yu, W. (2010). Stable feature selection for biomarker discovery, *Computational biology and chemistry* **34**(4) : 215–225.

Hoeffding, W. (1948). A class of statistics with asymptotically normal distribution, *The Annals of Mathematical Statistics* **19**(3) : 293–325.

Homma, T. and Saltelli, A. (1996). Importance measures in global sensitivity analysis of nonlinear models, *Reliability Engineering and System Safety* **52**(1) : 1–17.

Horváth, L. and Kokoszka, P. (2012). *Inference for Functional Data with Applications*, Springer Series in Statistics, Springer.

Ieva, F., Paganoni, A. M., Pigoli, D. and Vitelli, V. (2012). Multivariate functional clustering for the morphological analysis of electrocardiograph curves, *Journal of the Royal Statistical Society, Series C* **62** : 401–418.

Iooss, B. and Lemaître, P. (2014). A review on global sensitivity analysis methods. arXiv :1404.2405.

Iooss, B., Van Dorpe, F. and Devictor, N. (2006). Response surfaces and sensitivity analyses for an environmental model of dose calculations, *Reliability Engineering and System Safety* **91** : 1241–1251.

Ishwaran, H. (2007). Variable importance in binary regression trees and forests, *Electronic Journal of Statistics* **1** : 519–537.

Jacob, L., Obozinski, G. and Vert, J.-P. (2009). Group lasso with overlap and graph lasso, *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pp. 433–440.

James, G. M., Hastie, T. and Sugar, C. A. (2000). Principal component models for sparse functional data, *Biometrika* **87** : 587–602.

Janitza, S., Strobl, C. and Boulesteix, A.-L. (2013). An auc-based permutation variable importance measure for random forests, *BMC Bioinformatics* **14** : 119.

Jiang, H., Deng, Y., Chen, H.-S., Tao, L., Sha, Q., Chen, J., Tsai, C.-J. and Zhang, S. (2004). Joint analysis of two microarray gene-expression data sets to select lung adenocarcinoma marker genes, *BMC Bioinformatics* **5** : 81.

Johnstone, I. M. and Silverman, B. W. (1997). Wavelet threshold estimators for data with correlated noise, *Journal of the Royal Statistical Society, Series B* **59**(2) : 319–351.

Kalousis, A., Prados, J. and Hilario, M. (2007). Stability of feature selection algorithms : a study on high-dimensional spaces, *Knowledge and Information Systems* **12** : 95–116.

Kohavi, R. and John, G. H. (1997). Wrappers for feature subset selection, *Artificial Intelligence* **97** : 273–324.

Křížek, P., Kittler, J. and Hlaváč, V. (2007). Improving stability of feature selection methods, *Computer Analysis of Images and Patterns*, Springer Berlin Heidelberg, pp. 929–936.

Kwemou, M. (2012). Non-asymptotic oracle inequalities for the lasso and group lasso in high dimensional logistic model. arXiv :1206.0710.

Laloë, T. (2008). A k-nearest neighbor approach for functional regression, *Statistics and Probability Letters* **78** : 1189–1193.

Laurent, B. and Massart, P. (2000). Adaptive estimation of a quadratic functional by model selection, *The Annals of Statistics* **28** : 1245–1501.

Lazar, C., Taminau, J., Meganck, S., Steenhoff, D., Coletta, A., Molter, C., de Schaetzen, V., Duque, R., Bersini, H. and Nowe, A. (2012). A survey on filter techniques for feature selection in gene expression microarray analysis, *IEEE/ACM Transansaction on Computation Biology and Bioinformatics* **9** : 1106–1119.

Leng, X. and Müller, H.-G. (2006). Classification using functional data analysis for temporal gene expression data, *Bioinformatics* **22** : 68–76.

López-Pintado, S. and Romo, J. (2006). Depth-based classification for functional data, *DIMACS Series in Discrete Mathematics and Theoretical Computer Science* **72** : 103.

Louw, N. and Steel, S. (2006). Variable selection in kernel fisher discriminant analysis by means of recursive feature elimination, *Computational Statistics and Data Analysis* **51** : 2043–2055.

Mallat, S. (1989). A theory for multiresolution signal decomposition : the wavelet representation, *Pattern Analysis and Machine Intelligence, IEEE Transactions on* **11**(7) : 674–693.

Mallat, S. (2000). *Une exploration des signaux en ondelettes*, Éditions de l'École Polytechnique.

Mallat, S. (2008). *A Wavelet Tour of Signal Processing*, 3rd edn, Academic Press.

Mallows, C. L. (1973). Some comments on cp, *Technometrics* pp. 661–675.

Mara, T. and Tarantola, S. (2012). Variance-based sensitivity indices for models with dependent inputs, *Reliability Engineering & System Safety* **107** : 115–121.

Massart, P. (2003). *Concentration inequalities and model selection*, Springer.

Matsui, H. (2014). Variable and boundary selection for functional data via multiclass logistic regression modeling, *Computational Statistics and Data Analysis* **78**(0) : 176–185.

Matsui, H. and Konishi (2011). Variable selection for functional regression models via the regularization, *Computational Statistics and Data Analysis* **55**(12) : 3304–3310.

Meier, L., Van De Geer, S. and Bühlmann, P. (2008). The group lasso for logistic regression, *Journal of the Royal Statistical Society, Series B* **70** : 53–71.

Meinshausen, N. and Bühlmann, P. (2010). Stability selection, *Journal of the Royal Statistical Society, Series B* **72** : 417–473.

Mercer, J. (1909). Functions of positive and negative type, and their connection with the theory of integral equations, *Philosophical Transactions of the Royal Society, London* **209** : 415–446.

Misiti, M., Misiti, Y., Oppenheim, G. and Poggi, J.-M. (1994). Décomposition par ondelettes et méthodes comparatives : étude d'une courbe de charge électrique, *Revue de Statistique Appliquée* **42**(2) : 55–77.

Mostacci, E., Truntzer, C., Cardot, H. and Ducoroy, P. (2010). Multivariate denoising methods combining wavelets and principal component analysis for mass spectrometry data, *Proteomics* **10** : 2564–2572.

Neville, P. G. (2013). Controversy of variable importance in random forests, *Journal of Unified Statistical Techniques* **1** : 15–20.

Nicodemus, K. K. (2011). Letter to the editor : On the stability and ranking of predictors from random forest variable importance measures, *Briefings in Bioinformatics* **12** : 369–373.

Nicodemus, K. K. and Malley, J. D. (2009). Predictor correlation impacts machine learning algorithms : implications for genomic studies, *Bioinformatics* **25** : 1884–1890.

Nicodemus, K. K., Malley, J. D., Strobl, C. and Ziegler, A. (2010). The behaviour of random forest permutation-based variable importance measures under predictor correlation, *BMC Bioinformatics* **11** : 110.

Obozinski, G., Jacob, L. and Vert, J.-P. (2009). Group lasso with overlaps : the latent group lasso approach. arXiv :1110.0413.

Oliva, J. B., Poczos, B., Verstynen, T., Singh, A., Schneider, J., Yeh, F.-C. and Tseng, W.-Y. (2014). Fusso : Functional shrinkage and selection operator, *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics, AISTATS 2014, Reykjavik, Iceland, April 22-25, 2014*, pp. 715–723.

Osuna, E. E., Freund, R. and Girosi, F. (1997). Support vector machines : Training and applications, *Technical report*, Massachusetts Institute of Technology.

Percival, D. B. and Walden, A. T. (2000). *Wavelet Methods for Time Series Analysis*, Cambridge University Press.

Pigoli, D. and Sangalli, L. M. (2012). Wavelets in functional data analysis : Estimation of multidimensional curves and their derivatives, *Computational Statistics and Data Analysis* **56** : 1482–1498.

Platt, J. C. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods, *Advances in Large Margin Classifiers*, MIT Press, pp. 61–74.

Poggi, J.-M. and Tuleau, C. (2006). Classification supervisée en grande dimension. application à l'agrément de conduite automobile, *Revue de Statistique Appliquée* **4** : 39–58.

Rakotomamonjy, A. (2003). Variable selection using svm based criteria, *Journal of Machine Learning Research* **3** : 1357–1370.

Ramsay, J. O. and Silverman, B. W. (1997). *Functional Data Analysis*, Springer Series in Statistics Series, Springer-Verlag GmbH.

Ramsay, J. O. and Silverman, B. W. (2002). *Applied Functional Data Analysis : Methods and Case Studies*, Springer.

Ramsay, J. O. and Silverman, B. W. (2005). *Functional Data Analysis*, Springer Series in Statistics, Springer.

Randolph, T. W., Harezlak, J. and Feng, Z. (2012). Structured penalties for functional linear models – partially empirical eigenvectors for regression, *Electronic Journal of Statistics* **6** : 323–353.

Rao, C. R. (1973). *Linear statistical inference and its applications*, Wiley series in probability and mathematical statistics : Probability and mathematical statistics, Wiley.

Rossi, F., François, D., Wertz, V. and Verleysen, M. (2006). A functional approach to variable selection in spectrometric problems, *Proceedings of 16th International Conference on Artificial Neural Networks, ICANN 2006*, pp. 11–20.

Rossi, F. and Villa, N. (2006). Support vector machine for functional data classification, *Neurocomputing* **69** : 730–742.

Rossi, F. and Villa, N. (2008). Recent advances in the use of svm for functional data classification, *Proceedings of 1rst International Workshop on Functional and Operatorial Statistics, IWFOS*.

Saltelli, A. (2002). Making best use of model evaluations to compute sensitivity indices, *Computer Physics Communications* **145**(2) : 280–297.

Saltelli, A., Chan, K. and Scott, E. (2009). *Sensitivity Analysis*, Wiley paperback series, Wiley.

Saunders, C., Gammerman, A. and Vovk, V. (1998). Ridge regression learning algorithm in dual variables, *In Proceedings of the 15th International Conference on Machine Learning*, pp. 515–521.

Schölkopf, B. and Smola, A. J. (2001). *Learning with Kernels : Support Vector Machines, Regularization, Optimization, and Beyond*, MIT Press.

Schwarz, G. (1978). Estimating the dimension of a model, *The Annals of Statistics* **6** : 461–464.

Scornet, E., Biau, G. and Vert, J.-P. (2014). Consistency of random forests. arXiv :1405.2881.

Shah, R. D. and Samworth, R. J. (2013). Variable selection with error control : another look at stability selection, *Journal of the Royal Statistical Society, Series B* **75**(1) : 55–80.

Shawe-Taylor, J. and Cristianini, N. (2004). *Kernel Methods for Pattern Analysis*, Cambridge University Press, New York, NY, USA.

Shimodaira, H., Noma, K.-I., Nakai, M. and Sagayama, S. (2001). Dynamic time-alignment kernel in support vector machine, *Advances in Neural Information Processing Systems 14*, pp. 921–928.

Simon, N., Friedman, J., Hastie, T. and Tibshirani, R. (2013). A sparse-group lasso, *Journal of Computational and Graphical Statistics* **22**.

Simon, N. and Tibshirani, R. (2012). Standardization and the group lasso penalty, *Statistica Sinica* **22** : 983–1001.

Sobol, I. M. (1993). Sensitivity estimates for non linear mathematical models, *Mathematical Modelling and Computational Experiments* **1** : 407–414.

Song, J. J., Deng, W., Lee, H.-J. and Kwon (2008). Optimal classification for time-course gene expression data using functional data analysis, *Computational Biology and Chemistry* **32**(6) : 426–432.

Stitson, M. O., Gammerman, A., Vapnik, V., Vovk, V., Watkins, C. and Weston, J. (1997). Support vector regression with anova decomposition kernels, *Advances in kernel methods*, pp. 285–292.

Strobl, C., Boulesteix, A.-L., Kneib, T., Augustin, T. and Zeileis, A. (2008). Conditional variable importance for random forests, *BMC Bioinformatics* **9** : 307.

Strobl, C. and Zeileis, A. (2008). Danger : High power ! exploring the statistical properties of a test for random forest variable importance, *Proceedings of the 18th International Conference on Computational Statistics*.

Suykens, J. A. K., De Brabanter, J., Lukas, L. and Vandewalle, J. (2002). Weighted least squares support vector machines : robustness and sparse approximation, *Neurocomputing* **48** : 85–105.

Svetnik, V., Liaw, A., Tong, C. and Wang, T. (2004). Application of breiman's random forest to modeling structure-activity relationships of pharmaceutical molecules, *Multiple Classifier Systems*.

Tian, T. S. and James, G. M. (2013). Interpretable dimension reduction for classifying functional data, *Computational Statistics and Data Analysis* **57** : 282–296.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso, *Journal of the Royal Statistical Society, Series B* **58** : 267–288.

Tikhonov, A. (1963). Solution of incorrectly formulated problems and the regularization method, *Soviet Mathematics Doklady* pp. 1035–1038.

Toloşi, L. and Lengauer, T. (2011). Classification with correlated features : unreliability of feature ranking and solutions, *Bioinformatics* **27** : 1986–1994.

Tuleau, C. (2005). *Sélection de variables pour la discrimination en grande dimension et classification de données fonctionnelles*, PhD thesis, Université Paris Sud.

van de Geer, S. (2008). High-dimensional generalized linear models and the lasso, *The Annals of Statistics* **36**(2) : 614–645.

van der Laan, M. J. (2006). Statistical inference for variable importance, *The International Journal of Biostatistics* **2**(1) : 1–33.

Vapnik, V. N. (1995). *The nature of statistical learning theory*, Springer-Verlag, New York.

Vapnik, V. N. (1998). *Statistical Learning Theory*, Wiley-Interscience.

Veropoulos, K., Campbell, C. and Cristianini, N. (1999). Controlling the sensitivity of support vector machines, *Proceedings of the International Joint Conference on AI*, pp. 55–60.

Volkova, E., Iooss, B. and Van Dorpe, F. (2008). Global sensitivity analysis for a numerical model of radionuclide migration from the rrc "kurchatov institute" radwaste disposal site, *Stochastic Environmental Research and Risk Assessment* **22**(1) : 17–31.

Wasserman, L. (2006). *All of Nonparametric Statistics*, Springer New York.

Xiang, S., Tong, X. and Ye, J. (2013). Efficient sparse group feature selection via nonconvex optimization., *Proceedings of the 30th International Conference on Machine Learning*, Vol. 28 of *JMLR Proceedings*, pp. 284–292.

Yan, R., Liu, Y., Jin, R. and Hauptmann, A. (2003). On predicting rare classes with svm ensembles in scene classification, *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 21–24.

Yang, K., Yoon, H. and Shahabi, C. (2005). A supervised feature subset selection technique for multivariate time series, *Proceedings of the Workshop on Feature Selection for Data Mining : Interfacing Machine Learning with Statistics*.

Yoon, H. and Shahabi, C. (2006). Feature subset selection on multivariate time series with extremely large spatial features, *Proceedings of the Sixth IEEE International Conference on Data Mining - Workshops*, ICDMW '06, pp. 337–342.

Yuan, M. and Lin, Y. (2006a). Model selection and estimation in regression with grouped variables, *Journal of the Royal Statistical Society, Series B* **68** : 49–67.

Yuan, M. and Lin, Y. (2006b). Model selection and estimation in regression with grouped variables, *Journal of the Royal Statistical Society, Series B* **68** : 49–67.

Zhang, H. H., Liu, Y., Wu, Y., Zhu, J. et al. (2008). Variable selection for the multicategory svm via adaptive sup-norm regularization, *Electronic Journal of Statistics* **2** : 149–167.

Zhao, P., Rocha, G. and Yu, B. (2009). The composite absolute penalties family for grouped and hierarchical variable selection, *The Annals of Statistics* **37**(4A) : 3468–3497.

Zhou, N. and Zhu, J. (2010). Group variable selection via a hierarchical lasso and its oracle property, *Statistics and Its Interface* **3** : 557–574.

Zhu, H. and Cox, D. D. (2009). A functional generalized linear model with curve selection in cervical pre-cancer diagnosis using fluorescence spectroscopy, *Lecture Notes-Monograph Series* pp. 173–189.

Zhu, J. and Hastie, T. (2004). Classification of gene microarrays by penalized logistic regression, *Biostatistics* **5**(14) : 427–443.

Zhu, J., Rosset, S., Hastie, T. and Tibshirani, R. (2004). 1-norm support vector machines, *in* S. Thrun, L. Saul and B. Schölkopf (eds), *Advances in Neural Information Processing Systems 16*, MIT Press.

Zhu, R., Zeng, D. and Kosorok, M. R. (2012). Reinforcement learning trees, *Technical report*, University of North Carolina.

Zou, H. (2006). The adaptive lasso and its oracle properties, *Journal of the American Statistical Association* **101**(476) : 1418–1429.

Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net, *Journal of the Royal Statistical Society, Series B* **67**(2) : 301–320.