# Outils statistiques de traitement d'indicateurs pour le diagnostic et le pronostic des moteurs d'avions

Tsirizo Rabenoro

▶ **To cite this version:**

Tsirizo Rabenoro. Outils statistiques de traitement d'indicateurs pour le diagnostic et le pronostic des moteurs d'avions. Machine Learning [stat.ML]. Université Paris 1 Panthéon Sorbonne, 2015. Français. NNT : . tel-01225739

**HAL Id: tel-01225739**
**https://hal.science/tel-01225739v1**

Submitted on 6 Nov 2015

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

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

SAFRAN Snecma


UNIVERSITÉ PARIS 1 - PANTHÉON SORBONNE


# THÈSE DE DOCTORAT

Discipline : Mathématiques appliquées

présentée par

Tsirizo RABENORO

---

# Outils statistiques de traitement d'indicateurs pour le diagnostic et le pronostic des moteurs d'avions

---

sous la direction de Fabrice Rossi et Marie Cottrell

soutenue le 18 septembre 2015

Composition du jury :

| | | |
|---|---|---|
| M. Allou SAME | IFSTTAR | (Rapporteur) |
| M. Michel VERLEYSEN | Université catholique de Louvain | (Rapporteur) |
| M. Serge BLANCHARD | SAFRAN Snecma | (Invité) |
| M. Jérôme LACAILLE | SAFRAN Snecma | (Examinateur) |
| M. Ludovic DENOYER | Université Paris 6 Pierre et Marie Curie | (Examinateur) |
| Mme. Marie COTTRELL | Université Paris 1 Panthéon Sorbonne | (Directeur) |
| M. Fabrice ROSSI | Université Paris 1 Panthéon Sorbonne | (Directeur) |

# Résumé


## Résumé

Détecter les signes d'anomalies dans un système complexe est l'un des principaux objectifs de la maintenance préventive dans l'industrie. Cela permet d'éviter une défaillance ou de limiter les dégradations d'un composant en avançant une opération de maintenance. Le *Health Monitoring* des moteurs d'avions fait partie des domaines industriels pour lesquels cette détection d'anomalies est un enjeu fort. Ainsi, les motoristes, tels que Snecma, collectent de grandes quantités de données relatives au moteur durant chaque vol. Il s'agit de détecter automatiquement, à partir de ces données, les cas où un moteur dévie de son comportement normal. Plus précisément, Snecma développe des applications permettant de prévenir les pannes moteurs en détectant les anomalies.

Cette thèse présente comment le savoir des experts de Snecma est exploité pour traiter ces données moteurs. Ce premier travail a permis de mettre en avant les difficultés liées aux traitements des données : qu'il s'agisse des difficultés concernant le stockage des données ou bien des difficultés liées à la définition des algorithmes de traitement eux-mêmes. Ensuite, la thèse propose une méthodologie permettant de combiner le savoir expert à des méthodes d'apprentissage automatique tout en respectant les exigences d'un motoriste tel que Snecma. Parmi celles-ci, on peut citer le besoin de fusionner des informations variées, le contrôle des erreurs et l'interprétabilité des résultats de diagnostic.

Pour cela, la méthodologie exploite directement les données issues des algorithmes de traitement développées par les experts eux-mêmes. Cela est rendu possible par une nécessaire homogénéisation des données, autrement dit par une mise en forme commune de celles-ci permettant alors de procéder à leur fusion. L'homogénéisation des données rend possible l'utilisation des algorithmes de classification (supervisée) dont le but est de regrouper automatiquement, en classe, les individus (ici les moteurs) de même nature à partir des informations fournies et sans perdre l'information temporelle. L'homogénéisation des données permet également d'exploiter directement les applications de surveillance mises en place par les experts métier pour détecter les anomalies. De cette façon, la méthodologie mise à disposition par la thèse reste compréhensible par les experts métier.

Avant de procéder effectivement à la fusion, un algorithme de sélection de variables est utilisé. La thèse décrit comment le processus de sélection permet une calibration automatique des applications de surveillance développées par les experts métier. De plus, cette sélection permet de répondre en partie à la première exigence de Snecma concernant l'interprétabilité des résultats.

En définitive, la méthodologie présentée dans cette thèse a pour but d'aider Snecma à faire converger les labels des anomalies pour l'ensemble de ses utilisateurs. Elle vise également à faciliter et à inciter la mise en place d'une seule et même base de données

regroupant :

- d'une part toutes les mesures et leurs transformations prélevées sur les moteurs
- d'autre part les informations relatives aux moteurs pouvant être pertinentes telles que les résultats d'analyse des experts ou les dates de changement de pièces.

La base de données ainsi exploitable, cette thèse peut alors proposer un outil de labellisation qui pourra être utilisé pour améliorer, à travers la labellisation des données, les algorithmes de sélection et de classification supervisés.

## Abstract

Identifying early signs of failures in an industrial complex system is one of the main goals of preventive maintenance. It allows to avoid failure and reduce the degradation on a component by doing an earlier maintenance operation. Health monitoring for aircraft engines is one of the industrial fields for which this anomaly detection is very important and meaningful. Aircraft engine manufacturers such as Snecma collect large amount of engine related data during each flight. The idea is to be able to automatically detect when the engine is deviating from its normal behavior. Thus Snecma is developing applications allowing people to prevent engine failures by detecting early signs of anomaly.

This doctoral thesis is introdulcing how the experts' knowledge is used to process this engine related data. This first step has pointed out the difficulties in handling the data whether relating to their storage or relating to processing algorithms themselves. After that, this thesis offers a method to combine experts' knowledge with machine learning processes which follow Snecma needs such as the combination of various informations, error control or the interpretability of diagnostics results.

To do that the method is focusing directly on the data from the algorithms developed by the experts themselves. This is done by homogenizing the data and then by merging these data. This step allows for the use of supervised classification algorithms whom goal are to to group the items (here the engines) of a similar nature in the same class without losing the temporal component of the information. The homogenization of the data also allows the use of monitoring applications developed by experts in order to detect anomalies.

Before merging the data, a selection algorithm is used. This thesis describes how the selection process allows the monitoring algorithms to calibrate themselves. Moreover, this selection follows the first constraint imposed by Snecma concerning the interpretability of the results.

Eventually, the method introduced in this thesis aims at helping Snecma make the anomalies' labels converge for all its users. It also aims at incitating to gather all the data on a single database containing :

- the raw and the processed data from the engine,
- the engine related data that could be useful such as the results from experts analysis, etc.

Using this database, this thesis can then offer a labelisation tool that can be used to improve selection and classification algorithms.

# Remerciements

Je remercie, très chaleureusement, l'ensemble des personnes qui m'ont soutenu jusqu'au bout de cette thèse, en commençant par mes directeurs de thèse Marie Cottrell, Fabrice Rossi, Jérôme Lacaille et mon chef, Serge Blanchard. Ils m'ont accompagné, prodigué des conseils et encouragé, et pour cela, une simple page de remerciements ne sera malheureusement jamais suffisant.

Ensuite, je voudrais remercier tous les collègues de Snecma, en particulier les collègues du Pôle Monitoring qui ont contribué à la très bonne ambiance générale de l'équipe mais qui m'ont également partager leurs savoirs sur le fonctionnement des moteurs d'avions. Ils m'ont également soutenu durant toute la thèse. Sans eux, l'expérience n'aurait pas du tout été la même.

Je voudrais également remercier les collègues du SAMM et en particulier les doctorants qui apportent encore plus de vie à un laboratoire déjà bien vivant. Maintenant, c'est à votre tour, bon courage !

Je voudrais remercier Michel Verleysen et Alou Same qui ont accepté de rapporter mon manuscrit de thèse.

Enfin, je ne peux me permettre de ne pas remercier ma famille et mes amis qui m'ont tous soutenu avec une mention spéciale pour Dino qui a su se rendre disponible à des moments clés.

Mes pensées vont également aux proches d'Alain qui a été mon camarade de bureau durant six mois et qui nous a quittés trop tôt. A Maxine et Dad.

# Table des matières

CHAPITRE 1

# Introduction

## 1.1 Motivations

La société Snecma fait partie du Groupe Safran, un groupe international de très haute technologie leader en aéronautique, défense et sécurité. Snecma conçoit et vend des moteurs pour avions civils et militaires, pour lanceurs spatiaux et pour satellites. Dans cette thèse, nous nous intéressons seulement aux moteurs d'avions civils. Même si le principe du moteur à réaction est simple, chaque moteur est un système complexe dont le bon fonctionnement dépend d'un grand nombre de pièces et de sous-systèmes. Pour s'en convaincre, il suffit de voir la coupe du CFM56-7B (voir figure 1.1), moteur du Boeing 737. Un moteur est composé de plusieurs parties, allant du fan (entrée d'air) à la tuyère, en passant par les compresseurs basse pression et haute pression, puis la chambre de combustion, et les turbines haute pression et basse pression. Un œil novice peut se convaincre de la complexité du moteur s'il imagine que la plupart des parties sont tournantes.

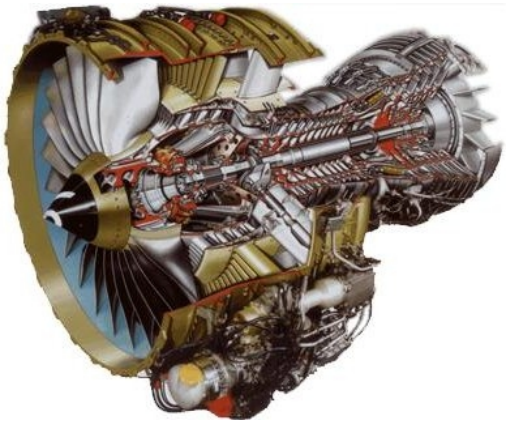

FIGURE *1.1: Coupe d'un CFM56-7B, moteur du Boeing 737.*

(*Source : http://www.cfmaeroengines.com/).*

Les moteurs d'avion sont soumis à des règles de sécurité très strictes afin d'éviter tout incident. Pour aider au respect de ces règles, plusieurs procédés sont mis en place. Cela commence par une conception et une fabrication très robustes permettant aux moteurs d'obtenir des certifications correspondant à des normes très précises et

très contraignantes. Ensuite, pendant l'exploitation, les moteurs sont soumis à des inspections, planifiées à l'avance ou décidées à la suite d'un événement. Les pièces ayant une durée de vie inférieure à celle du moteur, sont changées régulièrement selon un calendrier qui anticipe leur usure.

Grâce aux appareils de mesures et aux capteurs qui sont embarqués, de nombreuses données sont collectées durant les vols. Les données affichées sur les instruments de bord fournissent au pilote toutes les informations utiles au bon déroulement du vol, mais l'ensemble de ces données sont également enregistrées dans le but de détecter des défaillances avérées (diagnostic) ou imminentes (pronostic). Au vu de l'ensemble des données, une phase essentielle de la surveillance consiste à les interpréter afin de décider s'il y a ou non présence d'anomalie et éventuellement d'identifier son type. A partir de cette interprétation des données, les opérateurs soumettent des recommandations aux compagnies aériennes dans le but d'optimiser leur maintenance.

Pour faciliter l'analyse et l'interprétation de ces données, beaucoup d'outils ont été développés. L'un des plus utilisés est la maîtrise statistique des procédés (voir Oakland (2008) ou Sohn et al. (2000) pour une application industrielle) : on visualise l'écart de certains paramètres physiques à leurs valeurs de référence. Lorsqu'un écart trop important est constaté, une alerte graphique est présentée aux opérateurs chargés du suivi de l'état des moteurs.

Les motoristes et les compagnies aériennes ont besoin de développer constamment des outils supplémentaires pour améliorer encore plus la disponibilité des moteurs d'avion et optimiser les coûts de maintenance. C'est le travail essentiel des services de *Health Monitoring*. Le schéma de la figure 1.2 illustre l'impact du *Health Monitoring* sur les temps d'immobilisation à la suite d'une opération de maintenance du moteur. Dans le premier cas, sans Heath Monitoring, un événement opérationnel a lieu et une opération de maintenance non planifiée est faite. Il faut effectuer un diagnostic du moteur pour identifier les conséquences de l'événement opérationnel et faire parvenir les pièces nécessaires. Toutes ces opérations peuvent prolonger le temps de non disponibilité du moteur et donc de l'avion. Dans le deuxième cas, avec *Health Monitoring*, une anomalie est détectée avant que l'événement opérationnel n'ait eu lieu, une dégradation plus importante est alors évitée, et un diagnostic en amont permet de faire parvenir rapidement les pièces nécessaires. Une opération de maintenance, beaucoup plus courte avec *Health Monitoring* que sans, est alors planifiée, ce qui permet d'optimiser les coûts.

Une des limites des algorithmes disponibles aujourd'hui est qu'ils permettent d'analyser essentiellement des sous-systèmes. Par exemple, la surveillance va porter sur le balourd (déséquilibre de la répartition des masses autour de l'axe de révolution du moteur), sur les engrenages, les roulements, la nacelle (support et capot d'un moteur) ou encore sur l'ingestion des corps étrangers (*Foreign Object Damage*, FOD). Cependant, la calibration des différents algorithmes demande un soin particulier et l'aspect combinaison (fusion) des diagnostics est encore mal maîtrisé. Les avancées en termes d'applications des concepts d'apprentissage automatique présentent des perspectives d'amélioration des techniques de surveillance actuelles. Les résultats fournis par ces

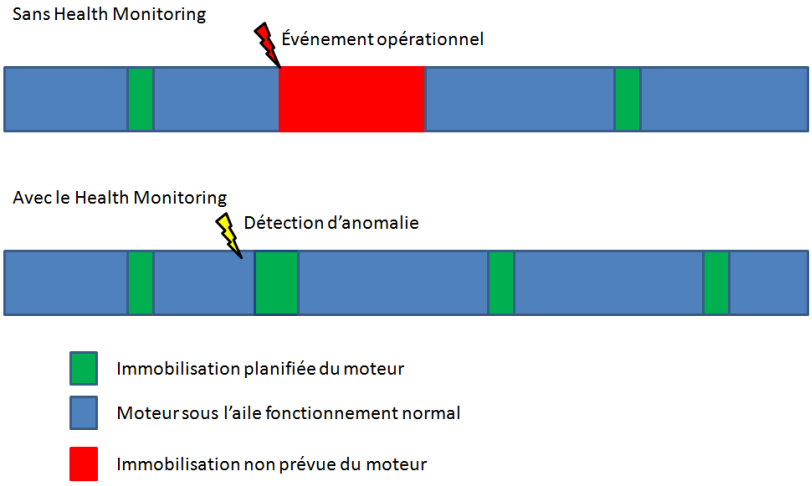


FIGURE *1.2: Influence du Health Monitoring sur les temps d'immobilisation. Le Health Monitoring peut détecter des prémices de panne et donc limiter la dégradation du moteur en évitant un événement opérationnel. Le temps d'immobilisation du moteur peut alors être fortement réduit.*

méthodes de type apprentissage automatique sont cependant parfois difficilement interprétables par l'homme métier, c'est-à-dire par les experts ou les opérateurs Snecma connaissant les moteurs et qui auront à traiter ces informations. Un des objectifs de la thèse est de proposer des solutions qui permettent d'éviter cet écueil.

De plus, si l'on parvient à une maîtrise plus complète des techniques de détection et d'identification des anomalies, on pourrait envisager à terme plus de tolérances concernant les exigences de conception du moteur, en cherchant par exemple à surveiller un sous-système plutôt qu'à ajouter continuellement des couches de protection.

Un deuxième objectif est d'améliorer les procédés d'aide au dépannage (*troubleshooting*) du moteur pour que, en cas d'anomalie, on puisse disposer de procédés systématiques et efficaces de recherche de la cause. Enfin, on s'intéresse à la surveillance de flottes de moteurs plutôt qu'à un moteur isolé. Dans cet esprit, une première approche proposée dans Côme et al. (2010) et Lacaille et Côme (2011) consiste à utiliser une carte de Kohonen pour visualiser un ensemble de moteurs. On construit une carte de Kohonen à l'aide de moteurs déjà analysés et connus, puis on projette les moteurs d'une flotte à l'instant présent. L'état d'un moteur peut être estimé à partir de sa position sur la carte (voir figure 1.3).

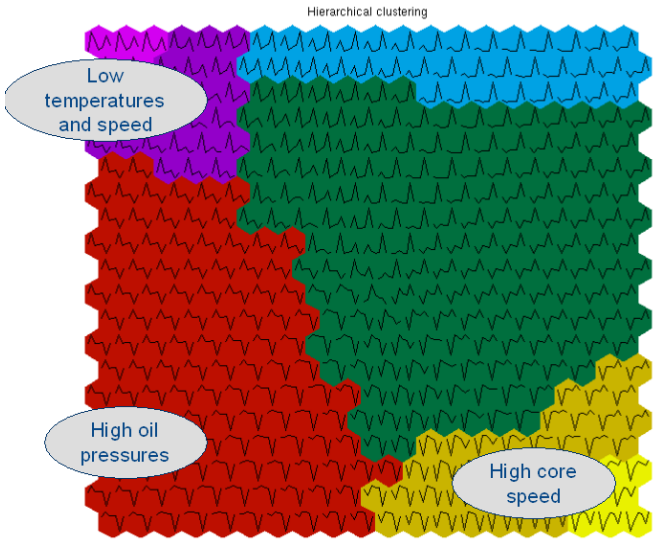


FIGURE *1.3: Exemple de carte permettant de visualiser l'état d'une flotte de moteur. En affichant sur la carte les moteurs, il serait possible d'identifier d'un coup d'œil les moteurs dont l'état de santé se dégrade.*

(*Source : Lacaille et Côme (2011).*)

## 1.2 Contributions

Le travail mené dans cette thèse a permis de mettre à disposition de Snecma et de la communauté aéronautique une méthodologie adaptée à la fusion et à l'interprétation des indicateurs (mesures, déviations, résultats d'algorithmes) pour le diagnostic et le pronostic, dans le cadre de la surveillance des moteurs. L'accent a été mis en particulier sur l'interprétabilité des résultats fournis par la méthodologie de façon à répondre aux exigences des opérateurs métier de Snecma. Cette interprétabilité est très importante car les opérateurs ne veulent pas travailler avec des outils de type boite noire ; ils doivent pouvoir interpréter facilement les informations fournies par les algorithmes de surveillance, notamment parce qu'il leur appartient de valider le résultat.

Il est intéressant de remarquer que le travail mené dans cette thèse a permis de confirmer qu'il n'est pas possible de se passer du savoir métier. En effet, un grand nombre de données, essentiellement temporelles, sont enregistrées durant chaque vol. Celles-ci dépendant du contexte (altitude, température...), elles nécessitent un prétraitement avant analyse. Les experts ont donc ici un rôle essentiel consistant à définir les variables à retenir pour l'analyse et les méthodes de prétraitement. Une des principales contributions de ce travail de thèse est de proposer une méthode de fusion exploitant le savoir expert.

La méthodologie que nous proposons permet également de souligner l'importance de la labellisation (caractérisation de l'état), d'abord pour évaluer et valider toutes les

applications de détection d'anomalie, mais aussi pour faire émerger des relations entre les sous-systèmes qui ne sont pas toujours parfaitement connues. Notre travail a permis de constater l'absence d'une définition commune à tous les opérateurs de ce qu'est un label. Nous proposons une structuration des données qui conduit à une première définition des labels utilisables par tous, même si cette structuration pourrait encore être améliorée. Notre travail permet également de montrer qu'une définition précise des labels contribue, à travers la sélection automatique des indicateurs, non seulement à aider à la calibration des applications de surveillance, mais aussi à faire des choix sur les types de normalisation à utiliser.

## 1.3 Plan de la thèse

La thèse est composée de 7 chapitres.

Ce premier chapitre d'introduction présente l'objectif de la thèse en expliquant les motivations et les contributions.

Le deuxième chapitre, intitulé « Introduction au *Health Monitoring* et aux problèmes de maintenance prédictive » correspond au positionnement du sujet de thèse : y est expliqué ce qu'est un événement opérationnel, le savoir Snecma et la maintenance prédictive. Ce chapitre explique notamment qu'il y a un besoin d'optimisation des opérations de maintenance, et décrit les solutions de maintenance prédictive utilisées aujourd'hui.

Le chapitre 3, intitulé « détection d'anomalies » constitue une transition, il décrit un peu plus précisément les solutions développées par Snecma, en donnant des exemples d'application de *Health Monitoring*. Ce chapitre permet de montrer comment est codé le savoir des experts Snecma, en décrivant notamment la structure des applications de surveillance et plus précisément l'étape de normalisation utilisée pour rendre les quantités étudiées indépendantes du contexte extérieur. Enfin, ce chapitre donne des exemples de visualisations souhaitées pour aider à l'identification des pannes.

Dans le chapitre 4, intitulé « fusion d'indicateurs », y sont décrites les principales difficultés rencontrées, notamment le fait que les informations sont nombreuses, dispersées et hétérogènes. Pour les fusionner, il y a un besoin d'uniformisation des données. Ce chapitre décrit la technique d'uniformisation qui consiste à transformer les données de sortie des applications de surveillance en un indicateur de décision. Il y est précisé la construction et la sélection des indicateurs retenus puis y sont définies les méthodes de classification qui permettent de passer de l'ensemble des indicateurs à l'attribution d'un label. Enfin, une section est consacrée aux méthodes de sélection de variables utilisées dans les problèmes avec des données en grandes dimensions.

Le chapitre 5, intitulé « mise en œuvre et expérimentations », montre que la méthode proposée permet effectivement de détecter des anomalies sur des données simulées. Cette partie présente la façon dont les données sont simulées, puis décrit plus précisément les indicateurs, et notamment les tests statistiques utilisés. Lors de la présentation des résultats, il est montré comment la méthodologie proposée dans cette thèse répond aux objectifs d'interprétabilité et d'identification correcte des anomalies, et comment elle

s'inscrit dans le prolongement des méthodes actuellement utilisées à Snecma supplémentées d'informations issues de l'expertise, du retour d'expérience et de l'interactivité avec les experts aéronautiques.

Le chapitre 6, intitulé « exploitation du système », montre des résultats sur des données réelles dans le cas particulier des *customers notifications reports* (CNR). Les difficultés rencontrées dans le cas réel sont remontées et un outil permettant une visualisation, une labellisation et la validation est décrit. Le chapitre s'achève par des exemples d'utilisations possibles de la méthodologie.

La thèse finit par les « conclusion et perspectives ».

CHAPITRE 2

# Introduction au *Health Monitoring* et aux problèmes de maintenance prédictive

Ce chapitre définit plus précisément ce qu'est le *Health Monitoring* dans le monde aéronautique, en décrivant notamment ce qu'est un évènement opérationnel. Ce chapitre présente les besoins et les solutions actuellement mises en œuvre pour la maintenance des moteurs d'avions

## 2.1 Évènements opérationnels d'origine moteur

### 2.1.1 Définitions et statistiques d'occurrence

Dans le monde de l'aéronautique, lorsque l'on parle d'évènements opérationnels, on entend tous les évènements pouvant perturber le bon déroulement d'un vol. Ces évènements opérationnels peuvent être de différentes natures et avoir différentes origines.

Dans le cadre de cette thèse, on s'intéresse uniquement aux évènements opérationnels ayant pour origine une défaillance du moteur et qui peuvent avoir comme conséquences :

- l'abandon du décollage alors que l'avion a déjà été lancé, *Aborted Take-Off* (ATO) ;
- le retour de l'avion à sa base de décollage, *Air Turn Back* (ATB) ;
- l'arrêt imprévu du moteur en vol (initié ou non par le pilote), *In-Flight ShutDown* (IFSD) ;
- un retard ou une annulation, *delay and cancellation* (D&C).

Un événement opérationnel peut être provoqué par le mauvais temps ou par l'ingestion d'un corps étranger tel qu'un oiseau, ce qu'on appelle FOD (*Foreign Object Damage*). Ces évènements extérieurs peuvent entraîner la dégradation d'une partie du moteur et être suivis par un événement opérationnel immédiatement ou après plusieurs vols. Ceci explique que bien que les moteurs soient conçus pour être extrêmement fiables, il n'est pas possible d'éviter tous les évènements opérationnels.

Ces évènements opérationnels restent heureusement très rares. Ainsi, dans le cas du moteur CFM56-7B, avec plus de 6000 moteurs en service équipant en exclusivité les Boeing 737 nouvelle génération, en 2013, il a été recensé 2 IFSD (*In-Flight ShutDown*) pour 1000000 d'heures de vol, et par ailleurs 5 ATO (*Aborted Take-Off*) pour 1000000 de départs. Le tableau 2.1 fournit les taux de disponibilités de l'ensemble des CFM56. À

titre d'information, les CFM56, c'est plus de 24 565 moteurs CFM56 et plus de 135000 heures de vol par jour.

### 2.1.2 Conséquences des évènements opérationnels

Les évènements opérationnels sont rares, mais chaque fois qu'il s'en produit un, cela peut se transformer en une gène importante pour les passagers. Les abandons de décollage (ATO), les retours de l'avion (ATB) entraînent des retards ou des annulations (D&C) qui peuvent dégrader fortement l'image d'une compagnie même s'ils se produisent rarement. Ces retards et annulations peuvent avoir d'autres origines mais ici, nous ne nous intéressons qu'aux évènements dus au moteur. Même les arrêts imprévus en vol du moteur (IFSD), bien que souvent imperceptibles pendant le vol, peuvent engendrer des retards ou des annulations (Delay and Cancellation, D&C) sur les vols suivants.

Les évènements opérationnels ont également des conséquences financières importantes pour les compagnies aériennes. Du fait de leur caractère imprévisible, ils imposent une maintenance non programmée du moteur et peuvent bouleverser les planifications. Le diagnostic et la réparation peuvent entraîner des coûts importants, en raison de l'immobilisation de l'appareil et du prix des réparations. Par exemple, pour un Boeing 737-800 (un des Boeing 737NG (*Next Generation*)), ce coût peut aller jusqu'à 50000$ par jour d'immobilisation (Demirci et al. (2008)). De même, les compagnies doivent faire face à des coûts dus au retard comme par exemples l'hébergement des passagers, les repas, les indemnités dues...

L'ensemble de ces coûts étant par nature difficiles à prédire et à budgéter, les compagnies cherchent à réduire encore plus l'occurrence de ces évènements opérationnels.

TABLE *2.1: Taux de disponibilité de différents moteurs CFM56 en 2013. Plus précisément, le tableau fournit le taux de dépose moteur non prévue, le taux d'In-Flight Shut-Down (ISFD), le taux d'Aborted Take-Off (ATO), et le taux de disponibilité du moteur au départ prévu.*

| Type de moteurs | Dépose moteur non prévue* | IFSD* | ATO** | Taux de disponibilité du moteur au départ*** |
|---|---|---|---|---|
| CFM56-3<br>*ex : B737-300/400* | 0.031 | 0.002 | 0.006 | 99.967 |
| CFM56-5A<br>*ex : A319/320* | 0.019 | 0.002 | 0.007 | 99.936 |
| CFM56-5B<br>*ex : A320/321* | 0.013 | 0.001 | 0.007 | 99.970 |
| CFM56-5C<br>*ex : A340* | 0.022 | 0.005 | 0.018 | 99.799 |
| CFM56-7B<br>*ex : B737NG* | 0.013 | 0.002 | 0.002 | 99.962 |

* Pour 1000 heures de vol.

** Pour 1000 départs.

*** Nombre de départs non touchés par un retard ou une annulation quelque soit la cause.

(*Source : Communication Snecma.*)

## 2.2 Maintenance des moteurs d'avions

Pour prévenir les évènements opérationnels, différents dispositifs de maintenance sont mis en place de manière systématique. En premier lieu, des pièces de moteurs d'avions sont changées régulièrement sur la base de la connaissance de leurs durées de vie moyennes. Parallèlement, les moteurs sont inspectés régulièrement afin de repérer ou identifier d'éventuelles pièces de moteurs endommagées ou usées prématurément.

Certains indicateurs permettent d'alerter les techniciens de maintenance : on sait par exemple que la consommation de carburant en croisière augmente avec le temps en raison de l'usure. On sait également que la température de l'air à la sortie de la turbine haute pression (appelé *Exhaust Gaz Temperature*, EGT) est un indicateur d'usure. Plus précisément, on calcule la marge EGT, c'est-à-dire la différence entre la limite supérieure de l'EGT à ne pas dépasser et l'EGT mesurée. Une marge EGT faible permet de suspecter une usure avancée.

Ces indicateurs (augmentation de la consommation, diminution de la marge EGT) peuvent être également un signe de saleté encrassant le moteur. De ce fait, les moteurs sont nettoyés régulièrement à grande eau (*water-wash*), c'est-à-dire que des centaines de litres d'eau sont pulvérisées plusieurs fois à l'entrée du moteur, tandis que celui-ci tourne sans allumage ou injection de carburant.

Dans tous les cas, lors d'une inspection programmée ou supplémentaire, si une anomalie est constatée, une opération de maintenance imprévue sera effectuée. Celle-ci peut aller jusqu'à la dépose du moteur. Alors que les maintenances programmées sont intégrées dans le planning de la compagnie, ces maintenances imprévues engendrent des coûts supplémentaires et de la désorganisation.

Ce principe peut se comparer à S.M.A.R.T. (*Self-Monitoring, Analysis, and Reporting Techniology*) pour les disques durs. Ce système fournit des indicateurs de fiabilité (taux d'erreur en lecture, température, etc.) dans le but de prévenir les défaillances prévisibles (usure, vieillissement de pièces mécaniques, etc.). Cela permet à l'utilisateur de prendre des précautions, par exemple en copiant les données sur un autre disque avant que les disques durs ne soient en panne.

## 2.3 Nécessité d'une aide au diagnostic pour l'optimisation des opérations de maintenance

Comme nous avons pu le voir précédemment, la maîtrise des coûts de maintenance est cruciale pour les compagnies aériennes. Elles cherchent donc à éviter et à prévenir les évènements opérationnels. Pour répondre à ce besoin, Snecma développe et met en œuvre des méthodes de surveillance, *Prognostic Health Monitoring* (PHM), dont l'un des rôles est de concevoir des solutions de détection des prémices de ces évènements opérationnels. Concrètement, la mission du PHM est d'aider les compagnies aériennes à minimiser le nombre d'évènements opérationnels, à optimiser et planifier les opérations de maintenance pour mieux prévoir les coûts.

Pour optimiser les opérations de maintenance, il faut fournir une aide au diagnostic de l'état d'un moteur d'avion. On sait que les moteurs sont d'une grande complexité (même si le principe en est très simple), et établir un diagnostic peut être long et coûteux. Par exemple, pour rechercher les causes d'une anomalie, on peut passer du temps à inspecter une partie non concernée du moteur, voire à le déposer entièrement pour finalement constater que cela était inutile et néanmoins très coûteux.

Il est indispensable que les experts motoristes puissent aider les compagnies aériennes à identifier l'origine d'une anomalie. Cela s'appuie sur leur savoir et sur le retour d'expérience (RetEx). Dans le cas où un événement opérationnel a réellement eu lieu, le constructeur peut décider d'envoyer sur place des experts en diagnostic des moteurs ou de faire rapatrier les pièces défectueuses pour les analyser et identifier l'origine de l'événement.

## 2.4 Les solutions actuellement mises en œuvre

### 2.4.1 Capteurs

Sur chaque moteur d'avion, des capteurs sont placés à différents endroits afin de recueillir, tout au long du vol, des informations sur le contexte extérieur (température extérieure, pression extérieure, ...) et sur l'état du moteur (vitesse de rotation, pression d'huile, EGT...). Ces informations sont indispensables pour le pilote qui doit assurer le bon déroulement du vol. Ces informations sont également utilisées par le système de régulation (FADEC [1]) qui transforme la position de la manette (équivalent à l'accélérateur pour une voiture) en une commande qui est envoyée au moteur et qui est compatible avec la sécurité du vol.

Les mesures issues des capteurs sont aussi essentielles pour optimiser les opérations de maintenance. Les experts les utilisent pour surveiller le fonctionnement des moteurs, identifier des prémices de pannes, lancer des alertes.

La figure 2.1 présente des exemples de quantités physiques telles que la température à la sortie de la turbine haute pression (EGT, *Exhausted Gas Temperature*), la vitesse de rotation de l'arbre haute pression (N2), la vitesse de rotation de l'arbre basse pression (N1) et le débit carburant (FMV, *Fuel Metering Valve*). Il s'agit d'exemples de données collectées et analysées dans le cadre de la surveillance des moteurs d'avion.

### 2.4.2 Mode de transmission de ces mesures

Durant chaque vol, des messages sont transmis de l'avion vers le sol par radio, via un système de communications codées. Les avions motorisés par Snecma utilisent, par exemple, le système ACARS (Aircraft Communications Addressing and Reporting System). Ces messages contiennent un ensemble résumé des paramètres de vol à des

1. Le système FADEC (Full Authority Digital Engine Control) est le système de contrôle et de régulation du moteur. Il assure un niveau élevé de sécurité, de fiabilité et de performance pour obtenir un fonctionnement optimal des moteurs.

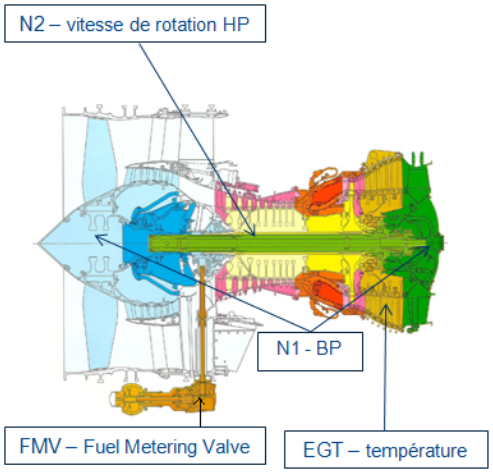


FIGURE *2.1: Exemples de paramètres suivis sur un moteur CFM. Une grande quantité d'information doit être analysée.*

instants jugés décisifs par les experts (décollage, phase de croisière, atterrissage...). Durant chaque vol, lors des instants identifiés comme pertinents par les experts, un *snapshot*, c'est-à-dire un aperçu de l'ensemble des mesures capteurs à cet instant, est réalisé. Ces mesures sont éventuellement prétraitées par un calculateur embarqué avant d'être transmises au sol.

Ce sont essentiellement ces données résumées que nous exploitons dans ce travail pour développer un outil de détection automatique des prémices de pannes.

Le savoir des experts permet de définir des seuils au-delà desquels les mesures sont jugées anormales. Grâce à la définition de ces seuils, on peut mettre en place un système de détection automatique des anomalies potentielles. Ces seuils sont choisis de façon à minimiser les risques de laisser passer une anomalie sans la détecter. A cette étape, la technique de détection reste essentiellement univariée, en examinant chaque mesure.

### 2.4.3 Critères de performance des méthodes de détection

Pour évaluer les performances d'un détecteur de prémices de panne, des indicateurs de performance (*Key Performance Indicators*, KPI) sont définis par les motoristes et les compagnies. Il s'agit de comparer et améliorer les différentes méthodes mises au point par les spécialistes des différentes parties du moteur.

La probabilité de bonne détection (*Probality Of Detection*, POD) est le premier critère. C'est la probabilité qu'il y ait une détection sachant qu'il y a effectivement une anomalie. La POD est calculée comme le rapport entre le nombre d'anomalies détectées et le nombre total d'anomalies.

On note :

$$POD = \mathbb{P}(détection|anomalie)$$

Le but est donc de maximiser cet indicateur en ajustant les valeurs des seuils.

Cependant, une alarme peut se révéler être une fausse alarme. En effet, il est possible d'ajuster les seuils de façon à toujours avoir une alarme, ce qui donnerait une POD de 100% mais évidemment dans ce cas, le système de détection n'est pas intéressant. Ainsi, tout comme pour la détection, un indicateur de performance a été défini pour mesurer le taux de fausses alarmes.

On définit la probabilité de fausse alarme (PFA) comme la probabilité que le moteur soit sain alors qu'il y a eu une alarme. C'est donc le rapport entre le nombre d'alarmes concernant des moteurs sains et le nombre total d'alarmes.

On note :

$$PFA = \mathbb{P}(sain|alarme)$$

La PFA ne doit pas être confondue avec l'erreur de première espèce utilisée par les statisticiens qui est la probabilité d'avoir une alarme sachant que le moteur est sain.

Cet indicateur (PFA) doit absolument être minimisé. Cela est très important pour Snecma qui ne peut se permettre de signaler à la compagnie aérienne des fausses alarmes trop fréquentes :

- d'abord parce qu'une fausse alarme peut avoir des conséquences financières non négligeables pour la compagnie qui sera amenée à inspecter le moteur, voire à le déposer inutilement,
- par ailleurs, des occurrences de fausses alarmes répétées peuvent entraîner un problème de confiance de la compagnie envers les diagnostics de Snecma.

### 2.4.4 Rôle de l'opérateur métier

Actuellement, le processus de détection automatique d'anomalie est configuré de façon à maximiser la POD, dans le but de ne laisser passer aucune anomalie avérée. En conséquence, la PFA est mal maîtrisée, et on doit alors faire appel, pour chaque alarme, à un opérateur métier qui analyse les données et confirme ou non l'alarme. Le rôle de l'opérateur métier est crucial pour limiter les fausses alarmes. En pratique pour 1000 alarmes, environ une dizaine seulement sont confirmées par l'opérateur métier.

Lorsqu'une anomalie est confirmée, l'opérateur métier apporte également son expertise dans la détermination de l'origine de l'anomalie constatée. Pour cela, il faut analyser l'évolution de chacun des paramètres mesurés à bord de l'avion et transmis au service de maintenance.

Pour aider ces opérateurs métier, une solution est actuellement en cours de développement à Snecma et consiste à définir des scenarii d'évolution des paramètres. A chaque scenario, les experts sont capables d'associer une classe d'anomalie, c'est-à-dire qu'ils peuvent identifier l'origine de l'anomalie potentielle à partir des évolutions des paramètres. Cette méthode a l'avantage d'avoir un processus de formation des résultats compréhensible pour l'opérateur métier.

Cependant, l'analyse des données et l'identification du scénario ne sont pas toujours aisées. En effet, deux scenarii peuvent se traduire par les mêmes évolutions, c'est-à-dire que la discrimination proposée par la description des scenarii n'est pas toujours suffisante. De plus, la liste des scenarii utilisée n'est pas exhaustive.

Or, le nombre élevé d'alarmes (induit par la recherche de la détection maximale) entraîne une sollicitation continue des opérateurs métier. C'est pourquoi il est important de leur fournir une aide pour l'analyse des données recueillies et la prise de décision. Un trop grand nombre de fausses alarmes et de non détections d'anomalie réelles peut ternir le service de suivi proposé aux compagnies clientes.

### 2.4.5 Algorithmes de détection d'anomalie

Pour cela, à côté de la définition des scenarii « suspects », des algorithmes de détection et d'identification des anomalies sont développés par les experts métier. Ces algorithmes de détection d'anomalie sont présentés dans le chapitre suivant, mais il faut souligner qu'ils surveillent essentiellement des sous-systèmes des moteurs alors que l'on vise à réaliser un diagnostic du système dans son ensemble. Le rôle de l'opérateur métier est alors de faire une synthèse de tous les résultats fournis par les différents algorithmes, ce qui n'est pas une tâche facile. En effet, les sous-systèmes d'un moteur sont dépendants les uns des autres et l'influence de ces dépendances sur les mesures n'est pas très bien connue.

## 2.5 Conclusions

Dans ce cadre, le but de notre travail est de développer une méthode permettant d'assister l'opérateur métier à agréger et à traiter toutes les informations disponibles lors de sa prise de décision pendant un diagnostic. Pour que l'opérateur puisse avoir confiance en la méthode proposée, celle-ci ne doit pas se présenter sous forme de boite noire. L'opérateur a besoin de comprendre le processus de formation des résultats.

Cette exigence est à prendre en considération lors du choix final de la méthode d'agrégation proposée dans cette thèse. C'est une condition nécessaire pour que la méthode puisse être considérée comme acceptable par l'opérateur.

Dans le chapitre suivant, nous détaillons les différentes méthodes existantes utilisées par Snecma pour la détection d'anomalie. Ces méthodes sont implémentées sous forme d'algorithmes provenant de spécialistes différents.

CHAPITRE 3

# Détection d'anomalies

Dans ce chapitre nous présentons les principes généraux de la détection d'anomalie tels qu'on peut les trouver par exemple dans Chandola et al. (2009). Nous insistons sur deux étapes fondamentales qui sont la fabrication des variables d'intérêt et leur normalisation, et nous rappelons l'architecture des algorithmes actuellement développés à Snecma.

## 3.1 Principes généraux sur la détection d'anomalies

Comme on peut le voir dans Chandola et al. (2009), le principe de base pour détecter une anomalie repose sur la définition d'une zone de normalité de fonctionnement. Chaque donnée identifiée en dehors de cette zone de normalité est alors considérée comme une anomalie. En pratique, il faut donc définir cette région de normalité et séparer les données « normales » des données « anormales ». Cette séparation est rendue complexe par le fait que toutes les mesures sont entachées de bruit.

De nombreuses méthodes ont été proposées dans la littérature. Une méthode très simple consiste à supprimer un ensemble de données « suspectes » apportant de l'hétérogénéité à l'ensemble des mesures, de sorte que l'ensemble restant soit le plus homogène possible. Le problème est alors d'enlever un ensemble de données le plus petit possible pour ne pas appauvrir l'information. Une première difficulté est de distinguer les données aberrantes qui ne sont que des erreurs d'acquisition, de saisie ou de *reporting*, des données véritablement signes d'anomalie et qu'il faudra analyser.

Une autre méthode très classique consiste à projeter les données dans un nouvel espace sur lequel les données « normales » et « anormales » sont bien séparées. Par exemple, les support vector machines (SVM) projettent dans un espace de caractéristiques de dimension très supérieure. Mais, il sera généralement préféré une projection sur un espace de dimension inférieure (idéalement 1 ou 2) pour l'établissement d'une métrique qui a un sens par rapport aux incertitudes de mesure mais également pour l'affichage et l'interprétation des résultats comme le propose l'analyse en composante principale.

On peut également citer toutes les méthodes de *clustering* (*competitive learning*, algorithme de Forgy, k plus proches voisins, algorithme de Kohonen, classification hiérarchique ascendante, ...) qui permettent avec plus ou moins de facilité d'isoler les données « anormales ». Bien que ces méthodes ne soient pas supervisées et qu'elles ne nécessitent pas de données labellisées, les labels sont indispensables pour pouvoir les tester et les exploiter.

Or, comme on la pu le voir précédemment, les moteurs sont extrêmement fiables et on dispose de très peu d'anomalies labellisées. De plus, pour des raisons de coût, il est difficile de réaliser des essais de dégradation de moteur. Une alternative est alors de simuler des anomalies en ajoutant des données anormales. Dans ce cadre, il devient possible d'utiliser des méthodes d'apprentissage supervisées telles que CART (*Classification And Regression Trees*, voir 4.3.2.1), Random Forests (voir 4.3.2.2), Perceptron Multicouches, etc.

Lorsque le problème est unidimensionnel, il existe des solutions, mais dans le cas qui nous intéresse, les anomalies sont essentiellement multidimensionnelles et mettent en jeu plusieurs composantes du vecteur de mesures. Il est donc indispensable de prendre en compte les corrélations de ces mesures. On pourra se référer par exemple à l'étude Klein et Issacharoff (2009) où les auteurs obtiennent des résultats très satisfaisants à condition de prendre en compte les corrélations entre les mesures. On voit clairement dans la figure que des données peuvent être vues comme normales quand on les regarde séparément, alors que le couple des mêmes mesures est anormal.

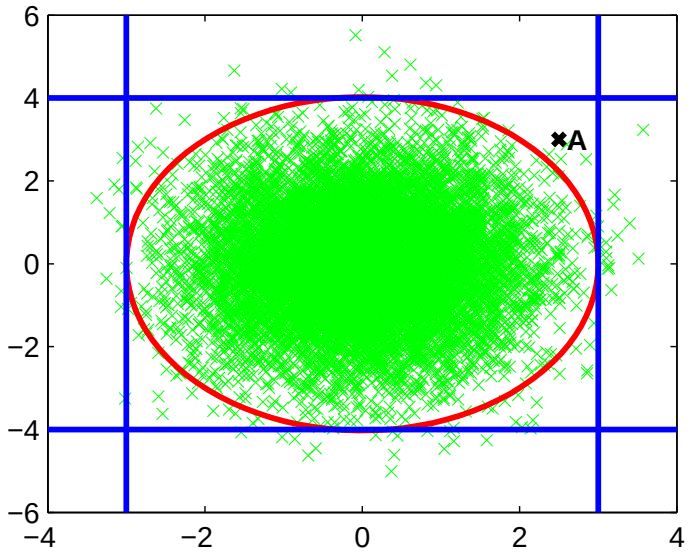


FIGURE *3.1: Les zones de fonctionnement normales de X, resp. Y, sont les intervalles [-3, 3], resp. [-4, 4] alors que la zone de fonctionnement du couple correspond à la surface de l'ellipse rouge. L'observation A serait normale si on considère X et Y séparément, mais est en réalité anormale.*

## 3.2 Comment détecter les anomalies d'un moteur d'avion

Dans la suite de ce chapitre, on décrit brièvement les principales étapes constitutives du *Health Monitoring*. Ces étapes sont :

- l'acquisition des données ;
- le prétraitement des données ;
- l'étude du comportement normal ;
- l'estimation de la déviation à la normalité ;
- l'identification de la panne le cas échéant.

Ces étapes sont communes à toutes les applications de surveillance développées par les experts.

La première étape du *Health Monitoring* est l'acquisition des données de mesure. Ce sont les experts qui définissent l'ensemble des données pertinentes à prélever. Ils choisissent le nombre et l'emplacement des capteurs à installer en tenant compte des coûts. Les données brutes doivent en général être transformées en un format utilisable par les programmes informatiques.

La seconde étape consiste à réduire le bruit, à comprimer les données et à définir les variables d'intérêt qui seront utilisées. Le rôle des experts est là aussi fondamental, en particulier pour permettre la distinction entre les variables de fonctionnement du moteur et les variables de contexte (environnement, température extérieure, type de pilotage, avion...).

La définition des variables d'intérêt n'est pas une tâche aussi simple qu'il pourrait paraître. La plupart des variables d'intérêt ne sont pas mesurées directement. Si on s'intéresse, par exemple, au temps écoulé entre l'ouverture d'une vanne et l'allumage dans la chambre de combustion, il faut pouvoir récupérer numériquement les instants précis de ces deux événements. Une solution à ce problème a été décrite dans Rabenoro et Lacaille (2013b) et a fait l'objet d'un brevet Rabenoro et Lacaille (2013a) sur la détection d'instants. Cette solution inclut une application qui permet aux experts de sélectionner des instants potentiels sur une série de courbes d'apprentissage. Une fois l'apprentissage effectué, l'application permet l'extraction automatique des instants recherchés sur de nouvelles courbes. Ce processus permet par exemple, à l'expert d'étudier la variabilité de ces instants en fonction du contexte. Une fois les variabilités étudiées, l'expert peut même être amené à définir une nouvelle variable plus pertinente. Des exemples sont donnés dans Rabenoro et Lacaille (2013b) et Flandrois et al. (2009).

Dans le schéma 3.2, l'évolution de trois paramètres d'intérêt sont représentés lors d'une phase de démarrage. Des exemples d'instants à extraire pour calculer des variables d'intérêt sont représentés. Ainsi la variable $t\_a$ se calcule en extrayant l'instant qui correspond au début d'injection du carburant dans la chambre de combustion (courbe FMV, *Fuel Metering Valve*) et l'instant correspondant au point d'inflexion visible sur la courbe N2 (vitesse de rotation de l'arbre haute pression).

D'autre part, les conditions de vol d'un avion peuvent être très variées et le contexte extérieur (température, humidité, pression, altitude) influence clairement les mesures relevées. D'autres paramètres entrent également en ligne de compte comme le type de

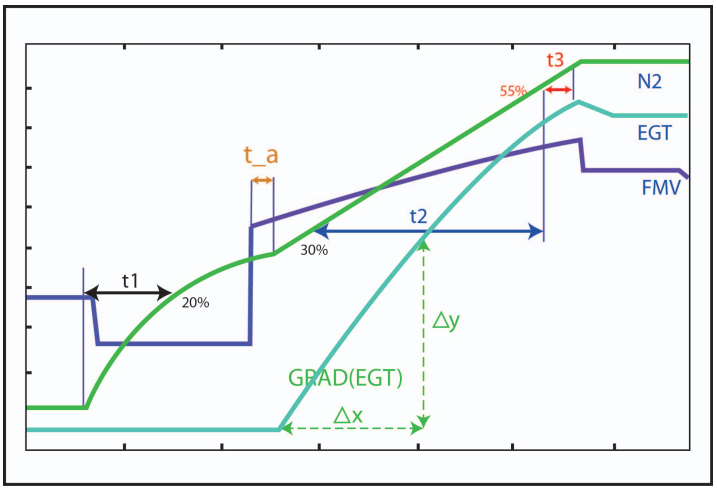


FIGURE *3.2: Exemples de variables d'intérêt identifiées par les experts lors de la phase de démarrage.*

pilotage ou l'état général du moteur. Par exemple, lorsqu'un avion effectue le premier vol de la journée, il démarre généralement avec un moteur froid. Lorsqu'il effectue un vol consécutif à un autre vol, il démarre avec un moteur chaud. Et de fait, les températures d'huile au démarrage ne sont pas identiques.

Le système de régulation du moteur a tendance à supprimer certains éléments de contexte. Par exemple, s'il y a une perte de poussée suite à une dégradation d'un composant, le système de régulation peut compenser en faisant en sorte que plus de carburant soit injecté dans la chambre de combustion. Cependant, tous les effets ne sont pas supprimés. Or, pour permettre la comparaison entre différents moteurs ou la comparaison de différents états d'un même moteur, il faut rendre les variables de fonctionnement indépendantes du contexte : ce processus s'appelle la *normalisation*.

Dans la figure 3.3, est illustré un exemple de variables d'intérêt avant et après normalisation. Dans la partie haute, on constate des points bas, et des points hauts. Il s'avère que ces points hauts et ces points bas correspondent respectivement à des démarrages à chaud ou à froid. La normalisation permet de supprimer, entre autres, l'influence de la température du moteur au démarrage qui fluctue en fonction de la proximité du vol précédent.

Cette normalisation se fait en particulier à l'aide de modèles physiques nécessitant la connaissance experte du fonctionnement d'un moteur. Les procédures détaillées sont confidentielles, issues d'une connaissance métier ancienne et spécialisée. Elles ne sont pas abordées dans la thèse.

D'un point de vue mathématique, Lacaille et ses coauteurs ont proposé dans Lacaille et al. (2011) une méthode de régression linéaire des variables de fonctionnement sur les variables de contexte de type LASSO et LARS. Plus récemment, Bellas et al. (2012) ont proposé de commencer par une classification des types de contexte, ce qui permet de

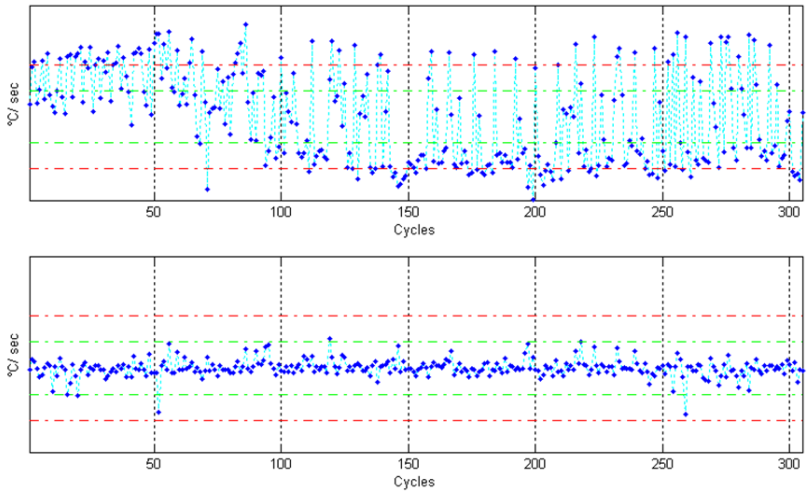


FIGURE *3.3: Exemples de variables d'intérêt avant (figure du haut) et après normalisation (figure du bas).*

faire une régression des variables de fonctionnement sur les variables de contexte à l'intérieur de chaque classe. Dans les deux cas, les variables de fonctionnement sont alors remplacées par les résidus de cette régression. L'exemple de la figure 3.4 tirée de Bellas et al. (2012) montre que le contexte peut jouer un rôle important pour différencier deux signaux qui peuvent paraître a priori semblables. On peut également utiliser des modèles autoregressifs pour supprimer la dépendance temporelle d'un instant à l'autre.

Une fois les données normalisées obtenues, l'étape suivante consiste à étudier leur distribution. Dans cette étape, si les données sont labellisées, on ne considère que les données sans anomalie. En l'absence de label, compte tenu de la rareté des données anormales, on considère l'ensemble des données en les supposant globalement sans anomalie. Cette modélisation peut être de très bas niveau : par exemple, on peut déterminer les bornes inférieures et supérieures de normalité. Elle peut être plus sophistiquée comme par exemple dans Bellas et al. (2013) où les auteurs supposent que les données « normales » résultent de l'observation d'un mélange de gaussiennes. Dans tous les cas, la difficulté principale ici est d'exploiter les relations de dépendance entre les mesures.

L'étape suivante consiste à mesurer pour chaque nouvelle donnée sa déviation à la normalité et une probabilité qu'elle soit anormale. On définit ainsi ce qu'on appelle un *score d'anomalie*.

Les données sont par nature temporelles, et détecter une future « probable » anomalie correspond à la détection d'un changement (de niveau, de pente, de variance...) dans la courbe temporelle du score d'anomalie. Le problème est donc équivalent à la détection de « rupture » dans un processus temporel.

Détecter une anomalie est indispensable, mais une fois l'anomalie détectée, c'est-à-dire en cas d'alerte, il est nécessaire d'identifier sa cause, ce qui correspond à la

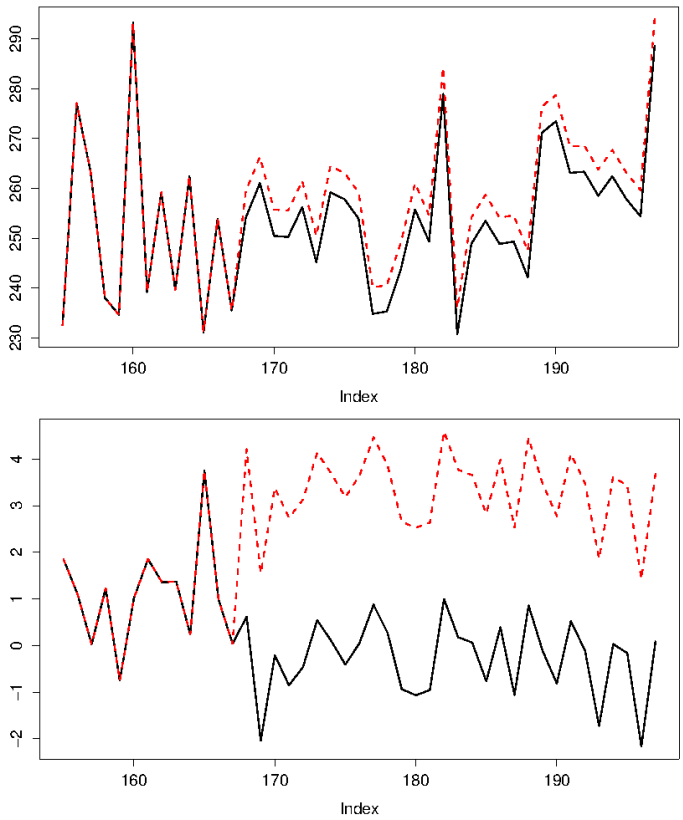


FIGURE *3.4: Deux signaux avant et après une normalisation mathématique. Leurs différences sont beaucoup plus marquées après normalisation.*

(*Source : Bellas et al. (2012).*)

dernière étape. On sait que les moteurs sont complexes et que démonter inutilement une partie du moteur peut être très coûteux : il faut donc tout faire pour l'éviter. Le problème de l'identification de la panne fait l'objet de la section 3.4.

## 3.3 Détection d'anomalies à Snecma

Pour aider à la détection des anomalies, Snecma a développé des algorithmes de détection dont les résultats, en cas d'alerte, sont destinés à l'opérateur métier. Il s'agit d'applications complètes qui vont de la collecte des données jusqu'à l'identification des composants fautifs, en passant par la transformation des données en indicateurs de défauts. Pour chaque faute potentielle, les experts ont identifié les quantités physiques (ou des transformations de ces quantités physiques) à surveiller.

### Normes OSA-CBM

Chaque application élaborée par les experts pour surveiller un sous-système est conçue sur un même schéma.

Plus précisément, les algorithmes déjà en place à Snecma respectent une architecture générique. Cette architecture est un standard appelé OSA-CBM [1] (*Open Systems Architecture for Condition-Based Maintenance*) et est structurée en plusieurs couches (voir le schéma 3.5). Cela a l'avantage de faciliter la compréhension des applications par les utilisateurs et d'en permettre l'amélioration continue couche par couche. Par exemple, on peut voir dans la figure 3.6 la structure de l'algorithme utilisé à Snecma dans le cadre de la surveillance de la consommation d'huile. Cela permet également de faciliter l'échange des applications, comme c'est le cas dans le Groupe Safran.

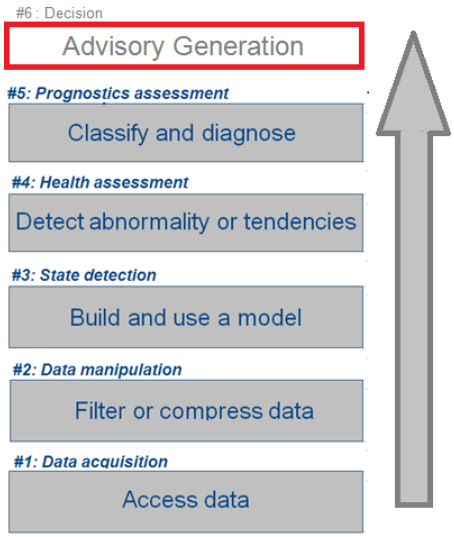


FIGURE *3.5: Schéma représentant les couches de la norme OSA-CBM.*

(*Source : http://www.mimosa.org/?q=node/361*.)

1. http://www.mimosa.org

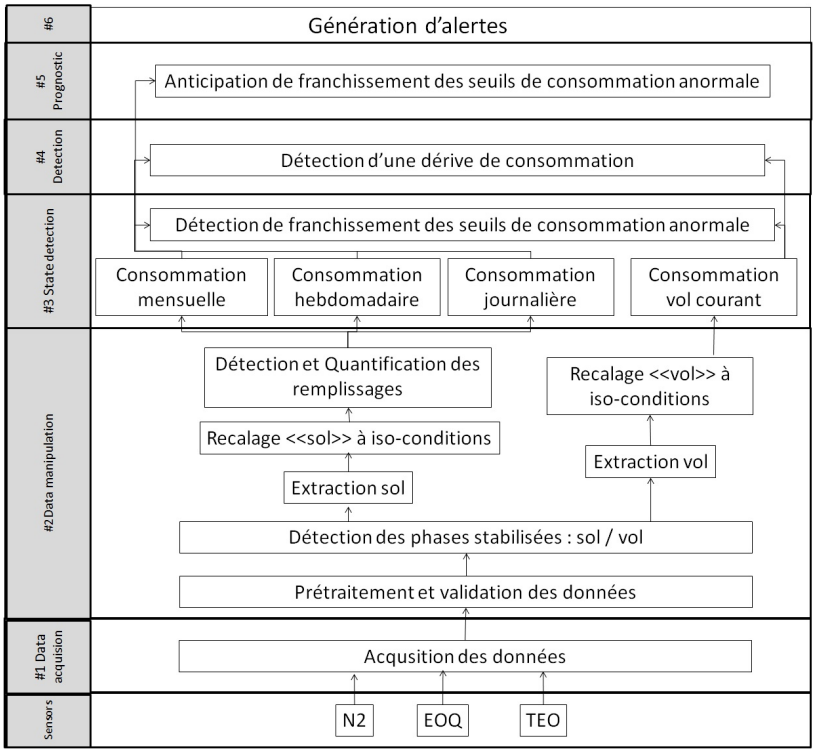


FIGURE *3.6: Schéma représentant la structure de l'algorithme utilisé dans le cadre de la surveillance de la consommation d'huile.*

(*Source : document interne Snecma.*)

## 3.4 Identification des pannes

Dans cette partie, on s'intéresse à l'étape d'identification des pannes.

Les variables d'intérêt ont été déterminées par les experts de façon à discriminer les différents états du moteur. Les experts ont tenté une classification des types de panne en associant à chaque type des tendances observées sur les variables d'intérêt. Le tableau 3.1 donne un extrait de cette classification. Par exemple, pour la valve HPTACC (*High Pressure Turbine Active Clearance Control*), qui contrôle des flux d'air dans le but d'améliorer efficacité du carburant, les experts ont observé qu'une détérioration de cette valve HPTACC correspond à trois phénomènes simultanés sur les variables DEGT, DFF et DN2 (le D signifie que c'est la différence entre la valeur théorique et celle effectivement mesurée qui est considérée) :

- une augmentation de la température de l'air à la sortie de la turbine haute pression (variable DEGT),
- une augmentation du flux du carburant dans la chambre de combustion (variable DFF),
- une diminution de la vitesse de rotation de l'arbre haute pression (variable DN2).

TABLE *3.1: Schéma représentant un extrait de classification par signatures. Les flèches indiquent le sens de variation de chacun des paramètres pour chaque cause probable de la panne.*

| Cause probable | DEGT (˚C) | DFF (%) | DN2(%) |
|---|---|---|---|
| *Bleed Leakage* | $\uparrow$ | $\uparrow$ | $\uparrow$ |
| *Bleed valve indication* | $\uparrow$ | $\uparrow$ | $\uparrow$ |
| *Core Deterioration* | $\uparrow$ | $\uparrow$ | $\downarrow$ |
| *Deicing fluid* | $\uparrow$ | $\uparrow$ | $\downarrow$ |
| *EGT increase* | $\uparrow$ | $\approx 0$ | $\approx 0$ |
| *Engine deterioration* | $\uparrow$ | $\uparrow$ | $\downarrow$ |
| *Fuel Flow Indication* | $\uparrow$ | $\approx 0$ | $\approx 0$ |
| *HPTACC Valve* | $\uparrow$ | $\uparrow$ | $\downarrow$ |
| *Pack Flow Controller* | $\uparrow$ | $\uparrow$ | $\uparrow$ |
| *VBV System* | $\uparrow$ | $\uparrow$ | $\uparrow$ |

(*Soucre : Document interne Snecma.*)

Lorsque l'on observe de telles variations, on peut donc suspecter une anomalie de la valve HPTACC. Cette observation se fait sur les données de vol comme celles représentées sur la figure 3.7 sur laquelle on peut voir l'évolution du DEGT (différence entre la valeur théorique et la valeur mesurée de la température à la sortie de la turbine haute pression). L'opérateur métier désigne l'instant de rupture (en rouge) et détermine

la variation de la mesure considérée entre l'instant de rupture et l'instant d'alerte (en violet).

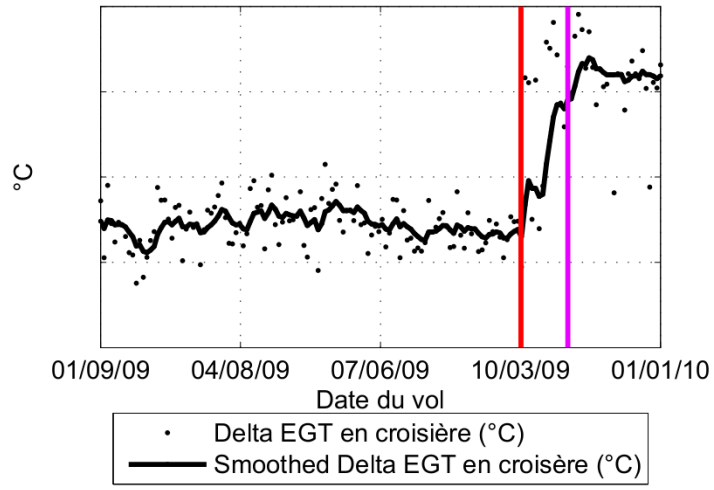


FIGURE *3.7: Exemple de données de vol surveillées. La différence d'ordonnées du paramètre surveillé (ici DEGT) entre l'instant de rupture (en rouge) et l'instant d'alerte (en violet) est un exemple de valeur utilisée lors d'un diagnostic (voir le tableau 3.1)*

(*Source : Snecma document interne.*)

En résumé, pour chaque type de panne, les experts ont identifié des « signatures » probables.

Une autre forme de représentation plus précise des signatures est donnée par la figure 3.8. Cette représentation a pour avantage d'être un peu plus précise car on prend en compte plus de variables et l'on considère également l'amplitude des variations.

Par exemple, dans le cas de la perte de rendement turbine basse pression, on peut observer que cela correspond à des évolutions simultanées sur 6 variables :

- une augmentation importante de la pression (PS3) et du débit carburant (WFE),
- une augmentation modérée de la vitesse de rotation de l'arbre haute pression (XN25), de la température en entrée de la chambre de combustion (T3) et de la température à la sortie de la turbine haute pression (T495),
- une baisse de la température en sortie du booster (T25).

Des procédés, notamment de visualisation, ont été élaborés pour aider à l'identification du type de panne. Une méthode simple consiste à projeter les données sur deux axes. Ces axes ont été identifiés par les experts comme permettant de séparer différents types d'anomalies et plus généralement les données normales des données anormales. Par exemple, dans la figure 3.9 présente les visualisations obtenues à l'aide d'un outil de projection élaboré à Snecma. Cet outil est interactif : les plans de projections peuvent être changés par les experts. Dans cette figure, ces derniers ont choisi de projeter sur

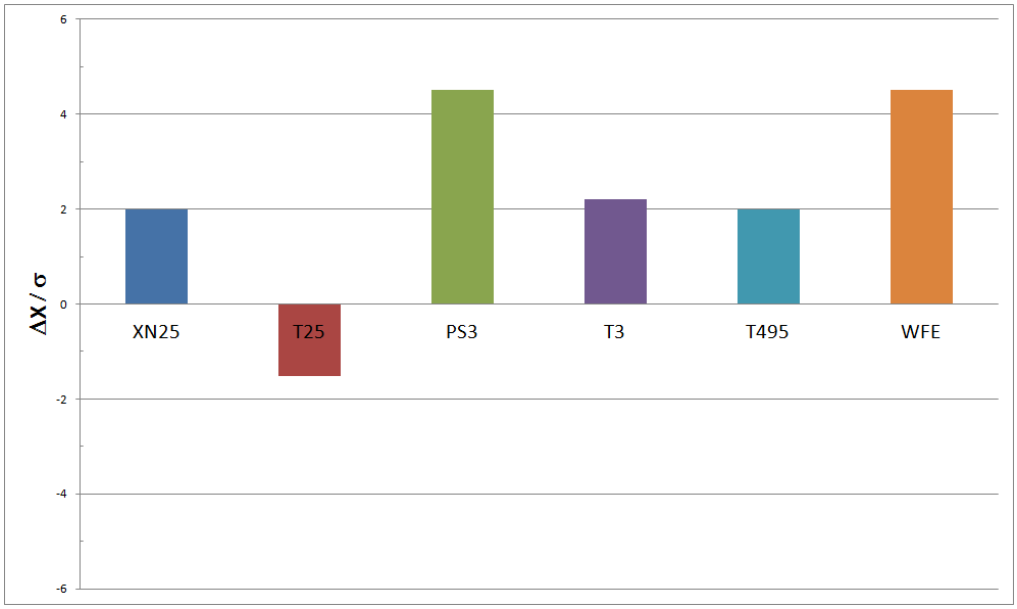


FIGURE *3.8: Signature d'une perte de rendement turbine basse pression : la perte de rendement turbine basse pression donne lieu à une augmentation importante des variables.*

(*Source : Snecma document interne.*)

deux variables d'intérêt les axes correspondant au gradient maximum de l'EGT lors de la phase de démarrage (Grad(EGT)), et la durée $t\_alum$ mesurée entre l'instant correspondant au début d'injection du carburant dans la chambre de combustion et l'instant correspondant au point d'inflexion visible sur la courbe N2 (voir figure 3.2).

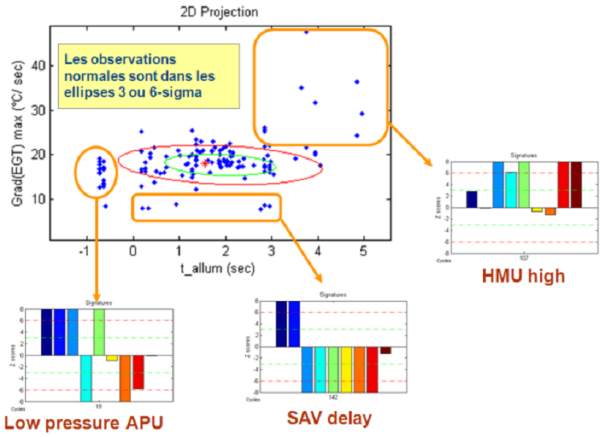


FIGURE *3.9: Exemple de plan de projection pour aider à l'identification de l'origine des anomalies. Les plans de projection peuvent être changés par les experts.*

Parmi les méthodes de projection, on peut citer également la cartographie qui est étudiée à Snecma et décrite dans Côme et al. (2010). La cartographie permet de déterminer l'état d'un moteur à partir de sa position sur la carte. Cette méthode consiste à projeter les variables normalisées sur une carte de Kohonen et d'identifier les différentes zones de la carte.

## 3.5 La prise de décision

Dans l'état actuel du développement de détection, on exploite le savoir expert à plusieurs niveaux : pour la détermination des variables d'intérêt, pour la suppression des effets de contexte, pour l'identification des signatures ou des axes de projection des données, et pour la prise de décision.

Le rôle de l'opérateur métier est d'étudier les résultats de tous les algorithmes de détection disponibles pour la prise de décision. Malgré l'aide apportée par les visualisations, la prise de décision reste difficile. La frontière délimitant deux états moteurs reste souvent floue. Ce flou peut avoir plusieurs origines : un mauvais choix des variables d'intérêt ou des axes de projection. De plus, on remarque que les méthodes actuelles n'utilisent pas l'ensemble des informations disponibles. En effet, les experts sont souvent spécialisés dans un sous-système et fournissent des alertes dont la synthèse est difficile à faire. C'est à l'opérateur d'agréger tous les résultats alors que les dépendances entre les sous-systèmes, lorsqu'il y a une anomalie, ne sont pas toujours connues. Une difficulté vient du fait qu'il y a beaucoup de moteurs à suivre et peu de temps pour prendre des décisions.

Ainsi l'opérateur doit prendre en compte l'ensemble des résultats de différentes procédures qui aboutissent à des messages qui peuvent être concordants ou non. L'apport de la thèse consiste à proposer une méthode aidant l'opérateur à fusionner différentes procédures (voir la figure 3.10). Cette méthode est présentée dans le chapitre suivant.

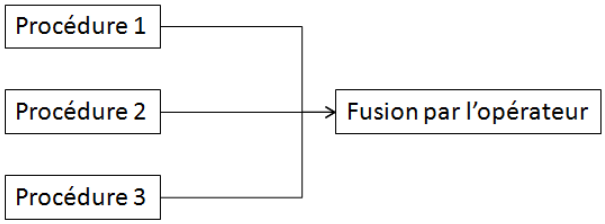


FIGURE *3.10: Actuellement la fusion des informations est faite par les opérateurs métier. Le principal objectif de la thèse est d'assister les opérateurs à fusionner ces informations.*

CHAPITRE 4

# Fusion d'indicateurs

Les chapitres précédents ont montré comment étaient élaborées les applications de surveillance pour un moteur d'avion. On a pu remarquer l'importance des experts métier lors du processus d'élaboration de ces applications. Leur rôle va du choix des mesures à prélever jusqu'à la calibration des applications de surveillance en passant par le choix des variables d'intérêt et par la description du type de changement attendu pour un type d'anomalie.

En outre, les experts sont souvent spécialistes d'un sous-système ; les algorithmes élaborés ne portent alors que sur des sous-systèmes, et les relations de dépendance entre ces sous-systèmes ne sont pas toujours prises en compte parce qu'elles ne sont pas bien connues. Ainsi, il n'est pas évident de demander à un expert ou à un opérateur métier de considérer un ensemble de diagnostics portant sur des sous-systèmes différents et de lui demander de conclure sur l'état de santé du système entier.

Actuellement lorsqu'une alerte est émise, l'opérateur métier doit agréger toutes les informations disponibles pour faire son analyse. Celle-ci se base essentiellement sur l'évolution des quantités physiques normalisées par le contexte. En cas de doute, les scores d'anomalie calculés par les algorithmes élaborés par les experts peuvent l'aider. Cependant, ces différentes informations collectées peuvent être parfois difficiles à comprendre ; elles peuvent contribuer à confirmer un diagnostic mais également fournir des résultats contradictoires.

Il y a donc besoin de disposer d'une méthodologie aidant l'opérateur métier à faire une synthèse de toutes les informations disponibles. Plus précisément, il s'agit de définir une méthodologie permettant de fusionner les informations de différentes sources de façon à améliorer la confiance lors de la prise de décision. Le rôle de la fusion est de permettre la réduction des incertitudes, de faire émerger des informations inconnues pour aider à faire une analyse globale du système. Un des objectifs de ce travail de thèse est donc d'aider à définir un processus de fusion.

Ce chapitre décrit les exigences à respecter pour répondre aux besoins de Snecma, puis présente le processus d'homogénéisation des données, étape nécessaire avant la fusion.

## 4.1 Difficultés rencontrées sur les solutions actuelles et exigences

### 4.1.1 Difficultés liées à la calibration

L'expérience montre que la calibration des détecteurs d'anomalie doit être effectuée régulièrement, notamment pour limiter l'occurrence des fausses alarmes. Cependant, ces réglages sont des tâches loin d'être triviales. Différents paramètres doivent être considérés comme par exemple, l'échelle à laquelle le changement peut être visible, ou encore l'historique à considérer pour la recherche d'une anomalie. Les experts peuvent indiquer les quantités physiques à surveiller et les transformations nécessaires pour assurer la surveillance. Mais ils ne peuvent pas toujours calibrer précisément l'ensemble de ces paramètres pour éviter les fausses alarmes.

Par exemple, dans le cas particulier d'une détection de rupture, une fois tous les paramètres de l'algorithme de détection de rupture définis, il reste à positionner les seuils d'alerte. On utilise pour cela en général un test statistique en calculant une *p-value* à partir des valeurs observées. Rappelons qu'une *p-value* est la probabilité qu'en situation normale, la valeur de la statistique soit plus grande que la valeur observée. Cette *p-value* peut alors être comparée à un seuil pour élaborer un test automatique de détection de rupture. Une des difficultés est alors de déterminer le niveau de seuil adéquat.

#### 4.1.1.1 Confirmation

La *p-value* étant très sensible aux données aberrantes, les experts métier ajoutent une phase de *confirmation* pour limiter le nombre de fausses alarmes. Elle consiste par exemple, à mesurer la persistance temporelle de dépassement de seuil avant d'émettre une alerte.

Différents types de mesures sont possibles. Une confirmation couramment utilisée est la confirmation ($k$ parmi $n$), c'est-à-dire qu'une alerte est émise uniquement si, sur les $n$ dernières observations, il y en a au moins $k$ au delà du seuil. Ce type de confirmation est utilisé par exemple dans Hmad et al. (2013) et Massé et al. (2014). On voit que le choix du type de confirmation utilisée ainsi que le choix des paramètres font également partie des réglages à effectuer.

La confirmation permet de diminuer le nombre de fausses alarmes. En effet, supposons qu'il y ait à chaque instant une probabilité $p$ qu'un score $S$ dépasse le seuil $S_e$ en l'absence d'anomalie. Alors en ajoutant l'étape de confirmation de type ($k$ parmi $n$), avec $k > \lfloor \frac{n}{2} \rfloor$, en supposant les scores indépendants, la probabilité d'avoir une alerte au vu des $n$ dernières observations, en l'absence d'anomalie, se calcule à l'aide de la queue d'une loi Binomiale :

$$\sum_{k \leq i \leq n} C_n^i p^i (1-p)^{n-i}.$$

Par exemple, si la probabilité qu'un score dépasse le seuil est de $5\%$, avec une

confirmation ($3$ parmi $5$), alors on se trouve dans le cas où $p = 5\%, n = 5$ et $k = 3$, et la probabilité d'avoir une alerte en l'absence d'anomalie avec l'étape de confirmation passe de $5\%$ à $0,12\%$.

Ce qu'il est intéressant de remarquer dans cet exemple, c'est qu'il n'y a pas de perte de taux de bonne détection. Avec ces mêmes valeurs et avec une probabilité d'alerter quand il y a une anomalie de 95%, avec une étape de confirmation, la probabilité d'avoir une alerte en cas d'anomalie passe à plus de $99\%$.

Dans les cas où l'on dispose d'un ensemble de couples (*probabilité d'alerter*, *probabilité d'avoir une alerte en l'absence d'anomalie*), il peut être intéressant de comparer les courbes de ROC pour chaque type de confirmation et faciliter le choix de la meilleure configuration.

#### 4.1.1.2 Dispersion des informations

Une difficulté importante réside dans la dispersion des données qui sont disponibles sur plusieurs bases. Par exemple, lorsqu'une inspection ou un lavage sont faits sur un moteur, l'information ne remonte pas toujours jusqu'aux experts ou aux opérateurs métier qui s'occupent de la surveillance du moteur. De la même façon, lorsqu'une recommandation est envoyée à une compagnie cliente, le retour d'expérience n'est pas toujours obtenu : les conséquences de cette recommandation ne sont pas toujours transmises par la compagnie cliente. Cette dispersion de l'information complique l'obtention de données « labellisées » et l'utilisation de méthodes supervisées qui sont pourtant nécessaires pour faire émerger des règles de fusion, aider à la calibration, évaluer les applications de surveillance, etc. Cette dispersion peut également rendre plus difficile l'analyse de l'opérateur métier qui n'a alors qu'une vision partielle de l'état du moteur.

### 4.1.2 Exigences à respecter pour de nouvelles solutions

Pour prendre une décision et faire des recommandations à une compagnie aérienne, les opérateurs métier et les experts doivent avoir confiance dans les indications données par les algorithmes. Ils doivent pouvoir les confronter aux retours d'expérience et se les approprier. La difficulté apportée par l'ajout de nouveaux d'indicateurs vient du fait qu'ils sont par définition nouveaux et qu'a priori les experts ne leur font pas tout de suite entièrement confiance.

Ainsi pour qu'une méthode nouvelle soit pertinente, il faut que le processus de formation des indicateurs soit compréhensible par les utilisateurs pour qu'ils puissent interpréter les résultats fournis avec confiance.

Toute nouvelle solution proposée doit être facile à comprendre, doit exploiter au maximum les informations disponibles et fournir la meilleure décision possible, c'est-à-dire rappelons le, une bonne probabilité de détection et un faible taux de fausse alarme.

## 4.2 Processus d'homogénéisation

Avant de pouvoir procéder à la fusion, il faut homogénéiser les données. Après une description des différents niveaux de fusion et des données à fusionner, on présente la méthode d'homogénéisation proposée dans cette thèse.

### 4.2.1 Niveau de fusion

On a vu jusqu'à présent que les informations disponibles sont de nature très variée. Chacune des couches OSA-CBM (voir la section 3.3) fournit en sortie des informations. Celles-ci vont des mesures brutes issues des capteurs aux résultats des applications de surveillance (indicateurs binaires d'alerte ou score d'anomalie).

Dans Tsitsiklis et al. (1993), Tsitsiklis cite différents niveaux de fusion possibles :

- au niveau des mesures où un centre de fusion collecte et stocke toutes les informations ;
- au niveau des variables d'intérêt ;
- au niveau des sorties d'algorithmes de détection où il s'agit alors de fusionner des décisions.

Dans le cadre de la surveillance des moteurs d'avion, une fusion qui serait faite au niveau des données brutes ne serait pas pertinente, puisque les experts simplifient l'analyse de l'état du moteur à partir de variables d'intérêt qu'ils ont eux même construites et identifiées, et qui sont un résumé des mesures brutes.

Les opérateurs métier utilisent les variables d'intérêt de plusieurs façons :

- ils peuvent suivre leurs évolutions après normalisation par le contexte et peuvent, dans le cadre de la détection automatique, définir des seuils d'alerte ;
- ils peuvent utiliser une application de surveillance qui leur fournira un score monodimensionnel ou une décision binaire sur la présence éventuelle d'une anomalie.

Dans ce dernier cas, on se trouve plutôt dans le dernier niveau de fusion proposé dans Tsitsiklis et al. (1993).

Dans le cadre de notre travail, à la suite des remarques faites dans la section 4.1 sur la calibration, nous proposons une méthode d'*homogénéisation* qui permet de combiner les deux derniers niveaux de fusion proposés dans Tsitsiklis et al. (1993).

### 4.2.2 Données labellisées et définition d'une anomalie

Pour évaluer une méthode, par exemple pour justifier qu'elle est meilleure qu'une autre, il est nécessaire d'avoir des données labellisées, le label pouvant être la présence ou l'absence d'anomalie.

Un opérateur métier considère qu'un moteur est anormal lorsqu'il constate une rupture de stationnarité : un changement dans l'évolution des variables d'intérêt considérées. Par exemple, dans le cas où il surveille l'écart lissé entre l'EGT (*Exhausted Gas Temperature*) prédite et l'EGT mesurée en croisière (DEGT), il va considérer qu'il y a anomalie s'il voit un changement de tendance ou un changement de variance.

Dans le cadre d'un algorithme de surveillance élaboré par les experts, les variables d'intérêt sont parfois converties en un score d'anomalie ou en toutes autres mesures traduisant l'écart à la mesure prévue en absence d'anomalie. Un seuil est alors défini, et il est considéré que le moteur présente une anomalie dans le cas où ce seuil est dépassé. Le score est souvent sensible aux données aberrantes, et un dépassement peut être dû à une erreur dans la chaîne de calcul du score. Une étape de confirmation temporelle est donc généralement ajoutée pour obtenir un label fiable (voir la section 4.1.1.1).

Dans le cadre de l'inspection visuelle d'un moteur, un début de dégradation peut être constaté. On peut supposer que cette dégradation pourrait se voir sur les données avant cette constatation.

Dans tous les cas, il est plus pertinent d'associer une anomalie à un intervalle plutôt qu'à un instant $t$. Il semble donc plus intéressant de considérer une série temporelle précédant l'instant $t$, et d'associer le label « anomalie » ou pas à cette série temporelle.

À Snecma, détecter une anomalie est crucial mais ce n'est pas suffisant. Une fois une anomalie détectée, il faut également identifier son origine. Les labels « anomalie » et « absence d'anomalie » ne sont donc pas suffisants.

Par conséquent, il faut enrichir le dictionnaire des labels. La première difficulté réside dans le fait que les labels sont souvent propres à un opérateur, ce qui a tendance à multiplier le nombre de labels pour un même type d'anomalie. De plus, un label défini par un opérateur $A$ peut ne pas être compréhensible par un autre opérateur $B$. Par exemple, un opérateur $A$ qui inspecte physiquement un moteur ne va pas nécessairement donner le même label qu'un opérateur $B$ qui analyse les données de ce même moteur. Pour que le dictionnaire des labels garde une taille raisonnable, il faut définir un ensemble de labels communs à tous les opérateurs en les uniformisant. Les anomalies sont rares et comme une minimisation du nombre de labels permet une diminution du nombre de classes à distinguer, cela permet en outre de limiter la quantité de données d'apprentissage nécessaires.

En résumé, pour chaque moteur, on peut, pour chaque instant $t$, extraire une série temporelle multivariée correspondant à un intervalle de temps précédant $t$. Chaque dimension de la série représente une mesure pré-traitée ou non, une variable d'intérêt, ou encore un score d'anomalie, ou l'information « un seuil a été dépassé ». Dans cette série temporelle, on peut également avoir des informations de type *water-wash* (nettoyage du moteur), *in-flight shutdown* (arrêt en vol),... C'est en analysant la série temporelle avant l'instant $t$ sur une ou plusieurs dimensions que l'opérateur peut décider du label à considérer et peut l'appliquer au moteur à l'instant $t$ (voir figure 4.1).

### 4.2.3 Normalisation et détection de rupture

L'hypothèse faite ici est que les résidus obtenus par la normalisation, décrite dans la section 3.2, sont stationnaires et indépendants en l'absence d'anomalie, et qu'il y a une rupture de la stationnarité dans le cas où une anomalie se présente.

Formellement, sous cette hypothèse, si l'on considère par exemple la surveillance

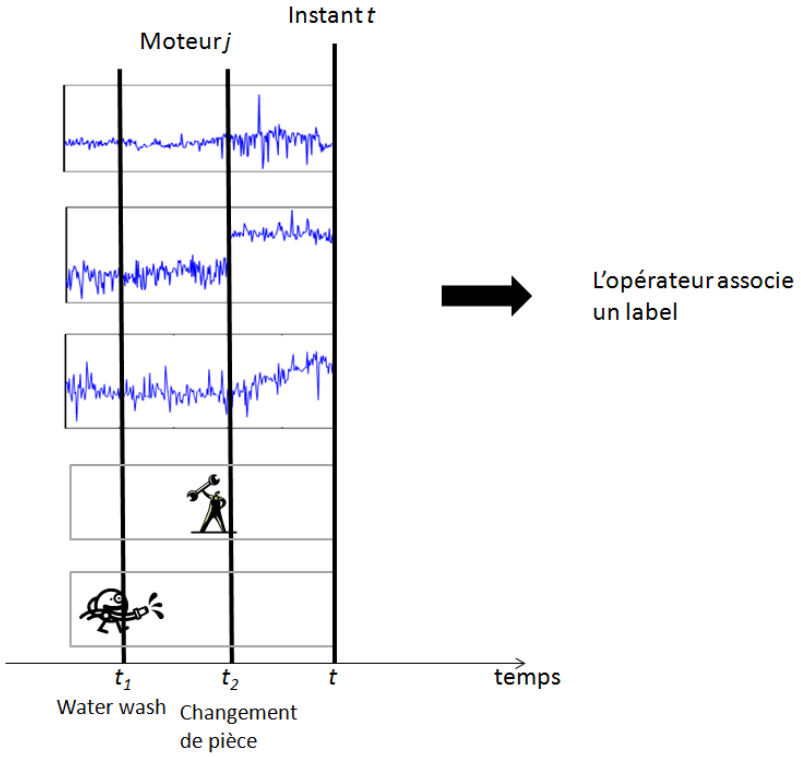


FIGURE *4.1: A un instant $t$, l'opérateur applique un label à un moteur $j$ après analyse des données précédant $t$ et informations de ce moteur. Notons qu'à l'instant $t_1$, le moteur a subi un water-wash.*

d'un système à partir d'une série temporelle $Z_1, \ldots, Z_n$, alors les $Z_i$ sont supposés identiquement distribués en absence d'anomalie. Et sous cette hypothèse, détecter une anomalie équivaut à détecter un changement dans la distribution des $Z_i$ à un moment $t$. Cela permet de profiter des nombreuses méthodes de détection de rupture existantes, comme celles qui sont énumérées dans Basseville et Nikiforov (1993). Si les experts peuvent indiquer le type de changement attendu, un algorithme peut alors être mobilisé.

Par exemple, si l'expert suggère qu'il faut surveiller les changements dans la moyenne des résidus, on peut se retrouver face à deux cas typiques :

- soit l'expert affirme que les résidus suivent une loi gaussienne avec une variance fixée, alors l'outil naturel sera un test de Student ;
- soit l'expert n'a pas d'a priori sur la distribution des résidus, alors un test de Mann Whitney Wilcoxon peut être adapté à la détection.

Et si l'expert considère qu'il faut surveiller les changements dans la variance des résidus et que les résidus suivent une loi gaussienne, alors on peut utiliser un test de Fisher de comparaison des variances décrit à la section 5.2.1.

### 4.2.4 Homogénéisation

Chacune des dimensions des séries temporelles que l'on considère peut avoir une nature différente des autres. Les natures possibles que l'on peut retenir ici sont :

1. Variables d'intérêt indépendantes du contexte extérieur :

   Ces données sont généralement celles qui sont directement analysées par les opérateurs métier. Ils recherchent des changements dans l'évolution des variables d'intérêt pour repérer une éventuelle anomalie et identifier, le cas échéant, l'origine de l'anomalie.

2. Détecteurs automatiques d'anomalie élémentaires :

   Pour chaque variable d'intérêt, des seuils sont définis. Dès que les seuils sont dépassés, une alerte est émise et la composante concernée est alors binaire.

3. Scores d'anomalie des applications algorithmiques de surveillance :

   Pour compléter l'analyse des opérateurs métier, différentes applications de surveillance de sous-systèmes du moteur ont été développées par les experts métier. Ces applications fournissent généralement des scores (probabilités) d'anomalie. On étudie également les tendances de ces scores.

4. Décision des applications algorithmiques de surveillance :

   Les experts cherchent à définir des seuils d'alerte, avec confirmation éventuelle, pour chaque application de surveillance. On obtient ainsi des détecteurs automatiques pour chaque sous-système considéré, et là encore les composantes concernées sont binaires.

La liste donnée n'est pas exhaustive, elle pourrait être complétée sans remettre en question la suite. Pour traiter mathématiquement l'ensemble de ces dimensions, on a besoin de les rendre homogènes.

La solution proposée est de ne considérer que les décisions, dimension par dimension, c'est-à-dire de résumer chacune des dimensions en des valeurs binaires $\{0, 1\}$, correspondant à {pas d'anomalie, anomalie}. Cette binarisation permet de résumer un ensemble de $N$ séries temporelles dans un ensemble de $N$ vecteurs binaires comme illustré dans la figure 4.2.

Pour les cas de type 2 et 4, les valeurs sont déjà binaires par définition. Pour les résidus normalisés (cas 1) et les scores (cas 3), on définit un (ou des) test statistique qui fournit une (ou des) décision binaire :

- 1 lorsque « le seuil a été dépassé », c'est-à-dire lorsque « le test est significatif » ;
- 0 lorsque « le seuil n'a pas été dépassé », c'est-à-dire lorsque « le test n'est pas significatif ».

A cela, on peut ajouter une étape de confirmation (voir 4.1.1.1).

Plus précisément, pour construire un test de rupture, on considère différents instants $s$ précédant l'instant $t$, sur un historique et une échelle d'observation. Considérer

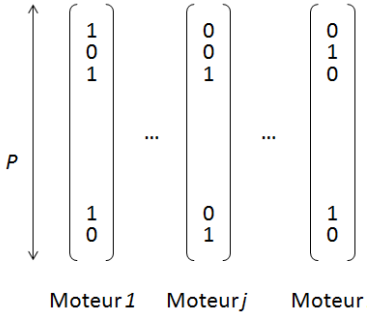


FIGURE *4.2: Chaque série temporelle (associée à un moteur) peut être transformée en un vecteur binaire.*

l'historique dans son intégralité n'est pas nécessairement pertinent ni réalisable : on se restreint à un historique limité dans le passé. Considérer un seul instant $s$ et faire un seul test sur cet historique n'est pas non plus adapté parce que l'on perdrait l'aspect temporel. Pour contourner cette difficulté, la solution proposée est de procéder à un test de rupture séquentiel : on considère une fenêtre glissante

$$[s-\frac{w}{2}, s+\frac{w}{2}],$$

où $w$ est la taille de la fenêtre. Puis on applique un test d'homogénéité aux deux ensembles de valeurs de part et d'autre du milieu de la fenêtre. La figure 4.3 illustre le procédé utilisé pour le test de rupture. Ce procédé peut s'apparenter à un test de rupture *online*, mais il est supposé que l'ensemble des observations sont disponibles et non pas récupérées à la volée.

Formellement,

- on considère l'hypothèse $H_{0(s)}$, les observations ont la même loi sur tout l'intervalle $[s-\frac{w}{2}, s+\frac{w}{2}]$
- sous l'hypothèse alternative $H_{1(s)}$, les observations suivent deux lois différentes sur $[s-\frac{w}{2}, s[$ et $[s, s+\frac{w}{2}]$.

Une rupture est détectée en $s$ lorsque le test de l'hypothèse $H_{1(s)}$ contre l'hypothèse $H_{0(s)}$ est significatif. Ainsi, une alerte peut être émise dès qu'il existe une fenêtre qui détecte une rupture. Une étape de confirmation peut être ajoutée. Différents types de confirmation, décrits au chapitre 5, sont possibles. La série temporelle peut être filtrée au préalable, par exemple avec un filtre gaussien ou une moyenne mobile. Pour améliorer la vitesse d'exécution de la recherche de rupture, la fenêtre glissante peut se déplacer avec un pas plus ou moins important. Ainsi, une augmentation de la valeur du pas permet de diminuer les temps de calcul, mais aussi d'assurer une plus grande indépendance entre les tests de deux fenêtres successives en limitant le recouvrement de ces fenêtres.

En résumé, après l'homogénéisation, les observations sont transformées en vecteurs binaires. Dans la figure 4.2, les vecteurs représentent des moteurs différents ou

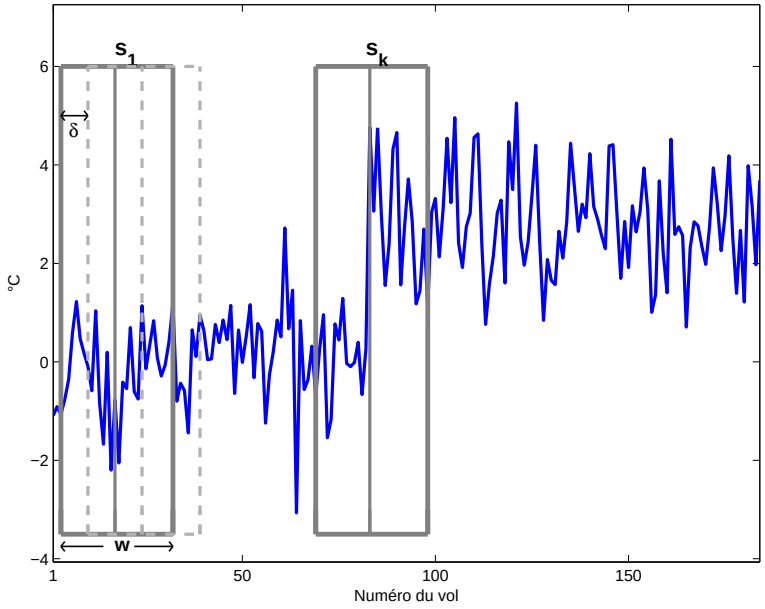


FIGURE *4.3: Illustration du procédé d'un test de rupture séquentiel : un test d'homogénéité est appliqué aux deux ensembles de valeurs de part et d'autre du milieu d'une fenêtre qui parcourt le signal.* $\delta$ *est le pas entre deux fenêtres consécutives pour lesquelles le test est fait. Dans la fenêtres* $s_1$*, le test devrait être négatif. Dans la fenêtre* $s_k$*, le test devrait être positif.*

observés à des instants différents. (Les instants sont suffisamment éloignés pour respecter l'indépendance.)

### 4.2.5 Exploration de l'espace des paramètres

En fait, pour chacune des notions introduites précédemment :
- définition des variables d'intérêt,
- distinction entre les variables de contexte et variables de fonctionnement,
- type et paramètres des méthodes de normalisation,
- niveau du seuil de détection,
- type et paramètres de confirmation,
- largeur des fenêtres glissantes,
- etc.

Il reste à définir de manière adéquate les paramètres et méta-paramètres.

Par exemple, le choix du *snapshot* et même des variables d'intérêt et de la normalisation ne sont pas évidents. Si on s'intéresse à la variable N1 par exemple (vitesse de rotation de l'arbre basse pression), elle peut selon les cas être considérée comme une

variable de contexte parce qu'elle est commandée par le pilote ou de fonctionnement parce qu'elle caractérise l'état du moteur.

De même, comme il a été vu dans la section 4.1.1, les niveaux de seuil doivent constamment être réajustés en fonction du retour d'expérience (RetEx). Et comme on le voit dans la figure 4.4, l'émission d'alerte dépend du niveau retenu. De la même façon, l'expert ne sait pas toujours quel type de confirmation ou quel type de test statistique mettre en œuvre.

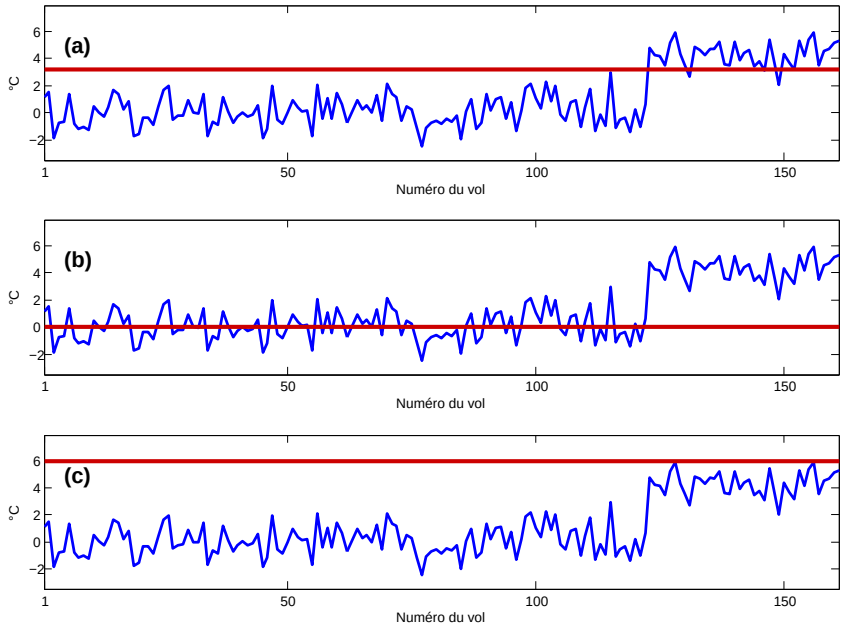


FIGURE *4.4: Exemples de 3 niveaux de seuils d'alerte possibles pour un même signal ; (a) est un niveau qui semble adéquat ; (b) est un niveau qui semble favoriser les fausses alarmes ; (c) est un niveau qui semble trop élevé pour les détections.*

La solution proposée dans cette thèse consiste à considérer un sous-ensemble fini de toutes les combinaisons des paramètres ou méta-paramètres compatibles avec les attentes des experts. Autrement dit, l'expert définit d'abord une plage pour les paramètres (type de normalisation, niveau du seuil, type de confirmation, historique considéré, échelle et type de lissage, etc.). Alors pour chaque combinaison de ces modalités, un indicateur est engendré. Il correspond à une décision parmi d'autres. Cet indicateur est à valeur dans $\{0, 1\}$ comme expliqué dans la section 4.2.4 définissant le processus d'homogénéisation.

En résumé, à partir d'un grand nombre de combinaisons de ces paramètres et méta-paramètres, des indicateurs binaires sont engendrés. Cet ensemble d'indicateurs forme un vecteur binaire de très grande dimension. Dans la suite de la thèse, la dimension de ce vecteur est de l'ordre de plusieurs milliers : on note $P$ le nombre d'indicateurs binaires

engendrés.

## 4.3 Décision par discrimination

Il s'agit maintenant de prendre une décision globale en fusionnant l'ensemble des indicateurs disponibles.

Si l'on cherche seulement à déterminer si une anomalie est présente ou non, on peut définir deux classes pour les moteurs :

– une classe $C_0$ qui est composée des moteurs qui ne présentent pas d'anomalie ;
– une classe $C_1$ qui est composée des moteurs présentant une anomalie.

Mais si l'on s'intéresse au type d'anomalie (ou à l'origine de l'anomalie), on est amené à considérer un nombre de classes égale au nombre de types d'anomalie + 1 [1].

Formellement, on considère donc $K$ classes $\{C_0, C_1, \ldots, C_{K-1}\}$ [2].

Alors, pour un moteur donné représenté par un vecteur de tous ses indicateurs, il s'agit simplement de définir la classe à laquelle il appartient. Le problème de *fusion des indicateurs* du moteur se ramène donc à un problème de *classification*.

Pour clarifier le vocabulaire utilisé, on s'autorise l'anglicisme *classification* jusqu'à la fin de la thèse pour désigner la classification supervisée. Il ne faut pas confondre ce terme avec sa version française qui correspond à la classification non supervisée (*clustering* en anglais).

Et quant aux notations, on utilise désormais les notations classiques de l'apprentissage statistique proposées par exemple dans Hastie et al. (2009). De cette façon, on a des notations plus générique et on désigne désormais les observations faites sur les moteurs (variables explicatives) par $x_i$ et les labels (variables à expliquer) par $y_i$.

### 4.3.1 Classification supervisée

On considère un couple de variables aléatoires $(X, Y)$ où pour toute réalisation $i$ :

– $x_i$ est un vecteur binaire de taille $P$ : $x_i \in \{0,1\}^P$,
– $y_i \in \{0, \ldots, K-1\}$.

On a affaire à un problème de classification à $K$ classes.

On note $x^1, \ldots, x^P$ les composantes de chaque observation $x_i$ :

$$x_i = \begin{pmatrix} x^1 \\ \vdots \\ x^P \end{pmatrix}$$

1. On peut remarquer qu'il peut y avoir plusieurs modes de fonctionnement normaux. On pourrait donc également avoir différentes classes pour les cas normaux.

2. On peut remarquer que pour faciliter l'interprétabilité par les experts métier, on peut définir un dictionnaire de labels $\mathscr{L}$ noté :

$$\mathscr{L} = \{L_0, L_1, \ldots, L_{K-1}\}$$

où il y aurait une bijection entre les classes et les labels, et chaque label serait une définition « métier » de la classe, fournie par les experts.

Pour la phase d'apprentissage, on suppose que l'on dispose d'une série de $N$ réalisations d'un couple de variables aléatoires que l'on note

$$\mathscr{D} = ((x_1, y_1), \ldots, (x_N, y_N)).$$

Le but est de prédire la sortie $y_i$ à partir de l'entrée $x_i$, autrement dit, il s'agit d'estimer, à partir de l'échantillon d'apprentissage, une fonction inconnue $f$ vérifiant $f(x) = y$.

Une fois cette fonction $f$ estimée par $\hat{f}$, la prédiction $\hat{y}$ de $y$ est calculée par $\hat{f}(x)$.

Pour prendre en compte l'incertitude de la prédiction, plutôt que de s'intéresser à des méthodes déterminant directement $y$, on va plutôt considérer des méthodes fournissant $\mathbb{P}(Y|X = x)$ et choisir la classe telle que :

$$\hat{y} = \hat{f}(x) = \underset{0 \leq k \leq K-1}{\arg\max}\, \mathbb{P}(y = k|X = x)$$

C'est-à-dire que l'on choisit, pour la réalisation $x$, la classe la plus probable sachant $x$.

Pour simplifier les notations, on se restreint à un problème à deux classes. Le but est de minimiser la probabilité de faire une erreur (en attribuant une mauvaise classe à l'observation), c'est-à-dire de minimiser

$$\mathbb{P}(erreur|x^1, \ldots, x^P) = \mathbb{P}(y = 1|x^1, \ldots, x^P)$$

dans le cas où l'on décide que la classe est $0$ et minimiser

$$\mathbb{P}(erreur|x^1, \ldots, x^P) = \mathbb{P}(y = 0|x^1, \ldots, x^P)$$

dans le cas où on décide que la classe est $1$.

Le choix optimal est celui qui va minimiser cette probabilité d'erreur.

Ainsi, si

$$\mathbb{P}(y = 0|x^1, \ldots, x^P) > \mathbb{P}(y = 1|x^1, \ldots, x^P),$$

on décide que c'est la classe $0$ qui est attribuée à la nouvelle observation, sinon on décide $1$.

En résumé on a :

$$\forall i,\ \hat{f}(x_i) = k_{max},$$

où

$$k_{max} = \underset{k}{\arg\max}\, \mathbb{P}(y = k|x^1, \ldots, x^P).$$

Par la suite, on utilise des méthodes robustes adaptées aux données en grande dimension ($P$ grand devant $N$) dans le but de minimiser les erreurs de classification. Dans notre travail, nous avons utilisé essentiellement la méthode des Random Forests et les classifieurs bayésiens naïfs. Les deux sections suivantes sont une introduction à ces deux méthodes. Pour cette partie, on se limitera au cas particulier de deux classes ($K = 2$ et $y \in \{0, 1\}$).

### 4.3.2 Random Forests

#### 4.3.2.1 Des arbres de décision aux Random Forests

Différentes méthodes de classification permettent de résoudre ce problème de discrimination. Certaines ont pour avantage d'être facilement appréhendables par les experts métier, ce qui est par exemple le cas des méthodes de type *arbres*.

Les arbres partitionnent l'espace $\{0,1\}^P$ en $M$ régions $(R_m)_{1\leq m\leq M}$. La forme des régions se limitent à des rectangles (en dimension 2) éventuellement avec une arête infinie mais avec pour condition que toutes les arêtes soient parallèles aux axes. A chaque région $R_m$ de l'espace est associée un modèle simple qui est la classe majoritairement présente dans notre problème. Autrement dit, si $x_i$ appartient à la région $R_m$, et que la classe majoritaire de $R_m$ est $k$ alors on a

$$\hat{y_i} = \hat{f}(x) = k.$$

Pour chaque région $R_m$, la probabilité $\mathbb{P}(y = k|X = x)$ peut être estimée empiriquement par la fréquence d'apparition de la classe $k$ dans la région.

Avant de présenter la méthode CART (*Classification and Regression Trees*) développée par Breiman dans Breiman et al. (1984), et qui procède par une partition récursive et unique de l'espace, on commence par illustrer un des avantages clés des arbres de décision : le processus de décision peut se lire directement sur un arbre où les régions sont les feuilles de l'arbre. Une fois l'arbre construit, pour chaque nouvelle donnée, il suffit de déterminer sa position dans l'arbre pour lui associer un label.

Considérons par exemple l'arbre donné dans la figure 4.5 dans un problème à deux classes. Chaque feuille (numérotée pour les besoins de l'illustration) correspond à une région. La couleur permet d'identifier la classe majoritaire de la région : vert pour la classe $0$ et rouge pour la classe $1$.

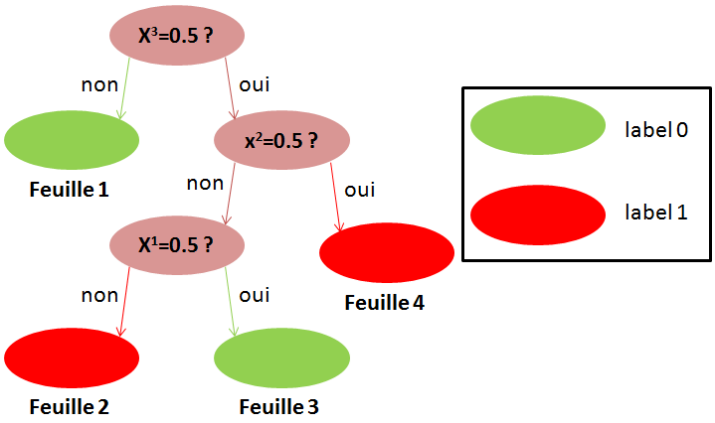


FIGURE *4.5: Exemple d'arbre binaire avec des variables binaires. On appelle feuilles les nœuds terminaux de l'arbre. Les feuilles sont numérotées de 1 à 4.*

Par exemple, considérons l'élément $x = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$ dont on veut déterminer la classe.

Pour cela, il suffit de parcourir l'arbre en partant de la racine, et de répondre à la question de chaque nœud rencontré jusqu'à ce que l'on se retrouve dans une feuille. Ainsi pour $x$, la question à la racine est :

« Est ce que $x_3 = 1$ ? ».

Ce n'est pas le cas : on se trouve alors à la feuille numéro 1 de couleur verte et donc on lui affecte le label $0$.

On peut remarquer que la construction de l'arbre n'est pas toujours triviale : d'abord parce qu'il n'y a pas unicité des arbres (plusieurs arbres peuvent parfaitement correspondre aux données d'apprentissage), et ensuite parce qu'un arbre peut devenir rapidement trop complexe (en considérant par exemple le nombre de feuilles comme mesure de complexité). Il faut donc choisir une complexité se situant à un équilibre entre le sur-apprentissage et le sous-apprentissage.

La méthode CART proposée par Breiman et al. (1984) partitionne récursivement l'espace de manière unique et en limitant le sur-apprentissage. Cette méthode se fait en deux temps. On décrit la méthode dans le cas binaire :

1. Construction de l'arbre de manière récursive jusqu'à avoir des feuilles pures.
2. Élagage de l'arbre de façon à minimiser une fonction coût dont la pénalisation est la complexité de l'arbre.

La construction se déroule comme suit : pour chaque feuille, deux nouvelles feuilles sont créées et les données sont réparties dans les nouvelles feuilles. Pour cela, la méthode CART choisit la variable qui minimise une mesure d'impureté, ce qui a pour conséquence de diminuer l'erreur de classification en séparant le plus possible les données de classes différentes (Hastie et al. (2009)).

Un exemple de mesure d'impureté couramment utilisée est l'indice de diversité de Gini qui, pour une feuille $m$ d'un arbre $\mathscr{T}$, est donné par :

$$Q_m(\mathscr{T}) = \sum_{0 \leq k \leq K-1} p_{mk}(1 - p_{mk}).$$

où $p_{mk}$ est la proportion de données avec la classe $k$ dans la région $R_m$.

Dans notre cas à deux classes, en notant $p_m$ la proportion de moteur avec le label 1, l'expression de cette mesure d'impureté devient :

$$Q_m(\mathscr{T}) = 2p_m(1 - p_m).$$

Il existe d'autres mesures d'impureté possibles comme le taux de mauvaise classification, mais l'avantage de l'indice de Gini est sa propension à favoriser les feuilles pures, c'est-à-dire qu'à taux d'erreur de classification égale, l'indice de Gini favorise les discriminations pour lesquelles une des feuilles est la plus pure possible.

Pour illustrer cela, on peut reprendre l'exemple de Hastie et al. (2009) où l'on a un problème à deux classes avec 400 individus (ou moteurs) dans chaque classe. Si l'on suppose qu'une division permet de créer les feuilles $(300,100)$ et $(100,300)$ (c'est-à-dire que la première feuille et la deuxième feuille ont respectivement $300$ et $100$ individus de la classe $0$ puis $100$ et $300$ de la classe $1$) et qu'une deuxième division permet de créer les feuilles $(200,400)$ et $(200,0)$, alors on obtient le même taux de mauvaise classification dans les deux cas. Mais l'indice de Gini favorise la deuxième division qui propose une feuille pure.

La construction de $\mathscr{T}$ continue de façon récursive à chaque nouvelle feuille jusqu'à ce que la mesure d'impureté totale $Q(\mathscr{T}) = \sum_m Q_m(\mathscr{T})$ passe sous un seuil défini à l'avance. Dans le cas où le seuil est $0$, toutes les feuilles contiennent des individus appartenant à une même classe : elles sont pures.

Ainsi, à la fin de cette première étape de construction, on dispose d'un arbre permettant une discrimination adéquate des moteurs. Mais cet arbre a, par construction, un grand nombre de feuilles $|\mathscr{T}|$ qui croit exponentiellement avec le nombre de niveaux. Pour diminuer cette complexité et donc limiter le sur-apprentissage, on procède ensuite à la deuxième étape qui consiste à élaguer l'arbre tout en cherchant à minimiser une fonction de coût pénalisée par la complexité.

Un exemple de fonction de coût couramment utilisée est donné par :

$$C_\alpha(\mathscr{T}) = \sum_{m=1}^{|\mathscr{T}|} N_m Q_m(\mathscr{T}) + \alpha|\mathscr{T}|,$$

où

- $N_m$ est le nombre d'éléments dans la feuille $m$ ;
- $\alpha$ est un paramètre à optimiser qui donne plus ou moins de poids à la pénalité ($\alpha = 0$ donne un arbre complexe).

Pour finir, on peut signaler, comme le font remarquer les auteurs dans Hastie et al. (2009), que les arbres ont pour défaut d'être très instables, c'est-à-dire qu'un petit changement de l'échantillon d'apprentissage peut complètement changer la structure de l'arbre, ce qui rend moins évidente l'interprétabilité de l'arbre de décision. De plus, les performances se dégradent lorsque la dimension $P$ des variables augmente.

#### 4.3.2.2 Précisions sur les Random Forests

Les Random Forests, développés par Breiman et décrits dans Breiman (2001) proposent une solution pour pallier ces inconvénients. Il s'agit d'une méthode de classification de type *bagging* par arbres (voir Breiman (1993)). Elle consiste à construire un ensemble d'arbres décorrélés entre eux, fournissant chacun une classification, et ensuite à procéder à un vote. On sait que la décorrélation entre les arbres est nécessaire pour pouvoir utiliser un système de classification par vote comme indiqué dans Ruta et Gabrys (2005) et Kuncheva et Whitaker (2003).

Plus précisément, à partir de l'échantillon étudié,

$$\mathscr{D} = ((x_1, y_1), \ldots, (x_N, y_N)),$$

on crée un nombre $B$ d'échantillons d'apprentissage de taille $N$ par *bootstrap*. Chaque échantillon *bootstrap* est obtenu par un tirage avec remise dans l'échantillon initial. Pour chaque échantillon *bootstrap*, on crée un arbre de décision de type CART. L'arbre est construit jusqu'à ce que chaque classe soit homogène, c'est-à-dire jusqu'à ce que tous les éléments d'une feuille aient la même classe, mais en utilisant uniquement un nombre $Q << P$ de variables tirées au hasard parmi l'ensemble des $P$ variables disponibles.

La classification finale se fait alors par un vote non pondéré : chaque moteur est associé à la classe majoritaire dans les arbres obtenus à partir des échantillons *bootstrap*.

Lors du processus de *boostrap*, on met de côté une proportion des données de l'échantillon initial (souvent environ $\frac{1}{3}$), ces données mises de côté sont appelées *out-of-bag* (OOB). Les échantillons « bootstrapés » ne contiennent donc aucune de ces données OOB. L'échantillon OOB est utilisé pour une estimation de l'erreur de classification. Ce procédé permet de se passer d'une étape de validation croisée.

L'échantillon *out-of-bag* (OOB) est également utilisé pour déterminer l'importance d'une variable. Pour cela, les valeurs d'une variable sont permutées de façon aléatoire, et une nouvelle estimation de la classe est faite par chaque arbre. On soustrait au nombre d'arbres qui, sans permutation, donnaient la bonne classe, le nombre d'arbres estimant la bonne classe après permutation. En moyennant, on obtient alors une mesure de l'importance de la variable. Autrement dit, plus il y a d'erreurs suite à la permutation, plus la variable est importante.

Les données étant binaires, elles se retrouvent toutes sur les sommets d'un cube.

Les Random Forests sont couramment utilisées, notamment à Snecma dans Ricordeau et Lacaille (2010) dans le cadre du *Health Monitoring*. Leurs bons résultats sont confirmés sur des données réelles, comme le constatent les auteurs dans Fernández-Delgado et al. (2014) qui ont évalué différentes méthodes. Les Random Forests ont fourni les meilleurs résultats et dans la suite de notre travail, ils servent de méthode de référence.

Cependant, un des défauts reste la difficulté d'interprétation. Ce défaut est majeur dans le cadre du *Health Monitoring* des moteurs d'avions. Les mesures d'importance des variables permettent de fournir une indication dans le choix de la classe, mais pourraient ne pas être suffisantes pour satisfaire l'opérateur métier, du moins dans un premier temps.

### 4.3.3 Classifieur bayésien naïf

Pour contourner les problèmes d'interprétabilité des Random Forests, une des solutions est d'utiliser les classifieurs bayésiens naïfs qui sont également adaptés aux vecteurs de grande dimension.

On suppose que l'on dispose d'un échantillon d'apprentissage :

$$\mathscr{D} = ((x_1, y_1), \ldots, (x_N, y_N)).$$

Ce sont les vecteurs $x \in \{0,1\}^P$ que l'on veut classifier en utilisant une approche générative, pour cela on a besoin de connaître $p(x|y = k)$.

Dans le cas du classifieur bayésien naïf, une hypothèse simplificatrice est que l'on suppose les variables $x^1, \ldots, x^P$ indépendantes conditionnellement à la classe.

Ce qui permet alors de factoriser $p(x|y=k)$ tel que :

$$p(x|y=k) = \Pi_{1\leq j\leq P} p(x^j|y=k).$$

Étant dans un cas binaire, les $x^j$ sont des Bernoulli, ainsi la loi de $x$ conditionnellement à la classe est :

$$p(x|y=k) = \Pi_{1\leq j\leq P} Ber(x^j|\theta_{jk})$$

où $\theta_{jk}$ est la probabilité que la variable $x^j$ soit positive pour la classe $k$.

Ainsi, pour la prédiction, on rappelle que d'après la formule de Bayes, on a pour chaque classe $k$ :

$$\mathbb{P}(y=k|x^1,\ldots,x^P) = \frac{\mathbb{P}(x^1,\ldots,x^P|y=k)\times\mathbb{P}(y=k)}{\mathbb{P}(x^1,\ldots,x^P)}.$$

Puis, en faisant remarquer que la vraie valeur du dénominateur, c'est-à-dire la probabilité de la réalisation $x$, ne dépend pas de la classe, on peut se restreindre au numérateur :

$$\mathbb{P}(y=k|x^1,\ldots,x^P) \propto \mathbb{P}(x^1,\ldots,x^P|y=k)\times\mathbb{P}(y=k).$$

Ce qui donne après factorisation (hypothèse *naïve* du modèle) :

$$\mathbb{P}(y=k|x^1,\ldots,x^P) \propto \mathbb{P}(y=k)\prod_{j=1}^{P}\mathbb{P}(x^j|y=k).$$

Il suffit donc, pour la prédiction, de calculer :

$$\forall k, \mathbb{P}(y=k)\Pi_{1\leq j\leq P} Ber(x^j|\theta_{jk}).$$

Étant dans le cas binaire ($k\in\{0,1\}$), pour la densité a priori de la classe, l'estimation de $\mathbb{P}(y=k)$ par maximum de vraisemblance est donnée par la proportion des réalisations ayant la classe $k$.

Et $\mathbb{P}(x^j=1|y=k)=\theta_{jk}$ est estimé par la proportion des valeurs de la variable $x^j$ (valant 0 ou 1) dans la classe $k$ (on rappelle qu'on a des Bernouilli).

La limite à l'utilisation de l'estimation par maximum de vraisemblance est le surapprentissage. Ce phénomène peut par exemple s'observer lorsque que l'une des variables $x^j$ de l'échantillon d'apprentissage prend toujours la même valeur $0$, respectivement la valeur $1$, dans les deux classes, autrement dit l'estimation de $\mathbb{P}(x^j|y=k)$ est $0$, respectivement $1$, pour les deux classes. Et dans le cas où une nouvelle observation a $x^j$ qui vaut $1$, respectivement $0$, alors il y a un cas indéterminé car on aurait $p(y=k|x)=0$ pour les deux classes.

Ainsi pour contourner cela, on propose d'utiliser une densité a priori pour les paramètres $\theta_{jk}$ des Bernouilli. Souvent, c'est une $Beta(1,1)$ qui est utilisée pour chaque $\theta_{jk}$ (ce qui correspond à la correction de Laplace) et donc on a pour estimation :

$$\hat{\theta}_{jk} = \frac{\#\{x^j = 1, y = k\} + 1}{\#\{y_k\} + 2}$$

Et de la même façon, on corrige la densité a priori de la classe et l'estimation du paramètre se fait par :

$$\frac{\#\{y = k\} + 1}{N + 2}.$$

Bien que les performances restent inférieures à celles de Random Forests comme ont pu le constater les auteurs dans Caruana et Niculescu-Mizil (2006), la classification bayésienne naïve est tout de même étudiée pour plusieurs raisons :

- Sa rapidité d'exécution. En effet, les quantités $P(y = k)$ et $\mathbb{P}(x^j | y = k)$ peuvent être calculées pour chaque valeur de $k$ et de $x^j$ possibles et stockées une fois pour toute.
- Son interprétabilité. En effet, contrairement aux Random Forests où l'influence d'une variable $x^j$ ne se lit qu'avec les variables d'importance associées, dans le cadre du classifieur bayésien naïf, on dispose de la probabilité sachant la classe qu'un indicateur soit positif (=+1). Ces valeurs peuvent aider l'opérateur métier à comprendre le modèle.

Malgré l'hypothèse (naïve) d'indépendance des indicateurs sachant la classe, entraînant des approximations lors du calcul des probabilités conditionnelles , les labels obtenus sont satisfaisants. Cette méthode donne de meilleures performances que certains modèles plus complexes (voir par exemple Koller et Friedman (2009) ou Hastie et al. (2009)).

Avant de finir, présentons un exemple illustrant le cas où l'estimateur du maximum de vraisemblance n'est pas adéquat. Soit un échantillon d'apprentissage comportant $N = 7$ observations en dimension $P = 3$ telles que :

$$(x_1, x_2, x_3, x_4, x_5, x_6, x_7) = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

avec pour labels

$$(y_1, y_2, y_3, y_4, y_5, y_6, y_7) = (0, 0, 0, 0, 1, 1, 1)$$

Le tableau 4.1 donne la probabilité qu'une variable vaille $1$ sachant la classe, c'est-à-dire que pour chaque variable $x^j$ et pour chaque classe $0$ et $1$, le tableau donne $\mathbb{P}(x^j = 1 | y = k)$ avec $k \in \{0, 1\}$. Chacun des termes du tableau est, rappelons le, déterminé par la proportion des valeurs de la variable $x^j$ (valant 0 ou 1) dans la classe $k$. Ainsi, en notant # l'opération de comptage, on a . :

$$\mathbb{P}(x^j = 1 | y = k) = \frac{\#\{x^j = 1, y = k\}}{\#\{y = k\}}.$$

Et c'est cette estimation qui permet alors de compléter les tableaux 4.1 et 4.2.

TABLE *4.1: Tableau donnant la probabilité qu'un indicateur vaille 1 sachant la classe.*

| | $y=0$ | $y=1$ |
|---|---|---|
| $x^1=1$ | $\frac{3}{4}$ | $\frac{1}{3}$ |
| $x^2=1$ | $\frac{1}{2}$ | $\frac{1}{3}$ |
| $x^3=1$ | $\frac{1}{4}$ | $1$ |

TABLE *4.2: Tableau donnant la probabilité qu'un indicateur vaille 0 sachant la classe.*

| | $y=0$ | $y=1$ |
|---|---|---|
| $x^1=0$ | $\frac{1}{4}$ | $\frac{2}{3}$ |
| $x^2=0$ | $\frac{1}{2}$ | $\frac{2}{3}$ |
| $x^3=0$ | $\frac{3}{4}$ | $0$ |

Ainsi, pour déterminer la classe d'une observation $x = \begin{pmatrix} x^1 \\ x^2 \\ x^3 \end{pmatrix}$, il suffit de prendre la colonne adéquate du tableau 4.1 et de la multiplier par la probabilité de la classe.

On considère une nouvelle réalisation $x_8 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$

Le tableau 4.3 donne les probabilités des $x^1$, $x^2$ et $x^3$ suivant que l'on suppose $x_8$ dans la classe $0$ ou la classe $1$.

TABLE *4.3: Tableau donnant les probabilités conditionnelles de l'observation $x_8$.*

| | $y=0$ | $y=1$ |
|---|---|---|
| $x^1=0$ | $\frac{1}{4}$ | $\frac{2}{3}$ |
| $x^2=1$ | $\frac{1}{2}$ | $\frac{1}{3}$ |
| $x^3=0$ | $\frac{3}{4}$ | $0$ |

Or, sachant que dans notre exemple, on a :

$$\mathbb{P}(y=0) = 1 - \mathbb{P}(y=1) = \frac{4}{7},$$

on obtient donc en exploitant l'hypothèse d'indépendance pour chaque classe $k \in \{0,1\}$ :

$$\mathbb{P}(y=k|x^1,x^2,x^3) \propto \mathbb{P}(y=k) \times \mathbb{P}(x^1|y=k) \times \mathbb{P}(x^2|y=k) \times \mathbb{P}(x^3|y=k)$$

Ce qui donne numériquement :

$$\mathbb{P}(y=0) \times \mathbb{P}(x^1|y=0) \times \mathbb{P}(x^2|y=0) \times \mathbb{P}(x^3|y=0) = 3/56$$

$$\mathbb{P}(y=1) \times \mathbb{P}(x^1|y=1) \times \mathbb{P}(x^2|y=1) \times \mathbb{P}(x^3|y=1) = 0$$

On conclut donc que $y=0$ est la classe la plus probable de $x_8$. Mais ce que l'on veut illustrer avec cet exemple, c'est un cas où, dans l'échantillon d'apprentissage, il y a une valeur de variable qui n'apparaît pas dans l'une des classes. Dans l'exemple, aucune des observations appartenant à la classe $1$ n'a la variable $x^3$ valant 1. Ce phénomène a pour conséquence, lors d'une estimation, que cette classe ne sera jamais sélectionnée dès qu'une observation aura cette variable avec sa valeur (dans notre exemple, ce serait dans le cas où l'on a une observation avec $x^3=1$), et cela indépendamment des valeurs des autres variables.

Une erreur peut même être possible si, dans l'échantillon d'apprentissage, il existe une variable qui est toujours la même quelque soit la classe. Ces cas sont des illustrations du sur-apprentissage et expliquent l'intérêt d'une utilisation de la correction de type Laplace.

### 4.3.4 Résumé

Le problème de détection d'anomalie a été converti en un problème de classification supervisée. L'avantage d'une classification supervisée est qu'elle permet de retrouver les relations entre les variables d'intérêt et les dépendances souvent inconnues entre les indicateurs.

L'information disponible est résumée en un ensemble d'apprentissage de couples (moteur, label). Ces informations ont été converties de façon à pouvoir être exploitées par des algorithmes d'apprentissage automatique. Ainsi, la méthode proposée permet d'intégrer le savoir expert à travers l'utilisation des applications de surveillance.

Les deux algorithmes, Random Forests et le classifieur bayésien naïf, ont été choisis pour leurs performances, leur interprétabilité et leur vitesse d'exécution. (Voir le tableau comparatif 4.4)

Pour améliorer l'interprétabilité des résultats qui est une exigence fondamentale de Snecma, et compte tenu de la grande redondance des indicateurs utilisés, nous allons étudier par la suite comment pratiquer la sélection de variables.

TABLE *4.4: Tableau résumant les avantages du Random Forests et du classifieur Bayésien Naïf.*

| | Random Forests | Classifieur Bayésien Naïf |
|---|---|---|
| Rapidité apprentissage | | + |
| Rapidité exécution | | ++ |
| Interprétabilité | | + |
| Robustesse | + | + |
| Performances | ++ | |

## 4.4 Sélection des indicateurs

Dans notre étude, il y a un nombre très élevé de variables utilisées, ce qui peut favoriser le sur-apprentissage.

De plus, le processus de formation des variables (indicateurs) décrit dans la section 4.2, implique de fortes redondances. Autrement dit, lorsque l'on considère un ensemble d'observations $x_1, \ldots, x_N$, où $x_i = (x_i^1, \ldots, x_i^P)$, il peut arriver qu'il existe $j$ et $j'$ tels que :

$$x_i^j = x^{j'}.$$

Or, dans le cadre de la classification bayésienne naïve, cette redondance infirme l'hypothèse d'indépendance des indicateurs conditionnellement à la classe, ce qui peut fortement dégrader les résultats obtenus.

On peut également faire remarquer que certains indicateurs peuvent se révéler être sans intérêt. Par exemple, des indicateurs peuvent être définis sur des seuils trop faibles (voir l'exemple (b) de la figure 4.4 p. 36).

Ces fortes redondances et le manque d'intérêt de certaines variables allongent inutilement les temps de calcul.

### 4.4.1 Apports de la sélection de variables

La sélection de variables permet de traiter les difficultés citées précédemment, notamment en cherchant à supprimer les variables qui ne sont pas pertinentes pour notre problème de classification. Elle permet alors d'améliorer la robustesse des méthodes décrites.

Un autre avantage de la sélection de variables est de permettre l'interprétabilité des résultats et donc leur exploitation par les experts métier. En effet, même si les variables (c'est-à-dire les indicateurs) sont simples à appréhender et qu'un classifieur tel que le bayésien naïf est simple à comprendre, il n'est pas réaliste de demander à un expert de passer en revue des centaines, voire des milliers de variables. Dans le cadre particulier des Random Forests, bien que cette méthode propose une mesure de l' importance des variables et une sélection implicite des variables à travers la construction des arbres, une sélection explicite des variables peut améliorer l'interprétabilité des résultats.

En résumé, la sélection de variables permet d'améliorer l'interprétabilité et ainsi que les temps de calcul en cherchant à supprimer les variables redondantes et les moins pertinentes : cela rend alors la méthode de classification plus robuste.

On rappelle que l'on se place dans un cadre supervisé, c'est-à-dire que l'on dispose de $N$ observations :

$$\mathscr{D} = ((x_1, y_1), \ldots, (x_N, y_N)).$$

On peut cependant signaler que des méthodes de sélection variables non supervisées existent. Dans ce cas, la sélection se fait à partir de l'élaboration d'un classement des variables à partir d'un score calculé selon un critère qui leur est propre (variance, entropie...). Une fois le classement fait, il suffit de garder les variables avec les meilleurs scores.

Ce procédé consistant à classer les variables peut s'étendre aux cas supervisés en calculant un score basé sur une mesure de pertinence liant la variable et la classe (corrélation, information mutuelle...). Cela permet d'avoir des scores plus pertinents, malgré l'inconvénient suivant : le score calculé de chaque variable est indépendant du comportement de toutes les autres. Par conséquent, une sélection de ce type ne supprime pas les redondances évoquées dans l'introduction. Elle ne prend pas non plus en compte le fait que deux variables peuvent avoir un faible score prises individuellement, mais que prises ensembles, elles peuvent parfaitement expliquer la classe.

Malgré tout, cette première approche a pour avantage d'être peu gourmande en calculs. Mais il peut y avoir un intérêt à trouver des stratégies de sélection offrant de meilleurs sous-ensembles de variables expliquant la classe. On peut faire remarquer qu'il y a $P$ variables et donc potentiellement $2^P - 1$ sélections possibles.

Dans la pratique, dès que $P$ est grand, il n'est pas envisageable d'explorer toutes les configurations possibles et on limite donc le nombre de sous-ensembles testés en utilisant des méthodes de sélection heuristiques ou aléatoires des indicateurs.

Mais avant d'aborder les stratégies de recherche des sous-ensembles de variables, il peut être intéressant de signaler qu'il y a différents critères possibles pour évaluer la pertinence de ces sous-ensembles, Dans Guyon et Elisseeff (2003), les auteurs citent deux approches générales pour la sélection des indicateurs :

– Approche par filtre ;
– Approche *wrapper*.

Par la suite, on présente brièvement différentes approches possibles décrites notamment dans Guyon et Elisseeff (2003).

### 4.4.2 Approches par filtre et *wrapper*

Dans l'approche par filtre, le critère de sélection utilisé est indépendant de l'algorithme de classification. Un classement des variables est effectué à partir d'un critère qui est indépendant du modèle. Cette approche peut être supervisée ou non.

Tandis que l'approche *wrapper* est nécessairement supervisée. Dans cette approche le critère de sélection est lié au modèle ; ainsi, le score obtenu par un sous-ensemble

de variables est lié à la capacité de prédiction de ce sous-ensemble sur un échantillon d'apprentissage.

Les exemples donnés dans la section 4.4.1 (variance, corrélation, ...) sont des illustrations d'approches par filtre mais la nature très redondante des variables rend a priori l'utilisation de ces exemples peu pertinente. En effet, des variables égales ($x^j = x^{j'} \Rightarrow \forall 1 \leq i \leq N,\ x_i^j = x_i^{j'}$) ont le même classement. Or, sélectionner deux variables égales n'apporte pas plus d'informations que de n'en sélectionner qu'une seule.

Ainsi, même s'il peut être intéressant de signaler que dans le cas du classifieur bayésien naïf, de bonnes performances peuvent être obtenues (malgré son hypothèse forte, voir Murphy (2012) ou Koller et Friedman (2009) par exemple), celles-ci peuvent être améliorées :

- lorsque les probabilités conditionnelles sont bien estimées,
- lorsqu'une bonne sélection de variables a été pratiquée de manière à se rapprocher de l'hypothèse naïve d'indépendance.

Dans Boullé (2007), l'auteur sélectionne les variables pour lesquelles l'hypothèse naïve est vérifiée, ce qui permet de contourner les difficultés rencontrées dans le cas de données complexes en grande dimension.

D'autres solutions par filtre sont disponibles comme la sélection utilisant des arbres de décision. Cette méthode utilise un classement des variables basé sur le nombre de fois où elles apparaissent sur les nœuds principaux d'un arbre. Cette méthode serait efficace couplée avec un classifieur bayésien naïf (voir Ratanamahatana et Gunopulos (2003) mais elle sur-apprend et est coûteuse en calculs (voir Boullé (2007)).

On peut également analyser les dépendances entre les variables en utilisant l'information mutuelle qui permet de détecter les relations non linéaires entre les variables. Elle a aussi l'intérêt de pouvoir être définie pour un ensemble de variables contrairement à la corrélation qui ne se calcule que pour des couples.

On verra (voir la section 4.4.3 sur les stratégies de recherches) que l'on peut exploiter ces propriétés dans des processus de sélection séquentielle de type *forward* / *backward* par exemple. Et d'ailleurs, bien que le choix du sous-ensemble de variables maximisant l'information mutuelle ne soit pas équivalent à la minimisation de la probabilité de mauvaise classification, ce choix conduit à des résultats très satisfaisants comme l'ont démontré Frenay et al. dans Frénay et al. (2012)), Frénay et al. (2013a) et Frénay et al. (2013b).

Dans le cadre de la thèse, c'est essentiellement une approche *forward* avec un critère basé sur l'information mutuelle qui est utilisée.

### 4.4.3 Stratégies de recherche

#### Sélection séquentielle *forward* avec une approche par filtre

Dans un algorithme de sélection séquentielle *forward*, les variables sont sélectionnées, une par une, de façon à maximiser un paramètre à chaque étape. Par exemple, l'algorithme appelé mRMR (*Max-Relevance, and Min-Redundancy*) ou CMIM (*Condi-*

*tional Mutual Information Maximization criterion*) présenté dans (Peng et al. (2005) et Fleuret (2004)) cherche à sélectionner les indicateurs qui expliquent au mieux la variable cible, sous la contrainte, qu'à chaque étape, la variable candidate à la sélection soit la moins redondante possible avec chacune des variables déjà sélectionnées.

Ainsi, si on note :

– $X^j, 1 \leq j \leq P$, la variable aléatoire binaire à valeurs dans $\{0,1\}$,

– $\nu(m)$ le numéro de la $m$-ième variable sélectionnée,

– $Y$ la variable aléatoire dont la valeur observée est la classe $k \in \{0,1\}$,

alors les deux algorithmes suivent le schéma de sélection ci-dessous :

$$\nu(1) = \arg\max_j I(Y;X^j)$$

$$\forall m > 1, \quad \nu(m+1) = \arg\max_j \min_{s \leq m} I(Y;X^j|X^{\nu(s)}).$$

Dans les définitions ci-dessus, $I(X,Y)$ est l'information mutuelle entre $Y$ et $X$ qui vaut :

$$I(Y;X) = H(Y) + H(X) - H(Y,X),$$

et $I(Y;X^j|X)$ est l'information mutuelle entre $Y$ et $X^j$ conditionnellement à $X$ qui vaut :

$$I(Y;X^i|X) = H(Y|X) - H(Y|X^i,X),$$

La fonction $H(X)$ (resp. $H(Y|X)$) est l'entropie de la variable aléatoire $X$ (resp. l'entropie conditionnelle). Ici, les variables $X^j$ sont des variables de Bernoulli à valeur dans $\{0,1\}$. On note $p_j$ le paramètre de la variable $X^j$ et on l'estime par la fréquence empirique de l'événement $(X^j = 1)$.

L'entropie est donnée par

$$H(X^j) = -\mathbb{E}_{p_j}[\ln L(X^j, p_j)].$$

où $L(X^j)$ est la vraisemblance de $X^j$. Dans le cas binaire, l'entropie est estimée par :

$$H(X^j) = -p_j \ln(p_j) - (1-p_j)\ln(1-p_j).$$

De la même façon, on peut calculer l'entropie du couple $(Y,X)$ :

$$H(Y,X^j) = -\mathbb{E}_{p_j}[\ln L(Y,X^j, p_j)]$$

en estimant les probabilités $\mathbb{P}(Y = k, X^j = x)$ et l'entropie conditionnelle $H(Y|X^j)$ en estimant les probabilités $\mathbb{P}(Y = k|X^j = x)$.

Bien que moins performante qu'une méthode utilisant un critère de sélection prenant en compte l'information mutuelle de la variable sélectionnée avec toutes les variables

déjà sélectionnées, cette méthode est populaire car elle a un coût en calcul qui est largement inférieur.

En résumé, dans cette section, une méthode de sélection *forward* par filtre a été présentée. D'autres méthodes par filtre sont possibles, mais cette méthode est retenue dans cette thèse car elle prend en compte les dépendances entre la variable candidate à la sélection et les autres variables tout en restant rapide (voir Peng et al. (2005)).

#### ***Forward* et *backward* avec une approche *wrapper***

On rappelle que dans le cas des approches *wrapper*, le coût algorithmique est a priori plus important car le critère de sélection utilisé dépend du modèle. Par conséquent, pour chaque sous-ensemble de variables candidat, le modèle intervient dans le calcul du critère. Des choix heuristiques sont alors nécessaires.

Un choix heuristique classique est le processus de sélection *forward* déjà utilisé précédemment. Plus précisément, c'est un processus de sélection séquentielle qui consiste, en partant d'un modèle de classification à 0 variable, à ajouter à chaque étape l'indicateur qui maximise un critère de sélection qui, dans l'approche *wrapper*, dépend du modèle. Dans notre cas, le critère de sélection d'une nouvelle variable est directement lié aux performances de classification obtenues par le modèle avec cette variable.

Par exemple, avec un classifieur bayésien naïf (voir 4.3.3), l'algorithme de sélection de type *wrapper* avec une stratégie de recherche *foward* est donné ci-dessous :

1. On démarre avec 0 variable ;
2. A chaque étape, on ajoute la variable qui maximise le taux de bonne classification.

On note $\nu(m)$ le numéro du $m$-ième indicateur choisi, $y(i)$ est la « vraie » classe de l'individu $i$ et $N$ le nombre d'individus.

On choisit à l'étape 1 la variable $x_{\nu(1)}$ tel que :

$$\nu(1) = \arg\max_j \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}_{k_1=y(i)}$$

où $k_1 = \arg\max_k \mathbb{P}(y_i = k|x_i^j)$

et à l'étape m, l'indicateur $x_{\nu(m)}$ tel que :

$$\nu(m) = \underset{i\notin\{\nu(1),\ldots,\nu(m-1)\}}{\arg\max} \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}_{k_m=y(n)}$$

où $k_m = \arg\max_k \mathbb{P}(y = k|x_i^{\nu(1)},\ldots,x_i^{\nu(m-1)},x_i^j)$.

Cette méthode peut être particulièrement intéressante dans le cas du classifieur bayésien naïf car une mise à jour du critère de performance à chaque ajout d'une variable est peu coûteux en calcul.

La sélection *backward* est le processus inverse. On part du modèle de classification contenant tous les indicateurs. A chaque étape, on teste la suppression de chacun des

indicateurs restants et on supprime la variable ayant le moins d'influence sur les performances de classification. Une amélioration des performances peut être obtenue en supprimant une variable.

#### Autres stratégies de recherche

D'autres méthodes de sélection séquentielles sont possibles :

- PTA($l$,$r$) (*Plus $l$ Take Away $r$*) : pour cette méthode, on applique $l$ fois *forward*, et $r$ fois *backward*, puis on répète le procédé.
- GPTA($l$,$r$) (*Generalized Plus $l$ Take Away $r$*) : ce qui différencie cette méthode de la précédente est que $l$ variables sont ajoutées en même temps au sous-ensemble de variables courant. Plus précisément, plutôt que d'ajouter une par une les variables en évaluant à chaque fois les performances, on les calcule en ajoutant $l$ variables en même temps. Le $l$-uplet de variables proposant les meilleurs performances sont gardés. Puis on procède de la même façon en retirant $r$ variables.
- SFFS (*Sequential Floating Forward Selection*) : cette méthode se différencie de la méthode PTA par le fait que $l$ et $r$ ne sont pas fixés : on parle de version flottante. Ainsi, les variables sont ajoutées une par une jusqu'à ce que les performances ne soient plus améliorées. Ensuite, on fait la même chose mais avec un processus *backward*.

Une étude de Kudo et Sklansky (2000) montre que SFFS et son équivalent en *backward* SFBS (*Sequential Floating Backward Selection*) donne les meilleurs résultats. Cependant, cette amélioration s'accompagne d'une augmentation des temps de calcul. Dans le chapitre 5, la méthode de sélection SFFS sera étudiée.

Une sélection par algorithmes génétiques est également possible, et plusieurs études ont été menées comme par exemple dans Yang et Honavar (1998) et Siedlecki et Sklansky (1989). Cependant, celle-ci s'avère beaucoup trop gourmande en temps de calcul : il faut évaluer le modèle de chaque individu de la population où un individu est un vecteur binaire traduisant la présence ou non d'un indicateur dans la sélection.

Dans le cas où le jeu de données est très grand, l'algorithme génétique semble adapté comme le font remarquer Kudo et Sklansky dans Kudo et Sklansky (2000). D'ailleurs la convergence a été théoriquement étudiée dans la thèse de Raphaël Cerf dans Cerf (1994), mais celle-ci reste très lente et l'amélioration des résultats est trop faible par rapport à l'explosion des temps de calcul.

### 4.4.4 Conclusion sur la sélection de variables

La méthodologie proposée dans cette thèse consiste à engendrer un très grand nombre d'indicateurs ce qui a un impact important sur les temps de calcul. De plus, beaucoup de ces indicateurs sont redondants. La méthode mRMR, présentée dans la section 4.4.3, sélectionne les variables de façon à minimiser la redondance avec une vitesse d'optimisation plus adéquate que dans le cas de l'utilisation d'une méthode de

sélection gloutonne. Cette méthode propose des performances intéressantes (voir Peng et al. (2005)). Ces raisons justifient l'utilisation de mRMR comme principale méthode de sélection des indicateurs dans cette thèse. Dans le cadre du classifieur bayésien naïf, les taux de bonne classification obtenus avec cette méthode seront comparés aux taux obtenus avec des méthodes de sélection de type *wrapper*.

Le point commun entre toutes les approches présentées réside dans le besoin de label. Des méthodes de sélection de variables sans label existent et permettraient de ne pas favoriser le sur-apprentissage. Ces méthodes, dont des exemples sont donnés dans la section 4.4.1, classent les variables en fonction d'un critère indépendant des labels. Elles seraient particulièrement adaptées dans le cas où les labels sont trop incertains ou non disponibles.

Cependant, on rappelle que l'on a besoin d'évaluer les performances de la méthodologie que nous proposons ici, et que la seule façon d'y parvenir est bien de disposer des labels. Quelques méthodes de sélection de variables non supervisées sont également testées, mais celles-ci montrent rapidement leurs limites en termes de performances comme l'illustrent les résultats dans le chapitre 5.

## 4.5 Conclusion du chapitre

On peut résumer la méthodologie proposée par la figure 4.6.

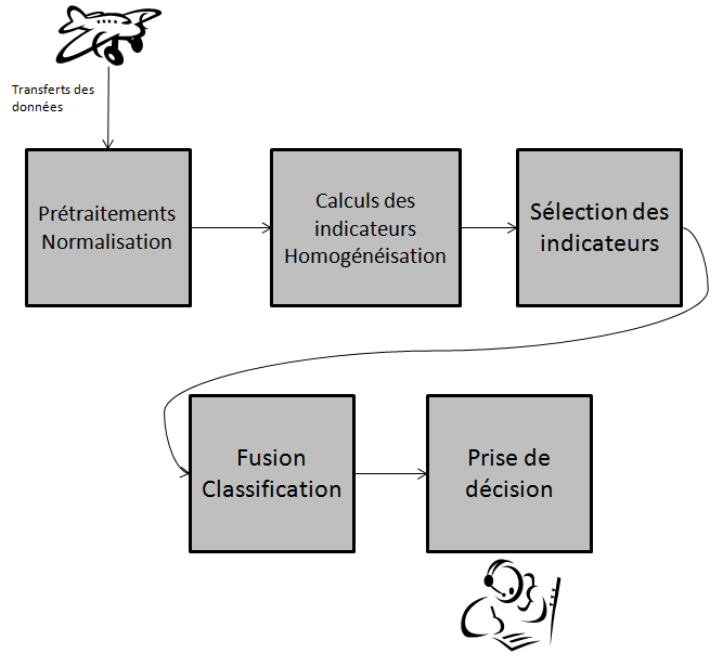


FIGURE *4.6: Architecture de la méthodologie.*

L'approche proposée a de nombreux avantages au-delà du fait qu'elle permet d'uti-

liser des méthodes statistiques performantes de détection de ruptures et de classification. Elle est construite directement à partir des variables choisies par les experts, ce qui facilite l'interprétation des résultats et donc évite l'utilisation de « boite noire ». Cette interprétation est rendue plus facile par le processus de sélection qui s'avère être une aide à la calibration des algorithmes, et par le processus d'homogénéisation en indicateurs binaires qui permet de contourner la complexité des signaux étudiés.

Enfin, cette méthodologie est très générique et peut être utilisée dans de nombreux domaines. D'ailleurs ce processus de binarisation n'est pas nouveau et peut s'apparenter à des méthodes utilisées dans Fleuret (2004) et Hegedus et al. (2011).

CHAPITRE 5

# Mise en œuvre et expérimentations

Dans ce chapitre, la méthodologie est mise en œuvre et évaluée sur des données artificielles. En effet, les données réelles disponibles à Snecma (comme on l'a vu dans les chapitres précédents) sont dispersées dans différentes bases et sous différentes formes : elles peuvent se trouver du côté des ateliers, des bancs d'essai mais aussi au niveau du *Customer Service Center*. Dans tous les cas, les données sont analysées par les experts des différents services qui leur attribuent des labels, mais le travail de centralisation des données correctement labellisées est long et coûteux à mettre en place. Ces difficultés nous conduisent donc à utiliser des données simulées imitant les signaux observés en cas d'anomalie.

Si l'étude est concluante, il sera donc possible de justifier la mise en place d'une labellisation des données réelles de vol et leur centralisation.

## 5.1 Construction des données artificielles

On rappelle que le processus de normalisation permet d'obtenir des données (résidus) que l'on peut supposer stationnaires en l'absence d'anomalie. Sur ces données normalisées, les changements d'évolution en cas d'anomalie sont assez bien connus et peuvent être observés sur les données réelles.

Considérons par exemple, des données semblables à celles présentées brièvement dans la section 2.4.2 qui, rappelons le, sont des données issues des messages transmis par l'avion au sol, contenant un ensemble résumé des paramètres de vol. Dans les figures 5.1, 5.2 et 5.3, on retrouve différents changements d'évolution sur des données réelles :

- un changement de variance dans la figure 5.1 ;
- un saut dans la moyenne dans la figure 5.2 ;
- un changement de tendance dans la figure 5.3.

Ces changements correspondent à des situations concrètes. Par exemple, un changement de pente peut être dû à une dégradation progressive d'un composant, alors qu'un changement de moyenne peut être dû au remplacement d'une pièce du moteur ou à un *water-wash*.

Pour la simulation des données, l'idée est de se rapprocher de ces types de changement observés.

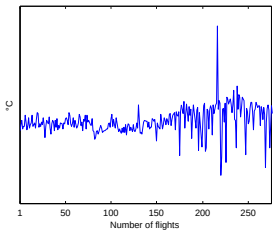


FIGURE *5.1: Changement de variance*

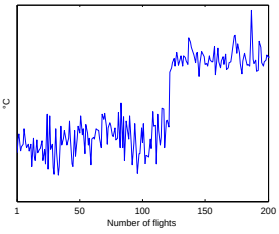


FIGURE *5.2: Changement de moyenne*

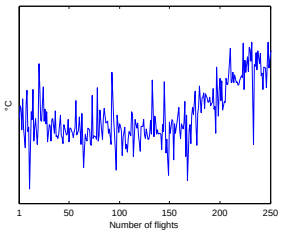


FIGURE *5.3: Changement de tendance*

### 5.1.1 Présentation des données

Dans cette thèse, on se place dans le cas qui est décrit dans la section 2.4.4, où des données d'un moteur $j$ sont normalisées puis présentées à l'opérateur métier. A partir de ces données, ce dernier doit confirmer si une anomalie est présente ou non sur le moteur, et définir, dans le cas où il y a une anomalie, le type de changement permettant de retrouver l'origine de l'anomalie.

Chaque donnée simulée est une série temporelle unidimensionnelle de longueur variable (simulant l'historique « disponible » du moteur). Le jeu de données utilisé comporte $N$ séries temporelles.

Fixons les notations. Pour un moteur $j$ ($j = 1, \ldots, N$), la série est de longueur $n(j)$ et est notée $Z_1^j, Z_2^j, \ldots, Z_{n(j)}^j = (Z^j)$.

Lorsque le moteur $j$ ne présente pas d'anomalie, on peut supposer que la série $(Z_1^j, Z_2^j, \ldots, Z_{n(j)}^j)$, est un bruit gaussien standard, c'est-à-dire que les observations $Z_i^j$ sont indépendantes et identiquement distribuées suivant la loi $\mathcal{N}(\mu = 0, \sigma^2 = 1)$ avec une longueur comprise entre 100 et 200.

Pour simuler une donnée $(Z_i^j)$ correspondant à un moteur avec anomalie, on se limite à trois types de changement tels qu'ils sont présentés dans les figures 5.1, 5.2 et 5.3 : changement de pente, changement de variance, changement de moyenne.

Les figures 5.4, 5.5 et 5.6 sont des exemples de séries où des anomalies ont été simulées.

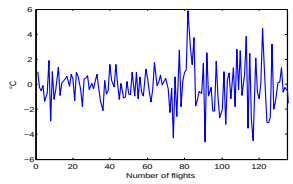


FIGURE *5.4: Changement de variance.*

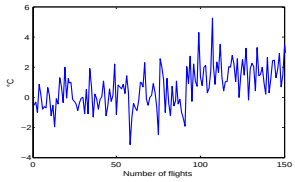


FIGURE *5.5: Changement de moyenne.*

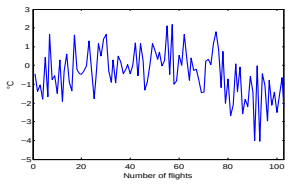


FIGURE *5.6: Changement de tendance.*

Les paramètres choisis pour les trois types de changement sont décrits ci-dessous :

TABLE *5.1: Répartition des types de changement du premier jeu de données $A$*

| Jeu de données $A$ | | |
|---|---|---|
| | Type de changement | Effectifs |
| Sans anomalie $C_0$ | - | 3000 |
| anomalie $C_1$ | rupture sur la variance | 1000 |
| anomalie $C_2$ | rupture sur la moyenne | 1000 |
| anomalie $C_3$ | changement de tendance | 1000 |

1. pour le changement de variance : pour chaque moteur $j$, les observations $Z_i^j$ suivent une loi $\mathcal{N}(0, \sigma^2)$ où $\sigma^2 = 1$ avant l'instant de rupture et où $\sigma$ est tiré uniformément sur $[1.01,\ 5]$ après l'instant de rupture.
2. pour le changement de moyenne : pour chaque moteur $j$, les observations $Z_i^j$ suivent une loi $\mathcal{N}(\mu, 1)$ avec $\mu = 0$ avant l'instant de rupture et où $\mu$ est tiré uniformément sur $[1.01,\ 5]$ après l'instant de rupture.
3. pour le changement de tendance : pour chaque moteur $j$ , les observations $Z_i^j$ suivent une loi $\mathcal{N}(\mu, \sigma^2 = 1)$ où $\mu = 0$ avant l'instant de rupture et où $\mu$ augmente linéairement à partir de $0$ dès l'instant de rupture avec une pente choisie uniformément dans$[0.02,\ 3]$.

La justification du choix des paramètres est donnée à la section 5.1.3.

Si on note $n(j)$ la longueur d'un signal $(Z_i^j)$, l'instant de rupture est tiré suivant une loi uniforme entre la $\frac{2}{10}n(j)$-ième observation et la $\frac{8}{10}n(j)$-ième observation.

Pour le jeu de données, que l'on note $A$, $N = 6000$ séries temporelles sont engendrées suivant les descriptions données, en répartissant les types de signaux selon le tableau 5.1. Comme il s'agit d'un cas monodimensionnel, on suppose qu'il existe une bijection entre le type d'anomalie et le type de changement, c'est-à-dire que la classe de l'anomalie est identifiée dès que le type de changement est défini. Ainsi, on définit les classes $C_0$, $C_1$, $C_2$ et $C_3$ telles que :

- $C_0$ regroupe l'ensemble des moteurs n'ayant pas d'anomalie,
- $C_1$ regroupe l'ensemble des moteurs dont l'anomalie se traduit par un changement dans la variance,
- $C_2$ regroupe l'ensemble des moteurs dont l'anomalie se traduit par un changement dans la moyenne,
- $C_3$ regroupe l'ensemble des moteurs dont l'anomalie se traduit par un changement dans la tendance.

### 5.1.2 Remarques pour un cas de données multivariées

Lors de l'analyse, un opérateur métier ne se limite pas à l'étude de l'évolution d'un seul paramètre, mais va en considérer plusieurs. La méthodologie, sans modification,

peut traiter les cas multidimensionnels.

Dans ce cas, les séries ont plusieurs composantes et chaque composante correspond à un paramètre étudié par l'expert métier. On peut enrichir les notations du cas unidimensionnel en posant :

$$(Z_i^{(l,j)}) = (Z_1^{(l,j)}, \ldots, Z_{n(j)}^{(l,j)})$$

où $Z_i^{(l,j)}$ est là $l$-ième composante de la $i$-ième observation du moteur $j$.

On peut remarquer que par souci de simplification des notations, celles-ci fixent le nombre d'observations sur chaque dimension égale à $n(j)$. Cependant ,la méthodologie proposée permet, sans modification, de traiter des cas où chaque dimension a un nombre d'observations $n(l, j)$ qui lui est propre.

Chacune des dimensions des séries temporelles a sa propre évolution en cas d'anomalie, comme le rappelle le tableau 3.1 (p.23) présentant un ensemble d'exemples de signatures de pannes. Dans ce cas, chaque signature peut être associée à une classe d'anomalie $C_k$ et on note $C_0$ la classe regroupant les moteurs sans anomalie.

Ainsi, pour traiter le cas multidimensionnel, il suffit de traiter chaque dimension comme on traite le cas unidimensionnel, et de concaténer les indicateurs calculés en un seul vecteur.

Par exemple, supposons que l'on soit en dimension 3 : $l \in \{1,2,3\}$. Alors pour chaque dimension, on calcule les vecteurs d'indicateurs indépendamment les uns des autres. Supposons que pour chaque dimension on ait $P' = 100$ indicateurs ; on se retrouve pour chaque dimension avec un vecteur de taille 100. Il suffit alors de concaténer les 3 vecteurs pour obtenir un vecteur de taille $P = 300$ et on se retrouve alors dans le cas unidimensionnel.

### 5.1.3 Remarques sur le choix des paramètres de simulation des données

Les données simulées l'ont été pour que la détection soit plus difficile que sur les données réelles. Les sauts de moyenne, de variance et de tendance sur les données simulées ont des valeurs nettement plus petites que sur les données réelles où on a constaté des anomalies.

Dans cette section, on montre comment certains paramètres ont été choisis.

La limite inférieure d'un saut significatif sur la moyenne est déterminée à partir d'un test de Student.

Sous $H_0$ : il n'y a pas de rupture au milieu de la fenêtre (voir la section 4.2.4),

$$T = \frac{\hat{\mu_1} - \hat{\mu_2}}{\hat{\sigma}} * \sqrt{n/2} \sim_{H_0} \mathscr{T}(2n-2)$$

où

- $n$ est la demi-taille de la fenêtre (la taille de chaque population).
- $\hat{\mu_1}$ et $\hat{\mu_2}$ sont les moyennes empiriques sur chaque demi-fenêtre.
- $\hat{\sigma} = \sqrt{\frac{1}{2}(\hat{\sigma_1}^2 + \hat{\sigma_2}^2)}$ est l'estimateur de l'écart-type sur la fenêtre entière et $\hat{\sigma_i}^2$ est l'estimateur de la variance de chaque demi-fenêtre.

Par exemple, pour que le saut soit détectable dans 95% des cas :
- si $n = 15$ et $\sigma = 1$, il faut que $\hat{\mu_1} - \hat{\mu_2} > 2.13\ \hat{\sigma} * \sqrt{2/15} \sim 0.78$.
- si $n = 50$ et $\sigma = 1$, il faut que $\hat{\mu_1} - \hat{\mu_2} > 2\hat{\sigma} * \sqrt{2/50} \sim 0.4$.

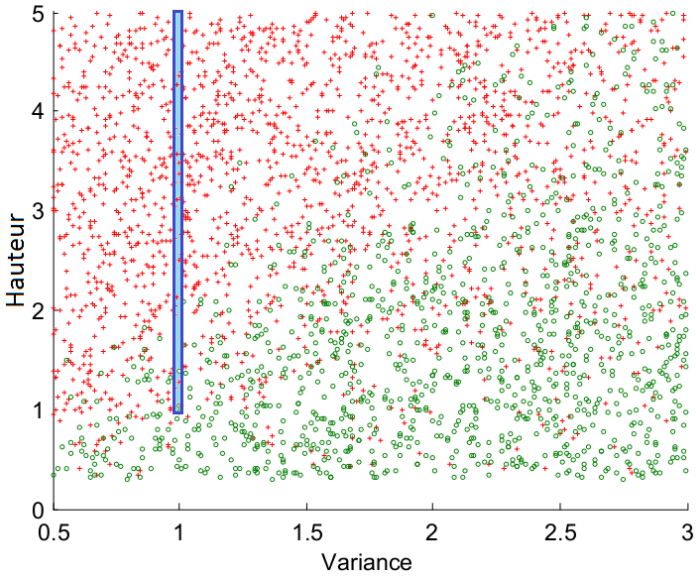


FIGURE *5.7: Influence de la hauteur du saut et de la variance sur le test de Student sur une fenêtre de taille 30. Les points rouges et verts correspondent respectivement à des test de Student significatifs et non significatifs. La barre bleue donne l'emplacement des sauts utilisés pour les données simulées.*

Dans la figure 5.7, on montre l'influence de la variance du signal sans anomalie et de la hauteur du saut sur le résultat d'un test de Student. Les niveaux de saut ont été choisis aléatoirement dans $[0.3, 5]$ et les variances dans $[0.5, 3]$. On voit bien que la détection sera d'autant plus difficile que l'on est proche de la diagonale de la figure 5.7.

## 5.2 Présentation des indicateurs

La plupart des indicateurs utilisés dans cette étude sont construits à partir du processus d'homogénéisation décrit dans 4.2.4. Les tests statistiques sont choisis suffisamment simples pour faciliter l'interprétation pour les experts métiers. Une étape de confirmation peut éventuellement être ajoutée.

On commence par donner une description brève des tests utilisés puis on explicite les paramètres utilisés pour le calcul des indicateurs.

### 5.2.1 Définition des tests

#### 5.2.1.1 Test de rang de Mann–Whitney–Wilcoxon

Il s'agit d'un test d'homogénéité entre deux populations (voir Wilcoxon (1945) et Van der Vaart (2000)). Il est étudié et généralisé dans Lung-Yut-Fong (2011).

Il a l'avantage d'être non paramétrique : on n'a aucun a priori sur les lois de $Z_1, \ldots, Z_n$. La seule hypothèse faite est l'indépendance des observations.

Dans ce test, toutes les observations sont classées par ordre croissant et on note $R_i$ le rang de l'observation $Z_i$.

On sépare l'échantillon en deux pour avoir $(Z_1, \ldots, Z_s)$ et $(Z_{s+1}, \ldots, Z_n)$. On note $W$ la statistique de Wilcoxon, qui correspond à la somme des rangs des observations appartenant à la première demi-fenêtre :

$$W_s = \sum_{i=1}^{n_1} R_i.$$

Pour ce test, on a deux hypothèses $H_0$ et $H_1$ données par :

– $H_0$ : les deux populations $(Z_1, \ldots, Z_s)$ et $(Z_{s+1}, \ldots, Z_n)$ ont les mêmes lois,

– $H_1$ : les deux populations $(Z_1, \ldots, Z_s)$ et $(Z_{s+1}, \ldots, Z_n)$ ont des lois différentes.

À partir d'une population de 10 individus, on peut montrer que sous l'hypothèse $H_0$ la statistique de test peut être approximée par une normale :

$$\frac{W_s - \mathbb{E}[W_s]}{\sqrt{Var(W_s)}} \sim_{H_0} \mathcal{N}(0,1).$$

où $\mathbb{E}[W_s] = \frac{1}{2}s(n+1)$ et $Var(W_s) = \frac{1}{12}s(n-s)(n+1)$.

Sous $H_1$, cette statistique se décale vers des grandes valeurs positives ou négatives.

#### 5.2.1.2 Test de Fisher pour l'égalité de deux variances

Il s'agit de tester l'égalité des variances de deux populations $Y_1 = (Z_1, \ldots, Z_s)$ et $Y_2 = (Z_{s+1}, \ldots, Z_n)$ sous l'hypothèse que ces populations sont issues d'une loi normale,

Pour ce test, on calcule la statistique $F$ donnée par :

$$F = \frac{\hat{\sigma_{Y_1}}^2}{\hat{\sigma_{Y_2}}^2}.$$

où $\hat{\sigma_{Y_1}}^2$ et $\hat{\sigma_{Y_2}}^2$ sont les estimateurs des variances des populations $Y_1$ et $Y_2$.

Sous l'hypothèse $H_0$ : $\sigma^2_{Y_1} = \sigma^2_{Y_2}$, $F$ suit une loi de Fisher :

$$\frac{\hat{\sigma_{Y_1}}^2}{\hat{\sigma_{Y_2}}^2} \sim_{H_0} F(s-1, n-s-1).$$

Sous $H_1$, la loi de F est celle d'un Fisher à un coefficient multiplicatif près ($> 1$ si $\sigma^2_{Y_1} > \sigma^2_{Y_2}$ ou $< 1$ si $\sigma^2_{Y_1} < \sigma^2_{Y_2}$).

#### 5.2.1.3 Test de Kolmogorov-Smirnov

Dans notre cas, il s'agit de tester l'égalité en loi de deux populations $Y_1$ et $Y_2$ en considérant la statistique $D$ calculée par :

$$D = \sup_x |F_1(x) - F_2(x)|.$$

où $F_1$ (resp. $F_2$) est la fonction de répartition empirique de la population de données $Y_1$ (resp. $Y_2$).

Sous $H_0$, $D$ reste dans une fourchette calculée à partir de la table de Kolmogorov-Smirnov. Sous $H_1$, cette statistique se décale vers des grandes valeurs.

#### 5.2.1.4 Test de changement de pente

On estime la pente $b_1$ de la première demi-population $Y_1 = (Z_1, \ldots, Z_s)$ par :

$$\hat{b_1} = \frac{\frac{1}{s-1}\sum_{i=1}^{s}(i - \frac{s+1}{2})(Z_i - \bar{Y_1})}{\frac{1}{s-1}\sum_{i=1}^{s}(i - \frac{s+1}{2})^2},$$

où $\bar{Y_1} = \sum_{i=1}^{s} \frac{Z_i}{s}$.

En effet, si on note :

$$Z_i = a_1 + b_1 \times i + \varepsilon_i,$$

où $\varepsilon_i$ est l'erreur supposée centrée et homoscédastique.
Alors on estime $a_1$ et $b_1$ en résolvant :

$$(\hat{a_1}, \hat{b_1}) = argmin_{a_1,b_1} \sum_{i=1}^{s} (Z_i - (a_1 + b_1 \times i))^2.$$

Puis, si on note $b_2$ la pente de la deuxième demi-population $Y_2 = (Z_{s+1}, \ldots, Z_n)$, et $\hat{b_2}$ son estimateur, sous l'hypothèse $H_0$ : $b_2 = b_1$, la statistique $t$ donnée par

$$t = \frac{\hat{b_2} - \hat{b_1}}{SE_{\hat{\beta}}},$$

où $SE_{\hat{\beta}} = \sqrt{\frac{\frac{1}{s-2}\sum_{i=s+1}^{n}(Z_i - \hat{Z}_i)^2}{\sum_{i=s+1}^{n}(i - \frac{n+1}{2} - \frac{s+1}{2})^2}}$,
$t$ suit, sous $H_0$ une loi de Student à $s-2$ degrés de liberté :

$$t \sim_{H_0} \mathscr{T}(s-2).$$

Sous $H_1$, cette statistique se décale vers des grandes valeurs positives ou négatives.

#### 5.2.1.5 Test d'existence de pente non nulle

On veut tester l'existence d'une pente non nulle sur toute la fenêtre, c'est-à-dire qu'on ne divise pas la fenêtre en deux populations.

On procède comme précédemment mais avec une seule population $Y = (Z_1, \ldots, Z_n)$ et en comparant la pente $b$ de $Y$ à $0$.

La pente $b$ est estimée par :

$$\hat{b} = \frac{1/(n-1)\sum(i - \frac{n+1}{2})(Z_i - \bar{Y})}{1/(n-1)\sum(i - \frac{n+1}{2})^2},$$

où $\bar{Y_1} = \sum_{i=1}^{n} \frac{Z_i}{n}$.

Sous l'hypothèse nulle : $b = 0$, la statistique $t$ donnée par :

$$t = \frac{\hat{b}}{SE_{\hat{\beta}}},$$

où $SE_{\hat{\beta}} = \frac{\sqrt{\frac{1}{n-2}\sum_{i=1}^{n}(Z_i - \hat{Z}_i)^2}}{\sqrt{\sum_{i=1}^{n}(i - \frac{n+1}{2})^2}}$

$t$ suit, sous $H_0$ une loi de Student à $n-2$ degrés de liberté :

$$t \sim_{H_0} \mathscr{T}(n-2).$$

Sous $H_1$, cette statistique se décale vers des grandes valeurs positives ou négatives.

### 5.2.2 Indicateurs considérés

Un grand nombre de paramètres et méta-paramètres sont à choisir pour calculer les indicateurs.

I - Le test statistique pertinent
- Test de Mann–Whitney–Wilcoxon pour l'égalité des lois ;
- Test de Kolmogorov Smirnov pour l'égalité des lois ;
- Test de Fisher pour l'égalité de la variance ;
- Test de Student pour la moyenne en supposant les variances égales ;
- Test de Student pour la moyenne en supposant les variances différentes ;
- Test d'existence de pente non nulle ;
- Test de changement de pente.

Pour chacun de ces tests, on doit choisir le niveau $\alpha$.

II - Le type de confirmation
- Confirmation qu'un test est positif, sur l'intervalle de temps considéré, s'il y a un nombre minimum de fenêtres pour lesquelles on obtient un résultat positif (type 1) ;
- Confirmation qu'un test est positif s'il existe un nombre minimum de fenêtres consécutives pour lesquelles on obtient un résultat positif (type 2) ;

– Confirmation qu'un test est positif de type $k$ parmi $n$ (type 3) (voir section 4.1.1.1).

III - Les paramètres concernant la fenêtre examinée

– Sa taille $w$ : $30, 50, 100$.
– Le pas (saut entre chaque fenêtre consécutive) : $1, 5$.
– L'échelle de lissage (taille de la fenêtre de lissage par moyenne mobile) : $1, 2, 5$.

Un grand nombre d'indicateurs sont engendrés à partir des différentes valeurs possibles de paramètres et méta-paramètres donnés en I, II et III. Ainsi, un premier indicateur pourrait être construit à partir d'un test de Mann–Whitney–Wilcoxon, avec une étape de confirmation de type 1, une fenêtre de taille $30$, un pas de $1$ et une échelle de lissage de $1$. Le dernier indicateur pourrait être construit à partir d'un test de changement de pente, une étape de confirmation de type 3, une fenêtre de taille $100$, un pas de $5$, et une échelle de lissage de $5$.

Les indicateurs valent $0$ ou $1$ selon que le test est positif (significatif) ou négatif (non significatif).

## 5.3 Résultats

Dans cette section, on présente une série de quatre expérimentations permettant d'évaluer la pertinence de la méthodologie.

La première expérimentation permet de s'assurer que les indicateurs créés par la première partie de la méthodologie sont suffisants pour répondre aux problèmes de classification (« sans anomalie » ou « avec anomalie ») en utilisant une sélection de variables de type mRMR (Rabenoro et al. (2014c)).

Dans la deuxième expérimentation, le problème de classification est désormais multi-classes et les données simulées sont un peu plus difficiles à discriminer que dans la première partie, toujours avec une sélection de variables de type mRMR (Rabenoro et al. (2014b)).

Dans la troisième expérimentation, le problème de classification est toujours multi-classes et cette fois-ci ce sont des méthodes *wrapper* qui sont utilisées pour la sélection de variables, l'objectif étant de vérifier s'il y a une amélioration sensible par rapport aux méthodes par filtre utilisées jusqu'alors (Rabenoro et al. (2015)).

Enfin, dans la dernière expérimentation, le bruit des données simulées est complexifié ; il s'agit dans cette expérience de vérifier que l'utilisation d'une phase de confirmation est pertinente (Rabenoro et al. (2014a)).

### 5.3.1 Première expérimentation

#### 5.3.1.1 Performances avec tous les indicateurs

Les premiers résultats sont obtenus sur deux jeux de données $A$ et $B$ structurés comme indiqué dans le tableau 5.1. Ils sont engendrés dans les mêmes proportions suivantes :

- 3000 données sans anomalie ;
- 1000 avec un changement de moyenne ;
- 1000 avec un changement de variance ;
- 1000 avec un changement de pente.

Les changements de moyenne, de variance et de pente sont choisis selon les spécifications de la section 5.1.1. Cependant, on se limite à un problème de classification à deux classes : « sans anomalie » ou « avec anomalie ». Le jeu de données $B$ est plus complexe que le jeu $A$ :

- le bruit modélisé est un $\chi^2$ à 4 degrés de liberté (ce choix d'un $\chi^2$ vient du fait qu'un score peut-être déterminé à partir de la statistique de test du rapport de vraisemblance) ;
- une variation déterministe a été ajoutée sur des signaux supposés sans anomalie choisis aléatoirement, ce qui permet de simuler une normalisation sous-optimale sur un ensemble aléatoire de signaux ;
- la longueur des signaux est plus courte : elle est tirée uniformément sur $[100, 150]$.

De plus, parmi les 3000 données sans anomalie, 1200 ont une variation lente. Cette variation est simulée par un sinus d'amplitude 1 et une période égale à $\frac{2}{3}$ de la longueur du signal. La figure 5.8 donne un exemple des données de l'échantillon $B$ présentant une variation lente.

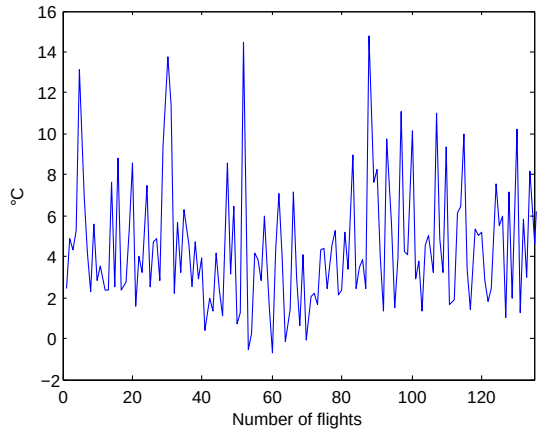


FIGURE *5.8: Exemple d'observations sans anomalie du jeu de données $B$ présentant une variation lente.*

Pour cette expérimentation, l'échantillon d'apprentissage est composé de 1000 signaux en gardant les proportions de répartition entre les classes. L'évaluation est faite sur les 5000 signaux restants qui forment l'ensemble test, et qui ont été divisés en 10 groupes de 500 signaux, dans le but de mesurer une variabilité des performances.

Pour les indicateurs, ils ont été construits à partir de 3 tests :

1. le test Mann–Whitney–Wilcoxon sur les moyennes ;
2. le test de Kolmogorov-Smirnov sur les distributions ;
3. le test de Fisher sur les variances.

Différents paramètres et méta-paramètres ont été utilisés :
- les tailles de fenêtres sont choisies parmi (30, 50 et $\min(n-2, 100)$) où $n$ est la longueur du signal ;
- les niveaux du test choisis sont (0.005, 0.1 et 0.5) ;
- l'échelle de lissage est telle que la version lissée du signal correspond à une moyenne mobile de 5 observations.

Les indicateurs de confirmation sont choisis parmi l'une des méthodes suivantes :

1. pour chaque test, l'indicateur binaire prend la valeur $1$ si le test détecte une rupture sur $\beta \times m$ fenêtres parmi $m$ fenêtres. Les paramètres sont donc le test à utiliser, les valeurs de $\beta$ (ici on considère 0.1, 0.3 et 0.5) et le nombre d'observations en commun entre deux fenêtres consécutives (ici on considère la longueur de la fenêtre $w$ moins 1, 5 ou 10 ou autrement dit le pas entre deux fenêtres consécutives est de 1, 5 ou 10.)
2. pour chaque test, l'indicateur binaire prend la valeur $1$ si ce test détecte une rupture sur $\beta \times m$ fenêtres consécutives parmi $m$ fenêtres.
3. pour chaque test, l'indicateur binaire prend la valeur $1$ s'il existe 5 fenêtres consécutives telles qu'un changement est détecté sur au moins $k$ ($k = 3, 4$) de ces 5 fenêtres.

De plus, ces indicateurs sont calculés à partir du signal d'origine mais aussi sur une version lissée du signal (en utilisant une moyenne mobile de 5 observations).

La description des indicateurs est résumée dans le tableau 5.2.

Plus de 50 configurations différentes sont utilisées pour chaque type confirmation amenant à un nombre de 810 indicateurs.

Dans le tableau 5.3 sont résumées les performances obtenues avec les Random Forests sur les deux jeux de données $A$ et $B$, en gardant l'ensemble des 810 indicateurs (sans sélection).

Les performances de l'*out-of-bag* (OOB) fournies par les Random Forests sont également données (voir 4.3.2.2).

Les performances obtenues sur le jeu de données $A$ sont très bonnes, tandis que sur le jeu de données $B$ elles sont acceptables. La similarité des résultats obtenus sur le jeu de validation et sur l'échantillon OOB permet de montrer que celui-ci est un bon échantillon pour estimer les performances réelles. Sur les données $B$ le sur-apprentissage

TABLE *5.2: Liste des valeurs utilisées pour les paramètres utilisés pour calculer les indicateurs. Pour plus d'informations sur la lecture du tableau, voir la section 5.2.2.*

| Paramètres | Valeurs |
|---|---|
| test statistique | Test Mann-Whitney-Wilcoxon<br>Test de Kolmogorov-Smirnov test<br>Test de Fisher |
| taille de la fenêtre ($\tau$) | 30<br>50<br>100 |
| niveau du test | 0.005<br>0.1<br>0.5 |
| pas ($\delta$) | 1<br>5<br>10 |
| confirmation type 1 et 2, ratio global et ratio consécutif ($\beta$) | 0.1<br>0.3<br>0.5 |
| confirmation type 3, $k$ parmi $l$ $(l,k)$ | (3,2)<br>(5,3)<br>(5,4) |
| moyenne mobile | 1<br>5 |

TABLE *5.3: Taux de bonne classification obtenus dans le cas binaire avec les Random Forests en utilisant 810 indicateurs binaires. Pour l'ensemble test, c'est la moyenne du taux de bonne classification qui est donnée avec son écart-type entre parenthèse.*

| Jeu | Taux de bonne classification sur l'ensemble d'apprentissage | Taux de bonne classification sur l'OoB | Taux de bonne classification sur l'ensemble test |
|---|---|---|---|
| $A$ | 1.00 | 0.953 | 0.957 (0.0089) |
| $B$ | 1.00 | 0.828 | 0.801 (0.032) |

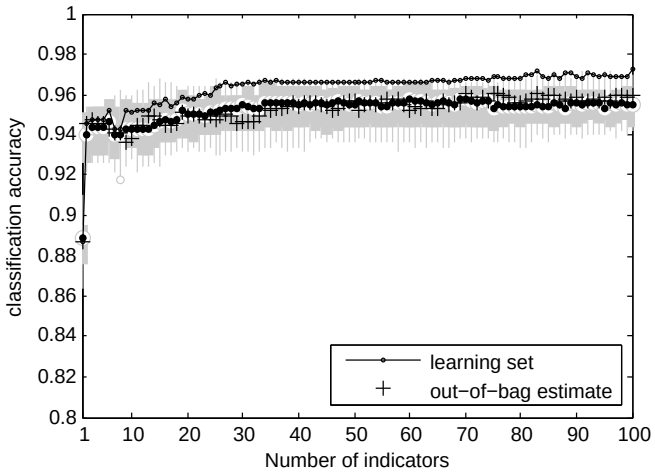


FIGURE *5.9:* ***Jeu de données*** $A$ *: Taux de bonne classification sur l'ensemble d'apprentissage (cercle noir) en fonction du nombre d'indicateurs. Une boxplot donne le taux de bonne classification pour l'ensemble test avec la médiane (point noir dans un cercle blanc). L'estimation des taux de bonne classification obtenus par l'out-of-bag (OOB) est donnée par les croix.*

est plutôt marqué (différences de 20% entre les taux de bonnes reconnaissances entre l'ensemble d'apprentissage et l'ensemble test).

#### 5.3.1.2 Sélection des indicateurs

Bien que les résultats soient satisfaisants, présenter 810 indicateurs à l'opérateur métier n'est pas envisageable. Il est nécessaire de réduire le nombre d'indicateurs quitte à réduire les performances en classification. De plus, étant donné le sur-apprentissage constaté notamment sur l'échantillon $B$, une réduction du nombre d'indicateurs semble appropriée avant de procéder à la classification.

En utilisant mRMR (voir 4.4.3), les 810 indicateurs ont été classés en utilisant l'information mutuelle. Une approche *forward* a été mise en œuvre pour évaluer le nombre d'indicateurs nécessaires pour obtenir des performances de prédictions acceptables. On rappelle que dans cette approche, les indicateurs sont ajoutés un par un en utilisant l'ordre fourni par mRMR et ne sont jamais retirés. Étant donné que mRMR prend en compte la redondance, on suppose que cela ne devrait pas être un inconvénient majeur. Pour chaque nouvel indicateur ajouté à l'ensemble des indicateurs déjà sélectionnés, un Random Forest est appris sur l'échantillon d'apprentissage, puis il est évalué sur l'ensemble test.

La figure 5.9 montre les résultats sur le jeu de données $A$. Les performances sont plutôt correctes même avec un petit nombre d'indicateurs. Ces performances s'améliorent constamment sur l'ensemble d'apprentissage et se stabilisent sur l'ensemble des

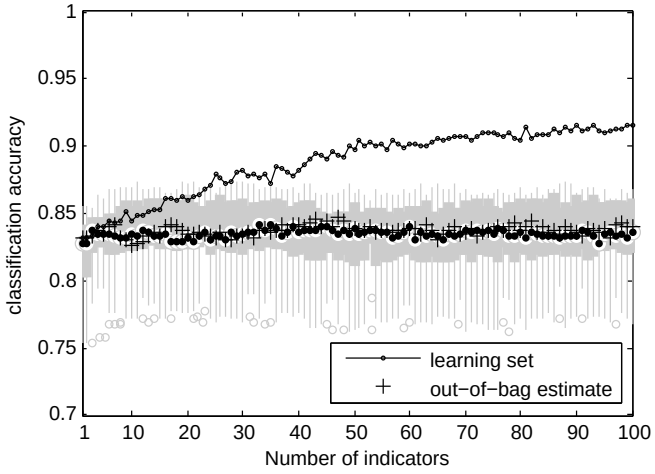


FIGURE *5.10:* ***Jeu de données*** $B$ *: Taux de bonne classification sur l'ensemble d'apprentissage (cercle noir) en fonction du nombre d'indicateurs. Une boxplot donne le taux de bonne classification pour l'ensemble test avec la médiane (point noir dans un cercle blanc). L'estimation des taux de bonne classification obtenus par l'out-of-bag (OOB) est donnée par les croix.*

données tests (et OOB) vers 40 indicateurs.

La figure 5.10 montre des résultats sur le jeu de données $B$. Les performances sont moindres que pour le jeu de données $A$, et le sur-apprentissage est un peu plus marqué, mais les comportements généraux restent les mêmes. Cette diminution des performances était attendue car le jeu de données $B$ a été conçu pour rendre l'analyse plus difficile. En effet, l'indicateur obtenu par le test de Fisher sur les variances est correct sous l'hypothèse de normalité (jeu $A$), alors que le jeu de données $B$ est bruité par une loi de $\chi^2$.

Cependant, dans les deux cas, le processus de sélection des indicateurs permet de montrer qu'un petit sous-ensemble d'indicateurs permet d'obtenir de bonnes performances. Ainsi, pour l'opérateur, il sera possible de traiter ce petit nombre d'indicateurs pour prendre une décision, et de le comparer à la décision proposée par les Random Forests.

#### 5.3.1.3 Indicateurs sélectionnés

Pour illustrer l'intérêt de la méthode, le tableau 5.3.1.3 présente les dix premiers indicateurs sélectionnés par mRMR dans le jeu de données A. Avec ces dix indicateurs, une bonne performance est atteinte avec un taux de bonne classification de 0.944 (sur OOB l'estimation est de 0.938).

Le tableau 5.3.1.3 montre les dix meilleurs indicateurs pour le jeu de données $B$. Encore une fois, les performances sont acceptables sur l'échantillon test avec une moyenne

TABLE *5.4: Les dix meilleurs indicateurs obtenus par mRMR pour le jeu de données $A$. Confu(k,n) est un indicateur de confirmation qui vaut 1 si le test de Mann–Whitney–Wilcoxon est positif dès qu'il existe pour n fenêtres consécutives, k fenêtre où le test est positif. Conff(k,n) est un indicateur de confirmation identique mais pour le F-test. Ratef($\alpha$) correspond à la positivité du F-test sur $\alpha \times m$ fenêtres quand il y a $m$ fenêtres au total. Lseqf($\alpha$) correspond à la positivité du F-test sur $\alpha \times m$ fenêtres consécutives quand il y a $m$ fenêtres au total. Lsequ($\alpha$) est le même que précédemment mais avec un test Mann–Whitney–Wilcoxon. On remarque ici qu'aucun des indicateurs n'utilise une version lissée du signal.*

| Type d'indicateur | Niveau | Taille de fenêtre | Pas |
|---|---|---|---|
| F test | 0.005 | 100 | 5 |
| confu(2,3) | 0.005 | 50 | 5 |
| ratef(0.1) | 0.005 | 50 | 5 |
| KS test | 0.005 | 100 | 1 |
| conff(3,5) | 0.005 | 100 | 5 |
| KS test | 0.1 | 100 | 5 |
| F test | 0.005 | 100 | 1 |
| KS test | 0.005 | 100 | 10 |
| lseqf(0.1) | 0.1 | 50 | 1 |
| F test | 0.005 | 50 | 10 |

TABLE *5.5: Les dix meilleurs indicateurs obtenus par mRMR pour le jeu de données $B$. Les notations sont les mêmes que celles du tableau 5.3.1.3. Il y a plus d'indicateurs de confirmation sélectionnés quand le bruit est plus complexe.*

| Type d'indicateur | Niveau | Taille de fenêtre | Lissage | Pas |
|---|---|---|---|---|
| KS test | 0.005 | 100 | non | 5 |
| lseqf(0.1) | 0.1 | 30 | oui | 1 |
| confu(4,5) | 0.005 | 30 | non | 1 |
| U test | 0.1 | 100 | non | 5 |
| confu(4,5) | 0.005 | 100 | non | 5 |
| confu(2,3) | 0.005 | 100 | non | 1 |
| lsequ(0.3) | 0.1 | 50 | non | 1 |
| F test | 0.005 | 100 | oui | 10 |
| confu(2,3) | 0.005 | 30 | non | 5 |
| KS Test | 0.005 | 100 | non | 10 |

de bonne classification de 0.831 (sur OOB l'estimation est de 0.826). Comme prévu, les tests de Fisher sont moins intéressants pour le jeu de données $B$ étant donné qu'il n'est pas gaussien. De plus, il y a plus d'indicateurs de confirmation sélectionnés.

Dans les deux cas, on peut voir que la méthode de sélection d'indicateurs utilisée permet de faire une sélection efficace parmi un très grand nombre d'indicateurs. Cette sélection permet indirectement d'ajuster automatiquement les paramètres des tests utilisés et de simplifier la classification. De plus, la simplicité des indicateurs utilisés et le fait qu'ils soient binaires les rendent faciles à comprendre pour les opérateurs.

#### 5.3.1.4 Conclusion

Ainsi, à partir des savoirs expert, on a pu construire des statistiques de test et des plages de paramètres crédibles. A partir de ces statistiques, un grand nombre d'indicateurs binaires sont engendrés, de façon à couvrir l'espace de toutes les configurations possibles, et peuvent être agrégés de façon simple. Ainsi, le problème de diagnostic a été converti en un problème de classification de données comprenant un très grand nombre de variables binaires. La sélection d'indicateurs permet d'en réduire le nombre, de façon à ce que l'analyse soit possible pour l'opérateur métier. Ainsi, celui-ci peut comprendre au moins partiellement comment la décision automatique a été obtenue puisque les indicateurs sont compréhensibles.

On peut remarquer que la méthodologie présentée peut être utilisée dans différents domaines d'application dès que les décisions de l'expert peuvent être traduites en des scores, y compris lorsque les signaux sont de longueurs différentes.

De plus, les résultats obtenus sur ces données simulées sont corrects, même lorsque l'on pratique la sélection des indicateurs et que l'on en conserve un nombre limité. Cette

sélection permet d'exclure les tests statistiques reposant sur des hypothèses non respectées par les données.

On remarque qu'on s'est limité à une classification binaire dans cette première expérimentation (signal anormal contre signal normal). Nous montrerons par la suite que la méthodologie peut être utilisée pour déterminer l'origine de l'anomalie.

### 5.3.2 Deuxième expérimentation

On crée un jeu de données simulées similaires aux jeux de données précédents. Cependant, on supprime sur le jeu de données $B$ les variations sinusoïdales. Ainsi, le jeu $A$ est le même que précédemment, tandis que pour le jeu $B$, les séries temporelles ne présentant pas d'anomalie sont toutes supposées être des bruits gaussiens standards. Sur les jeux $A$ et $B$, les signaux ont une longueur choisie aléatoirement de manière uniforme entre 100 et 200.

Les caractéristiques du jeu $A$ dans les cas avec anomalie sont les mêmes que précédemment (5.1.1). La seule différence entre le jeu de données $A$ et le jeu de données $B$ est l'amplitude du saut pour le changement de moyenne, qui est deux fois plus petit dans le jeu $B$ pour rendre la détection plus difficile.

Ainsi, pour le jeu $B$, on a :

1. pour le changement de variance : pour chaque moteur $j$, les observations $Z_i^j$ suivent une loi $\mathcal{N}(0, \sigma^2)$ où $\sigma^2 = 1$ avant l'instant de rupture et où $\sigma$ est tiré uniformément sur $[1.01,\ 5]$ après l'instant de rupture.
2. pour le changement de moyenne : pour chaque moteur $j$, les observations $Z_i^j$ suivent une loi $\mathcal{N}(\mu, 1)$ avec $\mu = 0$ avant l'instant de rupture et où $\mu$ est tiré uniformément sur $[0.505,\ 2.5]$ après l'instant de rupture.
3. pour le changement de tendance : pour chaque moteur $j$, les observations $Z_i^j$ suivent une loi $\mathcal{N}(\mu, \sigma^2 = 1)$ où $\mu = 0$ avant l'instant de rupture et où $\mu$ augmente linéairement à partir de $0$ dès l'instant de rupture avec une pente choisie uniformément dans$[0.02,\ 3]$.

L'instant de rupture est toujours choisi entre la $\frac{1}{5}n(j)$-ième observation et la $\frac{4}{5}n(j)$-ième observation.

#### 5.3.2.1 Les indicateurs

Les indicateurs sont construits à partir des mêmes tests (test Mann–Whitney–Wilcoxon sur les moyennes, test de Kolmogorov-Smirnov sur les distributions, test de Fisher sur les variances.) et avec les mêmes jeux de paramètres que l'expérimentation précédente pour obtenir 810 indicateurs. On peut donc remarquer qu'on n'utilise toujours pas de test de changement de pente.

Les 6000 signaux sont divisés de la même façon que précédemment :

– 1000 signaux pour l'apprentissage
– 10 groupes de 500 signaux pour les tests.

TABLE *5.6: Taux de bonne classification obtenus avec les Random Forests en utilisant 810 indicateurs binaires. Pour l'ensemble de test, c'est la moyenne du taux de bonne classification qui est donnée avec son écart-type entre parenthèse.*

| Jeu | Taux de bonne classification sur l'ensemble d'apprentissage | Taux de bonne classification sur l'OoB | Taux de bonne classification sur l'ensemble test |
|---|---|---|---|
| $A$ | 0.977 | 0.923 | 0.935 (0.010) |
| $B$ | 0.971 | 0.912 | 0.923 (0.011) |

Une des principales différences par rapport à l'expérimentation précédente est que l'on ne se restreint plus à une classification binaire : on se propose de discriminer sur 4 classes $C_0, C_1, C_2$ et $C_3$ données dans la section 5.1.1 : sans anomalie, anomalie sur la moyenne, anomalie sur la variance, anomalie sur la pente.

Par ailleurs, en plus des Random Forests qui servent alors comme méthode de référence, nous utilisons, dans cette expérimentation, les classifieurs bayésiens naïf (CBN). Nous évaluons les performances sur les deux méthodes et nous étudions la matrice de confusion. Nous étudions également les taux de bonne classification pour chaque classe afin d'analyser la répartition des erreurs pour chaque classe.

#### 5.3.2.2 Résultats avec tous les indicateurs

Le tableau 5.6 résume les performances globales de classification obtenues avec les Random Forests en utilisant tous les indicateurs. Les Random Forests sont connus pour ne pas souffrir face aux problèmes de dimensionnalité, et pour limiter le sur-apprentissage ; les résultats obtenus confirment ces propriétés.

Le tableau 5.7 résume les performances dans le cas du classifieur bayésien naïf (CBN). Ces performances sont significativement en deçà des performances obtenues avec les Random Forests. Cela était attendu étant donné la redondance des indicateurs. En effet, on rappelle que la théorie des CBN repose sur l'indépendance conditionnelle des indicateurs sachant la classe.

TABLE *5.7: Taux de bonne classification obtenus avec un classifieur bayésien naïf en utilisant 810 indicateurs binaires. Pour l'ensemble de test, c'est la moyenne du taux de bonne classification qui est donnée avec son écart-type entre parenthèse.*

| Jeu | Taux de bonne classification sur l'ensemble d'apprentissage | Taux de bonne classification sur l'ensemble test |
|---|---|---|
| $A$ | 0.786 | 0.772 (0.017) |
| $B$ | 0.755 | 0.738 (0.018) |

TABLE *5.8: Jeu de données $A$ : Matrice de confusion avec tous les indicateurs utilisés dans le cas du classifieur bayésien naïf sur l'ensemble test.*

| | 0 | 1 | 2 | 3 | total |
|---|---|---|---|---|---|
| 0 | 1759 | 667 | 45 | 29 | 2500 |
| 1 | 64 | 712 | 50 | 3 | 829 |
| 2 | 7 | 2 | 783 | 37 | 829 |
| 3 | 32 | 7 | 195 | 595 | 829 |

Comme l'illustre la matrice de confusion reportée dans le tableau 5.8, les erreurs de classification ne sont pas concentrées dans une seule classe. Cette observation laisse penser que les indicateurs engendrés sont suffisants pour faire cette classification (ce qui était clair pour les Random Forests).

Quant aux deux jeux de données $A$ et $B$, on observe que l'on obtient des résultats quasiment aussi bon pour le jeu $B$ que pour le jeu $A$, bien que le jeu $B$ soit plus « difficile » à traiter par construction.

#### 5.3.2.3 Sélection d'indicateurs

Comme dans le cadre de l'expérimentation précédente, les résultats obtenus à l'aide des Random Forests sont très satisfaisants, mais le système ne serait pas acceptable par l'opérateur métier car le processus de formation des résultats est difficilement interprétable. En effet, faire analyser 810 indicateurs, même simples, n'est pas réaliste.

De plus, nous constatons que les performances obtenues avec le classifieur bayésien naïf (CBN) sont vraiment inférieures aux performances obtenues par les Random Forests, et cela pourrait être dû à la redondance forte des indicateurs.

Ce dernier inconvénient constitue un argument supplémentaire nous incitant à utiliser une méthode de sélection des indicateurs. Celle-ci se fait encore avec le processus de sélection *forward* mRMR. A chaque nouvel indicateur ajouté, un Random Forest et un CBN sont construits sur le même ensemble d'apprentissage, puis sont évalués sur le même ensemble test.

Les figures 5.11, 5.12, 5.13 et 5.14 résument les résultats obtenus avec les 100 premiers indicateurs sélectionnés. Encore une fois, le taux de bonne classification des Random Forests croit de manière monotone avec le nombre d'indicateurs. Cependant, après 25 et 30 indicateurs, selon que l'on traite le jeu de données $A$ ou $B$, les performances obtenues sur le jeu test stagnent. En pratique, cette stagnation fournit une indication sur le nombre d'indicateurs à retenir en utilisant le taux de bonne classification comme critère de qualité.

Les résultats obtenus avec les classifieurs bayésiens naïfs (CBN) sont un peu plus complexes à interpréter sur le jeu de données $B$ que sur le jeu $A$, mais confirment que la sélection est avantageuse. De plus, réduire le nombre d'indicateurs est bénéfique pour les CBN, qui après sélection, atteignent des performances semblables à celles des Ran-

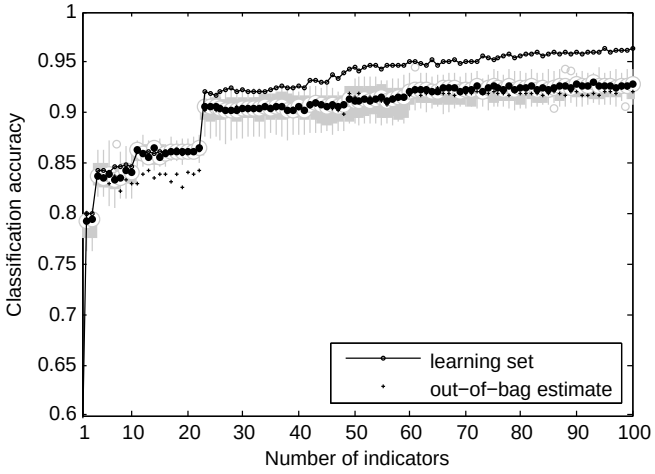


FIGURE *5.11:* ***Jeu de données*** $A$ ***Random Forest*** *: Taux de bonne classification sur l'ensemble d'apprentissage (cercle noir) en fonction du nombre d'indicateurs. Une box-plot donne le taux de bonne classification pour l'ensemble test avec la médiane (point noir dans un cercle blanc). L'estimation des taux de bonne classification obtenus par l'out-of-bag (OOB) est donnée par les croix.*

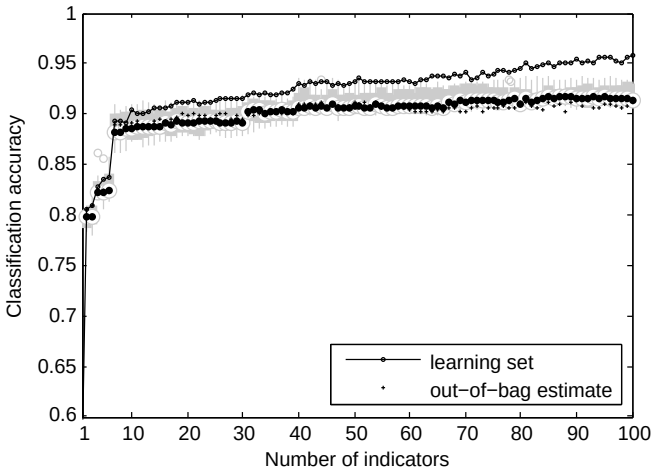


FIGURE *5.12:* ***Jeu de données*** $B$ ***Random Forest****, voir la figure 5.11 pour les détails.*

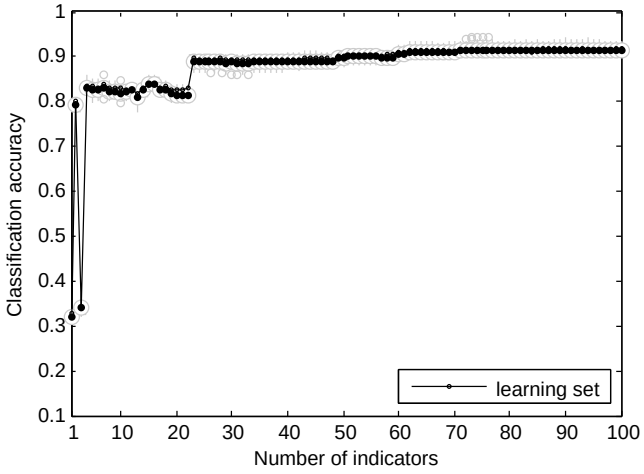


FIGURE *5.13:* ***Jeu de données*** $A$ ***Classifieur Bayésien Naïf*** *: Taux de bonne classification sur l'ensemble d'apprentissage (cercle noir) en fonction du nombre d'indicateurs. Une boxplot donne le taux de bonne classification pour l'ensemble test avec la médiane (point noir dans un cercle blanc).*

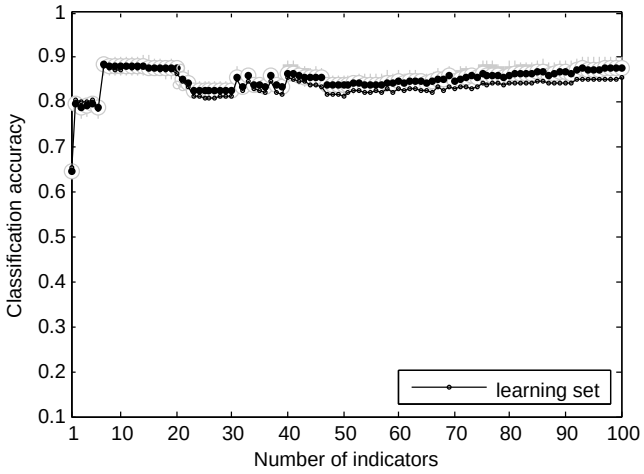


FIGURE *5.14:* ***Jeu de données*** $B$ ***Classifieur Bayésien Naïf****, voir la figure 5.13 pour les détails. Les performances peuvent se dégrader lorsque des indicateurs sont rajoutés. Cela peut être dû au fait que les indicateurs sélectionnés ne respectent pas l'hypothèse naïve.*

dom Forests. Les performances obtenues sur l'échantillon test sont presque identiques aux performances obtenues sur l'échantillon d'apprentissage. Cela peut s'expliquer naturellement par le fait que le classifieur utilise la fréquence de 1 (voir la section 4.3.3) pour chaque indicateur dans chaque classe. Or, dans l'échantillon d'apprentissage, on a au moins 250 observations de chaque classe, ce qui permet d'avoir des estimations précises et donc une décision plutôt stable. En pratique, pour les CBN, on pourrait sélectionner le nombre d'indicateurs, en utilisant les performances sur l'échantillon d'apprentissage sans utiliser d'échantillon test.

On peut constater qu'il y a des sauts dans les courbes de performances dans tous les cas. Cela indique que le classement proposé par l'algorithme mRMR n'est pas forcément optimal. Une des solutions serait d'utiliser une approche *wrapper* pour obtenir une meilleure sélection d'indicateurs. On a vu dans la section 4.4.2 que le principal désavantage de cette approche est d'être gourmande en calculs. Ici, ce désavantage pourrait être compensé par la rapidité de construction du classifieur bayésien naïf.

En se basant sur les résultats des figures 5.13 et 5.14, on peut sélectionner un nombre optimal d'indicateurs binaires tout en gardant un nombre maximum raisonnable pour éviter de surcharger l'opérateur métier. Par exemple, le tableau 5.9 donne les performances de classification du classifieur bayésien naïf en cherchant le nombre optimal d'indicateurs inférieur à 30.

TABLE *5.9: Taux de bonne classification obtenus avec un classifieur bayésien naïf en utilisant le nombre optimal d'indicateurs compris entre 1 et 30. Pour l'ensemble test, c'est la moyenne du taux de bonne classification qui est donnée avec son écart-type entre parenthèse.*

| Jeu | Taux de bonne classification sur l'ensemble d'apprentissage | Taux de bonne classification sur l'ensemble test | Nombre optimal d'indicateurs |
|---|---|---|---|
| $A$ | 0.896 | 0.891 (0.013) | 23 |
| $B$ | 0.883 | 0.881 (0.013) | 11 |

La figure 5.15 illustre de façon plus détaillée le phénomène de saut visible sur le taux de mauvaise classification pour chaque classe pour le jeu de données $A$. Cette figure montre les difficultés rencontrées pour différencier les sauts de moyenne et les sauts de tendance (on rappelle que dans les indicateurs choisis, il n'y a pas de tests spécifiques pour détecter les changements de tendance). Cependant, on constate que pour les deux types de changement, il y a un saut dans les performances dès que l'indicateur 23 est ajouté, ce qui pourrait nous amener à favoriser l'utilisation d'une sélection d'indicateurs avec approche *wrapper*.

Les performances du classifieur bayésien naïf ne valent pas celles des Random Forests, mais elles peuvent s'améliorer lorsque l'on réduit le nombre d'indicateurs. On peut le constater en comparant les tableaux 5.7 p. 72 (avec tous les indicateurs) et 5.9 p.76 (après réduction du nombre d'indicateurs). De plus, les indicateurs sélectionnés, asso-

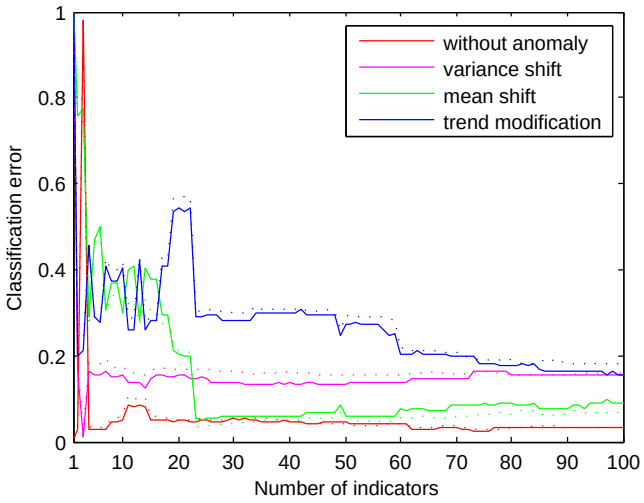


FIGURE *5.15:* ***Jeu de données*** $A$ ***Classifieur Bayésien Naïf*** *: Taux de mauvaise classification pour chaque classe sur l'ensemble d'apprentissage (traits continus) et sur l'ensemble test (traits pointillés, moyenne des taux).*

ciés avec leurs probabilités d'être positifs conditionnellement à chaque classe, peuvent être présentés à l'opérateur métier à partir d'un tableau semblable au tableau 5.10.

Dans cet exemple, le premier indicateur sélectionné $confu(2,3)$, est un indicateur utilisant un test Mann–Whitney–Wilcoxon avec une étape de confirmation telle que l'indicateur est positif uniquement s'il existe 2 fenêtres qui sont significatives dans un ensemble de 3 fenêtres consécutives (voir 4.1.1.1). Le classifieur bayésien naïf utilise les probabilités estimées pour parvenir à une décision : l'indicateur a peu de chance d'être positif s'il n'y a pas de changement ou si il y a un changement de variance. Et à l'opposé, il a de fortes chances d'être positif quand il y a un changement de moyenne ou un changement de tendance. Ce tableau n'explicite pas clairement comment la décision est prise par le classifieur bayésien naïf, mais il donne des indices facilement interprétables par l'opérateur métier.

#### 5.3.2.4 Conclusion

En partant d'un très grand nombre d'indicateurs, on arrive à des performances correctes en en sélectionnant un petit nombre (23 dans l'un des cas). Ces bonnes performances sont obtenues en associant ces 23 indicateurs à un simple classifieur bayésien naïf, même dans le cas où l'on a plusieurs classes.

Concernant l'interprétabilité, les tableaux de même type que le tableau 5.10, fournissant la probabilité qu'un indicateur vaille 1 conditionnellement à la classe, donnent de bons indices pour expliquer la décision du classifieur.

TABLE *5.10: Les 23 meilleurs indicateurs obtenus par mRMR pour le jeu de données $A$ et probabilité qu'ils vaillent 1 conditionnellement à la classe. Confu(k,n) est un indicateur de confirmation qui vaut 1 si le test de Mann–Whitney–Wilcoxon est positif dès qu'il existe pour n fenêtres consécutives, k fenêtres où le test est positif. Conff(k,n) est un indicateur de confirmation identique mais pour le F-test. Ratef($\alpha$) correspond à la positivité du F-test sur $\alpha \times m$ fenêtres quand il y a $m$ fenêtres au total. Lseqf($\alpha$) correspond à la positivité du F-test sur $\alpha \times m$ fenêtres consécutives quand il y a $m$ fenêtres au total. Lsequ($\alpha$) est le même que précédemment mais avec un test Mann–Whitney–Wilcoxon.*

| Type d'indicateur | Pas de changement | Variance | Moyenne | Tendance |
|---|---|---|---|---|
| confu(2,3) | 0.010 | 0.011 | 0.971 | 0.939 |
| F test | 0.020 | 0.83 | 0.742 | 0.779 |
| U test | 0.0273 | 0.03 | 0.977 | 0.952 |
| ratef(0.1) | 0.001 | 0.69 | 0.518 | 0.221 |
| confu(4,5) | 0.034 | 0.03 | 0.986 | 0.959 |
| confu(3,5) | 0.001 | 0.001 | 0.923 | 0.899 |
| U test | 0.02 | 0.022 | 0.968 | 0.941 |
| F test | 0.042 | 0.853 | 0.793 | 0.813 |
| rateu(0.1) | 0 | 0.001 | 0.906 | 0.896 |
| confu(4,5) | 0.019 | 0.02 | 0.946 | 0.927 |
| conff(3,5) | 0.052 | 0.721 | 0.54 | 0.121 |
| U test | 0.037 | 0.038 | 0.983 | 0.951 |
| KS test | 0.016 | 0.294 | 0.972 | 0.936 |
| confu(3,5) | 0.049 | 0.043 | 0.988 | 0.963 |
| F test | 0.030 | 0.841 | 0.77 | 0.801 |
| U test | 0.043 | 0.043 | 0.981 | 0.963 |
| lseqf(0.3) | 0.009 | 0.749 | 0.59 | 0.36 |
| rateu(0.1) | 0.001 | 0.002 | 0.896 | 0.895 |
| lsequ(0.1) | 0.062 | 0.06 | 0.992 | 0.949 |
| confu(3,5) | 0.025 | 0.021 | 0.963 | 0.936 |
| lseqf(0.3) | 0.008 | 0.732 | 0.656 | 0.695 |
| KS test | 0.016 | 0.088 | 0.955 | 0.93 |
| confu(3,5) | 0 | 0 | 0.003 | 0.673 |

### 5.3.3 Troisième expérimentation

#### 5.3.3.1 Introduction

Dans l'expérimentation précédente, les analyses nous ont amenés à suggérer l'utilisation des méthodes *wrapper* pour la sélection de variables. Pour cette expérimentation, on compare plusieurs méthodes de sélection de variables. Puisque les méthodes de type *wrapper* sont gourmandes en calculs, on va se limiter on au classifieur bayésien naïf qui fournit des résultats satisfaisants comme l'a conclus l'expérimentation précédente.

On simule un nouveau jeu de données similaire au jeu $A$ de la deuxième expérimentation. Les ensembles d'apprentissage et de test ont les mêmes caractéristiques : ils sont chacun composés de 6000 séries temporelles dont 3000 supposées sans anomalie (avec un bruit gaussien standard). On considère toujours les 3 mêmes types d'anomalies :

- dans le cas du changement de moyenne, $\mu$ passe de $0$ à un $\mu$ tiré aléatoirement dans $[1,5]$.
- dans le cas du changement de variance, $\sigma$ passe de $1$ à un $\sigma$ tiré aléatoirement dans $[2,6]$.
- dans le cas du changement de tendance, les anomalies ajoutent une tendance linéaire à partir du point de rupture, de façon à arriver à la fin du signal à une moyenne égale à un nombre tiré aléatoirement dans $[1,5]$.

Les longueurs des signaux sont toujours choisies aléatoirement dans $[100,200]$, et l'instant de rupture est toujours choisi aléatoirement entre la $\frac{1}{5}n$-ième observation et la $\frac{4}{5}n$-ième observation où $n$ est la longueur du signal.

Dans cette expérimentation, on se limite aux trois mêmes tests :

- le test U test ( test de changement de moyenne)
- le test de Kolmogorov-Smirnov (test de changement de distribution)
- le test de Fisher (test de changement de variance)

On fait varier la longueur de la fenêtre utilisée. Des confirmations sont également ajoutées.

Les paramètres utilisés pour le processus de binarisation permettent d'obtenir un ensemble de 814 indicateurs.

Le processus d'évaluation est modifié par rapport aux deux premières expérimentations. Pour chaque méthode de sélection de variables, le classifieur bayésien naïf est construit sur la moitié de l'ensemble d'apprentissage (en gardant les proportions entre les classes) et la sélection du meilleur sous-ensemble d'indicateurs est fait sur la deuxième moitié de l'ensemble d'apprentissage (en choisissant le plus petit sous-ensemble parmi ceux ayant obtenu les meilleurs taux de bonne classification). Le sous-ensemble d'indicateurs est ensuite évalué sur l'ensemble de test.

#### 5.3.3.2 Les méthodes de sélection de variables

Différentes méthodes de sélection de variables sont comparées. Deux méthodes de type filtre :

1. *Ranking* par Information Mutuelle : les variables sont ajoutées une à une en commençant par la variable ayant la plus forte information mutuelle avec la classe ;
2. mRMR : c'est la méthode utilisée dans les deux premières expériences (voir 4.4.3).

Les autres méthodes utilisées sont des méthodes de type *wrapper*. Les mesures de performances sont le taux d'erreur de classification et la probabilité d'erreur. Les méthodes *forward* (à chaque étape, le meilleur indicateur est ajouté) et *backward* (à chaque étape le plus mauvais indicateur est retiré) sont comparées, ainsi que les méthodes full *forward*/*backward*. Pour ces dernières, une phase *forward* est suivie d'une phase *backward* (et vice versa) jusqu'à ce que le résultat ne soit plus amélioré. Elles sont aussi appelées *floating search* dans Guyon et al. (2006)).

Par exemple, on démarre la recherche *backward* en trouvant le premier sous-ensemble optimal d'indicateurs, puis on procède à une recherche *forward* à partir de ce sous-ensemble pour en obtenir un meilleur. S'il y a une amélioration, la procédure est répétée en partant du dernier sous-ensemble en procédant à une recherche *backward* et ainsi de suite.

#### 5.3.3.3 Résultats

Les résultats sont résumés dans le tableau 5.3.3.3. Comme prévu, les méthodes de type *wrapper* obtiennent des performances qui dépassent celles de type filtre. On remarque également que la forte redondance des indicateurs binaires (qui vient de leurs constructions) influence défavorablement la méthode de *ranking* utilisant l'information mutuelle simple. En effet, cette dernière a tendance à sélectionner des indicateurs très redondants. La méthode de sélection mRMR contourne ce défaut mais les résultats ne sont pas optimaux.

TABLE *5.11: Taux de mauvaise classification sur l'ensemble test.*

| Méthode | Mesure de Perf. | # d'indicateurs | Taux d'erreur |
|---|---|---|---|
| filtre MI | Erreur | 422 | 0.139 |
| filtre mRMR | Erreur | 19 | 0.144 |
| Forward search | Erreur | 136 | 0.124 |
| Forward search | Probabilité | 207 | 0.123 |
| Backward search | Erreur | 27 | 0.131 |
| Backward search | Probabilité | 86 | 0.128 |
| Forward–Backward | Erreur | 92 | 0.124 |
| Forward–Backward | Probabilité | 123 | 0.124 |
| Backward–Forward | Erreur | 112 | 0.127 |
| Backward–Forward | Probabilité | 122 | 0.117 |

On peut également remarquer que l'utilisation de la probabilité d'erreur (plutôt que le taux de mauvaise classification) pour choisir les indicateurs, améliore les résultats lors de l'approche *wrapper*. Cela permet d'avoir un meilleur *ranking* des indicateurs que ne peuvent fournir les seules procédures de recherche.

Enfin, on constate que passer d'un simple algorithme de recherche *backward* à une recherche flottante permet d'améliorer les résultats. Cela peut s'expliquer par la tendance à sélectionner trop peu de variables dans le cas simple. Dans le cas d'une recherche *forward* flottante, les performances sont dégradées mais le nombre d'indicateurs sélectionnés est fortement diminué par rapport au cas simple.

#### 5.3.3.4 Conclusion

Ces résultats montrent que les approches de type filtre combinées avec les classifieurs bayésiens naïfs ne sont pas bien adaptées. Ils montrent également que minimiser la probabilité d'erreur (au lieu du taux de mauvaise classification) dans le cadre des processus de sélection de type *forward/backward*, permet d'obtenir un meilleur classement des indicateurs à chaque étape du processus.

Dans le cadre de cette expérimentation, les méthodes de type *wrapper* donnent en général des résultats comparables aux méthodes de type filtre et les résultats sont nettement meilleurs dans le cas des *floating search*.

Finalement, et pour résumer, dans notre cadre d'utilisation (indicateurs binaires très redondants), la *backward floating search*, cherchant à minimiser la probabilité d'erreur, donne de meilleurs résultats que les approches classiques généralement recommandées pour le classifieur bayésien naïf (telle qu'une méthode par filtre d'information mutuelle).

### 5.3.4 Quatrième expérimentation

Les experts conseillent fortement d'ajouter une étape de confirmation pour limiter les fausses alarmes. Dans cette dernière expérimentation, on souhaite étudier l'influence des indicateurs de confirmation et vérifier que cette étape est nécessaire. Au préalable, on peut tout même faire remarquer que dans les expérimentations précédentes (voir par exemple le tableau 5.10) plusieurs indicateurs de confirmation sont conservés après sélection, ce qui justifie l'ajout de cette étape de confirmation.

Pour cette expérimentation, on crée un nouveau jeu de données $C$ plus compliqué que dans les expérimentations précédentes. On considère un bruit issu d'une loi de Student à 3 degrés de liberté et une variance aléatoire.

Plus précisément, pour chaque moteur $j$, on suppose que l'on dispose $n$ variables aléatoires $Y_1^j, \ldots, Y_n^j$ indépendantes et identiquement distribuées selon $\mathscr{T}(3)$.

Puis chaque observation $Z_1^j, \ldots, Z_n^j$, est simulée telle que

$$Z_i^j \sim aY_i^j,$$

où $a$ est choisi aléatoirement dans $[0.5, 3]$.

Pour les changements, on utilise toujours les trois types de changements des expérimentations précédentes avec les configurations suivantes :

1. pour le changement de variance : après l'instant de rupture, une variance d'une valeur tirée aléatoirement dans $[1.05, 5]$ est ajoutée au signal ;
2. pour le changement de moyenne : après l'instant de rupture, un saut d'une valeur tirée aléatoirement dans $[0.3, 5]$ est ajoutée au signal ;
3. pour le changement de tendance : après l'instant de rupture, le signal voit sa moyenne augmenter de façon linéaire avec une pente ayant une valeur choisie uniformément dans$[0.02,\ 3]$.

Pour les indicateurs, quatre nouveaux tests sont intégrés :

- Test de Student pour comparer les moyennes dans le cas où les variances sont supposées égales ;
- Test de Student pour comparer les moyennes dans le cas où les variances sont supposées différentes et estimées sans erreur ;
- Test d'existence de pente non nulle ;
- Test de changement de pente.

Ainsi, pour chaque moteur $j$, on arrive à un ensemble de 2565 indicateurs $C_{all}$.

Pour l'étude de l'influence des indicateurs de confirmation, on considère également $C_w$, un sous-ensemble des indicateurs $C_{all}$ pour lequel tous les indicateurs de confirmation ont été supprimés : il en reste alors 945.

Le tableau 5.12 fournit les résultats avec et sans les indicateurs de confirmation.

Dans la figure 5.16, on peut comparer les performances obtenues avec et sans les indicateurs de confirmation en utilisant le processus de sélection mRMR. On constate que les indicateurs de confirmation permettent d'améliorer les performances générales. Par exemple, on peut remarquer à partir de la figure 5.16 qu'il faut utiliser plus de 20 indicateurs sans confirmation pour obtenir les mêmes performances sur l'*out-of-bag* (OOB) qu'avec 3 indicateurs avec confirmation. Ou encore, il faut 5 indicateurs sans confirmation pour obtenir les mêmes performances sur l'OOB qu'avec 2 indicateurs avec confirmation.

Les résultats obtenus sur ces données simulées sont corrects, même avec un nombre limité d'indicateurs. La sélection permet d'exclure les tests statistiques qui demandent des hypothèses qui ne sont pas respectées par les données.

TABLE *5.12: Taux de bonne classification obtenus avec un Random Forest. Pour le jeu $C_w$ (indicateurs sans confirmation), il y a 945 indicateurs. Pour le jeu $C_{all}$ (indicateurs avec confirmation), il y a 2565 indicateurs. $C_w$ et $C_{all}$ sont les mêmes données, mais ce qui les différencie, c'est que dans $C_w$ tous les indicateurs ayant une confirmation ont été supprimés.*

| Jeu | Taux de bonne classification sur l'ensemble d'apprentissage | Taux de bonne classification sur l'ensemble test |
|---|---|---|
| $C_{all}$ | 0.963 | 0.766 ( 0.016) |
| $C_w$ | 0.953 | 0.732 (0.017) |

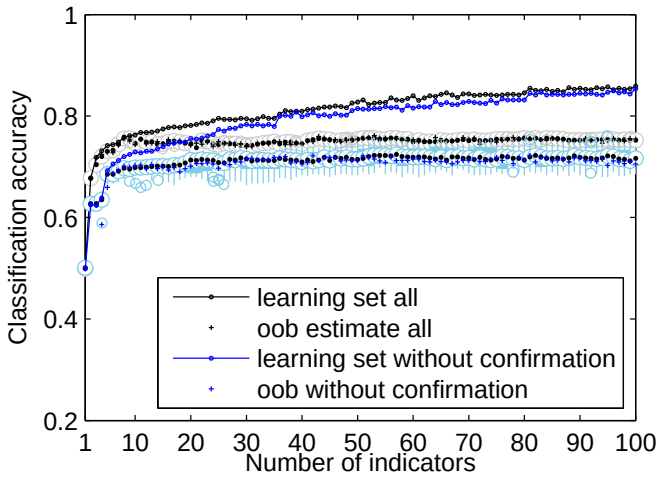


FIGURE *5.16:* ***jeu de données $C_{all}$ (noir) et $C_w$(bleu) Random Forest****, voir la figure 5.11 pour plus de détails. $C_w$ est un jeu d'indicateurs sur les mêmes données que $C_{all}$ mais tous les indicateurs de confirmation ont été supprimés.*

CHAPITRE 6

# Exploitation du système

Dans le chapitre précédent, on a évalué la méthodologie proposée sur des données simulées. Les résultats obtenus sont suffisamment intéressants pour qu'on envisage d'évaluer cette méthodologie sur des données réelles. Cependant, sur les données simulées, les labels sont connus avec précision, et on rappelle que c'est loin d'être le cas des données réelles. Or, certaines méthodes de classification peuvent perdre en précision lorsque les labels sont incertains. De plus, malgré la simplicité de la méthodologie présentée, il faut prévoir sa prise en main et donc son acceptation par les opérateurs et les experts métier avant d'envisager une labellisation précise mais coûteuse d'un gros ensemble de données.

Dans ce chapitre, on commence par montrer les performances obtenues sur des données réelles avant de présenter un outil permettant, d'un côté d'aider les utilisateurs à comprendre la décision et donc d'accepter la méthodologie, et de l'autre de montrer comment cet outil peut contribuer à obtenir un ensemble de labels avec le moins d'erreurs possibles.

## 6.1 Confrontation aux données réelles

### 6.1.1 Présentation des données

Les données utilisées pour évaluer la méthodologie sont issues des *Costumer Notification Report* (CNR). Les CNR sont des documents envoyés à la compagnie cliente. Ces documents sont rédigés par l'opérateur métier du *Customer Service Center* (CSC), éventuellement avec l'aide des experts. Lorsque qu'une alerte est émise, l'opérateur analyse les données du moteur correspondant et en cas de confirmation de l'anomalie, le CNR est rédigé. Plus précisément, un CNR contient les recommandations de maintenance ainsi que les données et les analyses effectuées sur ces données.

Alors, un label peut être donné au moteur après ces analyses et indiquer l'origine de l'anomalie. La détermination du label est une tâche difficile résultant d'une analyse pointue et d'une connaissance experte, ainsi on préfère, dans le cadre de cette confrontation aux données réelles, se restreindre à une classification à deux classes pour simplifier le problème :

- $C_0$ est la classe regroupant les moteurs sans anomalie ;
- $C_1$ est la classe regroupant les moteurs ayant une anomalie (d'origine indéfinie).

On peut remarquer que la base utilisée est constituée uniquement d'avions ayant

fait l'objet d'un rapport (CNR), ce sont donc, par définition, des avions qui ont connu une anomalie (confirmée par les opérateurs). Pour obtenir des données sans anomalie dans le cadre de cette étude, l'astuce est de considérer, à raison, que les avions de la base ont au moins deux moteurs. Ainsi on peut faire l'hypothèse que le moteur n'ayant pas provoqué d'alarme est sans anomalie. Cette hypothèse peut se justifier par le fait que dans la base CNR, il n'y a que des anomalies dont l'origine est le moteur (et non l'avion), ce qui signifie qu'une déviation devrait se voir uniquement sur les paramètres du moteur ayant une anomalie. D'ailleurs durant l'analyse, les données de l'ensemble des moteurs de l'avion sont étudiées pour s'en assurer. Ainsi, on néglige les mouvements anormaux qui pourraient apparaître sur les moteurs n'ayant pas provoqué d'alarme, alors que ces mouvements pourraient s'expliquer par le fait que le moteur sain cherche à compenser les défaillances du moteur ayant une anomalie.

Au moment où les données ont été récoltées pour ce travail, les applications de surveillance des moteurs d'avion fournissant des scores d'anomalie n'étaient pas suffisamment matures, et la calibration n'était pas encore au point. Ainsi, pour les données, on s'est restreint aux données normalisées de cette base CNR, c'est-à-dire que l'influence du contexte extérieur a été retirée (voir la section 4.2.3 sur l'étape de normalisation) pour rendre raisonnable l'hypothèse de stationnarité en absence d'anomalie.

Les moteurs avec anomalies sont rares, mais pour augmenter le nombre de moteurs dans la base, d'autres données de moteurs n'ayant a priori connu aucun évènement ont été ajoutées. Le défaut de cette hypothèse est que « non évènement » ne signifie pas forcément absence d'anomalie, et dans le cas où il y a effectivement eu un évènement, il est tout à fait possible que cet évènement n'ait pas été notifié.

Finalement, la base finale se compose de 1616 moteurs dont :

- 343 avec anomalie
- 1273 sans anomalie

Pour limiter les temps de calcul, on se restreint aux trois paramètres principalement étudiés par les opérateurs métier lors des analyses :

- DEGT, écart entre l'EGT (*Exhausted Gas Temperature*, Température de l'air à la sortie de la turbine haute pression) prédite et l'EGT mesurée ;
- GPCN25, écart entre le N2 (vitesse de rotation de l'arbre haute pression) prédit et le N2 mesuré ;
- GWFM, écart entre le *Fuel Flow* (flux du carburant dans la chambre de combustion) prédit et le *Fuel Flow* mesuré.

On pourra se reporter à la figure 2.1 (p.12) pour se représenter les paramètres.

Les indicateurs sont construits à partir des tests suivants (voir 5.2.1 pour obtenir des précisions sur les tests) :

- Test de Mann–Whitney–Wilcoxon pour l'égalité des moyennes avec un saut positif ;
- Test de Mann–Whitney–Wilcoxon pour l'égalité des moyennes avec un saut négatif ;

- Test de Kolmogorov Smirnov pour l'égalité des lois ;
- Test de Fisher pour l'égalité de la variance ;
- Test de Student pour la moyenne en supposant les variances égales ;
- Test de Student pour la moyenne en supposant les variances différentes mais précises ;
- Test d'existence de pente non nulle positive ;
- Test d'existence de pente non nulle négative ;
- Test de changement de pente.

Ainsi, pour ajouter de la précision et pour répondre à une attente des experts, en plus des tests déjà présentés dans les chapitres précédents, on ajoute une information donnant le sens (positif ou négatif) des sauts et des changements de pentes.

Les différents types de confirmation présentés dans les sections précédentes (voir par exemple la section 4.1.1.1) sont également ajoutés pour chaque test.

### 6.1.2 Résultats

#### 6.1.2.1 Cas monodimensionnel

Dans le cas monodimensionnel, on ne considère que la variable DEGT. L'ensemble des tests utilisés a pour conséquence d'engendrer 1368 indicateurs pour chaque moteur.

Les résultats obtenus en monodimensionnel sont donnés dans le tableau 6.1, avec les matrices de confusion obtenues avec Random Forest et avec le classifieur bayésien naïf (tableaux 6.2 et 6.3).

TABLE *6.1: Taux de bonne classification sur les données réelles en utilisant 1368 indicateurs dans le cas monodimensionnel.*

| Données | Méthode | Taux pour l'apprentissage | Taux pour le test |
|---|---|---|---|
| Réelles | Random Forests | 0.973 | 0.876 |
| Réelles | CBN | 0.706 | 0.726 |

TABLE *6.2: Matrice de confusion sur les données réelles en utilisant les Random Forests dans le cas monodimensionnel avec la totalité des 1368 indicateurs.*

| | pas d'alarme | alarme | total |
|---|---|---|---|
| sain | 1233 | 40 | 1273 |
| anomalie | 83 | 260 | 343 |

TABLE *6.3: Matrice de confusion sur les données réelles en utilisant un classifieur bayésien naïf dans le cas monodimensionnel avec la totalité des 1368 indicateurs.*

| | pas d'alarme | alarme | total |
|---|---|---|---|
| sain | 850 | 423 | 1273 |
| anomalie | 36 | 307 | 343 |

#### 6.1.2.2 Cas multidimensionnel

Dans le cas multidimensionnel, on considère les trois paramètres : DEGT, GPCN25 et GWFM [1].

On rappelle que, comme vu dans la section 5.1.2, la méthode s'étend naturellement au cas multivarié. Les indicateurs sont calculés sur chacune des dimensions, et l'on obtient 14256 indicateurs pour chaque moteur.

Les résultats pour le Random Forest et les classifieurs bayésiens naïfs sont donnés dans le tableau 6.4.

TABLE *6.4: Taux de bonne classification sur les données réelles multidimensionnelles en utilisant la totalité des 14256 indicateurs.*

| Données | Méthode | Taux pour l'apprentissage | Taux pour le test |
|---|---|---|---|
| Réelles | Random Forests | 0.991 | 0.861 |
| Réelles | CBN | 0.477 | 0.472 |

La matrice de confusion pour les Random Forests est donnée dans le tableau 6.5.

TABLE *6.5: Matrice de confusion dans le cas multidimensionnel en utilisant les Random Forests et les paramètres DEGT, GPCN25 et GWFM avec la totalité des 14256 indicateurs.*

| | pas d'alarme | alarme | total |
|---|---|---|---|
| sain | 1240 | 33 | 1273 |
| anomalie | 87 | 256 | 343 |

1. rappels :
   - DEGT, écart entre l'EGT (*Exhausted Gas Temperature*, Température de l'air à la sortie de la turbine haute pression) prédite et l'EGT mesurée ;
   - GPCN25, écart entre le N2 (vitesse de rotation de l'arbre haute pression) prédit et le N2 mesuré ;
   - GWFM, écart entre le *Fuel Flow* (flux du carburant dans la chambre de combustion) prédit et le *Fuel Flow* mesuré.

#### 6.1.2.3 Interprétation

Dans le cas des Random Forests, les résultats obtenus en monodimensionnel et multidimensionnel sont comparables, mais le sur-apprentissage est plus marqué dans le cas multidimensionnel.

Avec le classifieur bayésien naïf (CBN), les performances se dégradent fortement dans le cas multidimensionnel et ces dégradations peuvent s'expliquer par l'ajout d'un grand nombre d'indicateurs induisant encore plus de possibilités de redondances.

En se limitant aux 20 meilleurs indicateurs sélectionnés par la méthode de sélection CMIM (voir la section 4.4.3), les performances s'améliorent considérablement pour le classifieur bayésien naïf (voir le tableau 6.6.) L'amélioration des résultats pour le classifieur bayésien naïf (CBN) peut s'expliquer par le fait que la méthode de sélection utilisée prend en compte la redondance entre les indicateurs candidats et les indicateurs déjà sélectionnés.

TABLE *6.6: Matrice de confusion à partir des 20 meilleurs indicateurs en utilisant un classifieur bayésien naïf et les paramètres DEGT, GPCN25 et GWFM.*

| | pas d'alarme | alarme | total |
|---|---|---|---|
| sain | 996 | 277 | 1273 |
| anomalie | 81 | 262 | 343 |

Les résultats se dégradent de façon non négligeable pour les Random Forests comme on peut le constater dans la matrice de confusion donnée dans le tableau 6.7, mais cette perte de performance peut être acceptée par l'opérateur métier en échange d'une meilleure interprétabilité.

TABLE *6.7: Matrice de confusion à partir des 20 meilleurs indicateurs en utilisant les Random Forests et les paramètres DEGT, GPCN25 et GWFM.*

| | pas d'alarme | alarme | total |
|---|---|---|---|
| sain | 1156 | 117 | 1273 |
| anomalie | 195 | 148 | 343 |

Pour aider à l'interprétabilité des résultats par l'opérateur métier, le tableau 6.8 donne les 5 premiers indicateurs sélectionnés et en fournit la description. On peut par exemple remarquer que le premier indicateur porte sur l'étude de la variance pour le paramètre DEGT. La taille de la fenêtre utilisée est 50, il n'y a pas de lissage et le pas entre deux fenêtres consécutives est de 10. Le deuxième indicateur porte toujours sur le paramètre DEGT, et il consiste à déterminer via un test d'homogénéité sur des populations de taille 25 (pour une taille de fenêtre 50), et sur un signal lissé par une moyenne mobile, s'il y a une pente positive. Le pas entre deux fenêtres consécutives est de 5.

TABLE *6.8: Description des 5 meilleurs indicateurs sélectionnés par CMIM pour le jeu de données réelles.*

| Tests | Niveau du test | Taille de fenêtre | Lissage | Pas | Paramètres d'alerte |
|---|---|---|---|---|---|
| F-Test | 0,01 | 50 | 1 | 10 | DEGT |
| Pente pos. | 0,01 | 50 | 5 | 5 | DEGT |
| Pente pos. | 0,05 | 30 | 5 | 5 | GPCN25 |
| Chgt pente | 0,01 | 65 | 1 | 10 | DEGT |
| F-Test | 0,01 | 30 | 5 | 5 | GWFM |

Le tableau 6.9 fournit des informations supplémentaires à l'opérateur métier. Il donne pour les 5 premiers indicateurs par la méthode CMIM, la probabilité qu'ils vaillent 1 conditionnellement à la classe, c'est-à-dire au type d'anomalie s'il y en a une. Par exemple, l'indicateur numéro 1 (changement de variance sur le paramètre DEGT) aura plutôt tendance à valoir 1 dans les cas où le moteur appartient aux classes correspondant aux anomalies de type 1, 2 et 4.

TABLE *6.9: Les 5 meilleurs indicateurs obtenus par CMIM pour le jeu de données réelles et probabilité qu'ils vaillent 1 conditionnellement à la classe.*

| Tests | Pas d'anomalie | Anomalie 1 | Anomalie 2 | Anomalie 3 | Anomalie 4 |
|---|---|---|---|---|---|
| F-Test | 0.498 | 0.818 | 0.846 | 0.666 | 0.838 |
| Pente positive | 0.136 | 0.090 | 0.115 | 0.25 | 0.139 |
| Pente positive | 0.012 | 0 | 0.038 | 0 | 0.007 |
| Changement pente | 0.728 | 1 | 0.961 | 0.916 | 0.985 |
| F-Test | 0.740 | 1 | 0.923 | 0.916 | 0.897 |

#### 6.1.2.4 Conclusion

Finalement les résultats sur les données réelles sont a priori très satisfaisants. Les tableaux fournis tels que 6.8 et 6.9 aident à l'interprétabilité et semblent constituer un atout supplémentaire, justifiant la mise en place de la méthodologie dans un cas réel à grande échelle.

Cependant, en étudiant un peu plus profondément les données, on peut constater quelques incohérences au niveau de la labellisation choisie, en particulier parce que non labellisation ne signifie pas toujours absence d'anomalie sur les données. Par exemple, la figure 6.1 est l'illustration qu'un moteur qui a été labellisé « sans anomalie » peut présenter des sauts importants. Ainsi on constate dans le tableau 6.9, que lorsqu'un

indicateur a tendance à valoir 1 (resp. 0) quand il y a une anomalie, il a également tendance à valoir 1 (resp. 0) quand il n'y a pas d'anomalie, ce qui rend l'interprétation difficile. Autrement dit, à l'œil nu, les indicateurs n'ont pas l'air de discriminer les moteurs normaux des moteurs anormaux : intuitivement, l'expert aimerait voir des indicateurs valant 1 quand il y a une anomalie et 0 sinon. Cette constatation autorise le doute sur l'hypothèse faite qui consiste en la stationnarité des paramètres normalisés en l'absence d'anomalie.

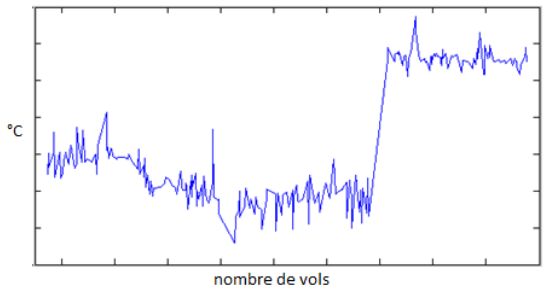


FIGURE *6.1: Exemple de moteur qui a été labellisé « sans anomalie ».*

Les résultats obtenus dans le tableau 5.10 (p.78) portant sur les données simulées est bien plus intuitif et donc interprétable. Le premier indicateur sélectionné a tendance à valoir 0 lorsqu'il n'y a pas d'anomalie et à valoir 1 lorsqu'il y a un changement de moyenne ou de tendance dans le signal.

Ainsi, il est possible que les paramètres considérés et donc les indicateurs qui en découlent, ne soient pas pertinents pour faire la classification dans le cas réel. Plus précisément, il est possible que la normalisation faite sur les données ne soit pas suffisante. Pour contourner ce type de difficulté et permettre tout de même d'utiliser la méthodologie sur des données réelles, il serait intéressant de disposer des scores d'anomalie fournis par les applications de surveillance afin d'éviter des phénomènes de sauts importants en l'absence d'anomalie.

Cependant, une autre partie du problème peut venir du fait que les labels sont très bruités, et qu'un passage en revue de tous les labels soit nécessaire pour limiter les erreurs de classification. En effet, comme il est signalé dans Dietterich (2000), les méthodes ont tendance à apprendre l'erreur en sur-pondérant les données ayant des labels erronés. Pourtant, étant donné le grand nombre de moteurs en service, il n'est pas réaliste de chercher à labelliser méticuleusement tous les moteurs.

## 6.2 Labellisation et outil de labellisation

### 6.2.1 L'importance de la labellisation

Des tentatives d'évaluation sur des données réelles Snecma ont permis de mettre en avant la fréquente absence de labels, ou plus précisément ont mis en évidence le travail nécessaire pour les récupérer. D'une part, les compagnies clientes ne font pas systématiquement de retours suite aux recommandations envoyées par le *Customer Service Center*, d'autre part, les analyses sont faites par différents opérateurs sur différents sites et les diagnostics portent sur différents composants. Par conséquent, pour plusieurs moteurs, les informations qui serviraient de labels peuvent être difficiles à comprendre, non directement disponibles ou tout simplement inexistantes.

Dans la section précédente, cette absence d'information a mené à une hypothèse consistant à affirmer qu'une donnée sans label peut être considérée sans anomalie. Cependant, cette hypothèse n'est pas toujours vérifiée. Lorsque des labels sont tout de même disponibles, un dépouillement un peu plus approfondi de la base de données a pu mettre en avant quelques incohérences sur ces labels. Ces incohérences peuvent être dues à des erreurs lors de la retranscription des résultats de l'analyse dans la base ou à une incertitude de l'opérateur métier lors de l'analyse.

Mais les labels sont importants car ils contribuent à l'analyse des résultats. Ils servent par exemple à évaluer une méthodologie de diagnostic. En effet, cette évaluation nécessite l'accès à une base contenant des données dont les résultats des diagnostics sont connus. De plus, pour une méthodologie utilisant peu d'a priori, comme celle présentée dans cette étude, la disponibilité des diagnostics est nécessaire pour faire émerger des « règles » de dépendances entre certains paramètres et les résultats de diagnostic ou plus précisément pour pouvoir utiliser des méthodes supervisées.

Enfin, on peut également faire remarquer que disposer de labels peut contribuer à la formation des nouveaux opérateurs ou aider à la calibration des applications de surveillance sur des données réelles.

### 6.2.2 Étapes de labellisation

Un travail de centralisation des données serait une première étape, certes complexe, mais nécessaire pour diminuer le temps d'accès à un maximum d'informations sur les données.

Une deuxième étape serait l'analyse de toutes les données pour leur attribuer un label, mais on rappelle que rien que pour le moteur CFM56, il y a plus de 24500 moteurs en service. Il n'est donc pas envisageable de faire analyser toutes les données disponibles. Il faut donc disposer d'un processus adéquat de labellisation pour limiter le temps et l'effort nécessaires.

Mais avant de parler du processus, attardons-nous sur la labellisation elle-même, et pour cela reprenons les définitions de la section 4.2.2.

On considère que l'on dispose d'un premier ensemble de $N$ moteurs $M_1, \ldots, M_N$ et

des « informations » sur ces moteurs. A chaque moteur $i$ est associé un label $L(i)$ (voir la figure 4.1 p.32).

Pour chaque moteur $M_i$, un vecteur des indicateurs $X_i$ est calculé, et ainsi l'information sur les $N$ moteurs est résumée dans $X_1, \ldots, X_N$.

Comme on l'a vu, les labels peuvent être erronés. Il faut donc pouvoir les corriger lors des analyses. La méthodologie classique consistant à corriger les labels en amont, c'est-à-dire lors d'une première analyse des données, n'est pas toujours envisageable étant donné le nombre de moteurs et la connaissance experte nécessaire pour ce nettoyage.

Ainsi ici, on préfère utiliser une labellisation a priori fournie par le modèle qui a été appris sur un échantillon d'apprentissage de taille raisonnable ou dont les labels sont déjà disponibles. Plus précisément, notons

$$\mathscr{A}_v = ((X_1, L(1)), \ldots, (X_N, L(N)))$$

l'ensemble des données d'apprentissage correspondant à des couples (moteurs, labels) où les labels ont été fournis par un ensemble d'experts.

On note $\mathscr{A} = ((X_1, \hat{L}(1)), \ldots, (X_N, \hat{L}(N)))$ l'ensemble des données d'apprentissage avec les labels estimés par la méthodologie. Ces estimations de labels peuvent différer des labels fournis par les experts.

Notons $\mathscr{E}_{\mathscr{A}}$ le taux d'erreur sur les labels :

$$\mathscr{E}_{\mathscr{A}} = \sum_{i=1}^{N} \mathbf{1}_{\{\hat{L}(i) \neq L(i)\}}$$

Une première étape consiste donc à déterminer un modèle qui minimise $\mathscr{E}_{\mathscr{A}}$ sur l'échantillon d'apprentissage tout en évitant le sur-apprentissage. Pour rappel, à l'issue de la phase d'apprentissage, les indicateurs adéquats ont été sélectionnés et le modèle de fusion/classification a été calibré.

L'étape suivante consiste alors à confronter sur les données d'apprentissage ou sur des nouvelles données, les labels $L$ définis par les experts et les labels $\hat{L}$ estimés par le modèle. Cela se fait à l'aide d'un outil de labellisation.

### 6.2.3 Outils de labellisation

Pour arbitrer entre $L$ et $\hat{L}$, l'expert métier a besoin des « informations » qui ont mené à ces choix. Pour cela, il faut disposer d'une interface affichant ces informations. Ensuite, lorsque l'expert fait un choix, il faut qu'il puisse enregistrer ce choix, d'où le nom d'outil de labellisation.

Cet outil permet donc d'analyser les données et de vérifier quelle décision $\hat{L}(i)$ est suggérée par le modèle pour un moteur $M_i$. On peut alors se retrouver dans six cas :

1. Le moteur $M_i$ a un label $L(i)$, ce label est en accord avec la décision du modèle : $L(i) = \hat{L}(i)$ et l'opérateur métier approuve ;
2. Le moteur $M_i$ a un label $L(i)$, ce label est en désaccord avec la décision du modèle : $L(i) \neq \hat{L}(i)$ et l'opérateur métier n'accepte pas la décision du modèle ;

3. Le moteur $M_i$ a un label $L(i)$, ce label est en désaccord avec la décision du modèle : $L(i) \neq \hat{L}(i)$ mais l'opérateur métier accepte la décision du modèle ;
4. Le moteur $M_i$ a un label $L(i)$, ce label est en accord avec la décision du modèle : $L(i) = \hat{L}(i)$ et l'opérateur métier est en désaccord avec la décision du modèle et donc avec le label du moteur ;
5. Le moteur $M_i$ n'a pas de label, l'opérateur métier accepte la décision du modèle ;
6. Le moteur $M_i$ n'a pas de label, l'opérateur métier n'accepte pas la décision du modèle.

Dans les situations correspondant au troisième cas, on peut supposer que les labels $L$ sont erronés, en particulier si les analyses ont été faites par différents experts. Ainsi, le but de l'outil de labellisation est de permettre la correction de ces labels. L'opérateur métier demande à être acteur, mais finalement sa participation reste indispensable. Cette participation de l'opérateur métier à la phase d'apprentissage et de validation peut contribuer grandement à la mise en confiance de l'opérateur métier vis-à-vis du modèle. Et parallèlement, cette participation permet d'aider aux nettoyages de la base de données. Or, avoir une base de données correctement labellisées est un minimum pour garantir un contrôle de l'erreur de généralisation.

Ces bons résultats pourront ensuite contribuer à l'acceptation d'un modèle un peu moins compréhensible mais permettant de meilleures performances comme les Random Forests.

### 6.2.4 Cas avec plusieurs experts

Dans le cas où plusieurs experts sont disponibles, les labels que ces derniers proposent peuvent être contradictoires. On peut d'ailleurs faire remarquer que le modèle peut être vu comme un expert supplémentaire. Dans ce cas, une des solutions est de procéder par un vote. Si par ailleurs, on suppose que les experts peuvent associer une confiance (on peut aussi gérer une pertinence de chaque espère), on peut utiliser un vote pondéré pour déterminer le modèle.

Formellement, pour un moteur $M_i$, on suppose que $ne$ experts, notés $(Ex_1, \ldots, Ex_{ne})$, ont donné un label. On note $L^{Ex_l}(i)$, le label donné par l'expert $Ex_l$ au moteur $M_i$. Dans le cas où une confiance est disponible, on peut alors avoir une formule de choix du label donnée par :

$$L(i) = \mathbf{1}_{Form > 0.5}$$

où $Form$ est donné par

$$Form = \sum_{l \in \{1,\ldots,ne\}} L^{Ex_l}(i) * Conf_{Ex_l}(i)$$

où $Conf_{Ex_l}(i)$ est la confiance normalisée donnée par l'expert $Ex_l$ au moteur $M_i$. On entend par normalisée la somme des confiances de tous les experts pour un moteur $M_i$ vaut $1$.

## 6.3 Interprétabilité

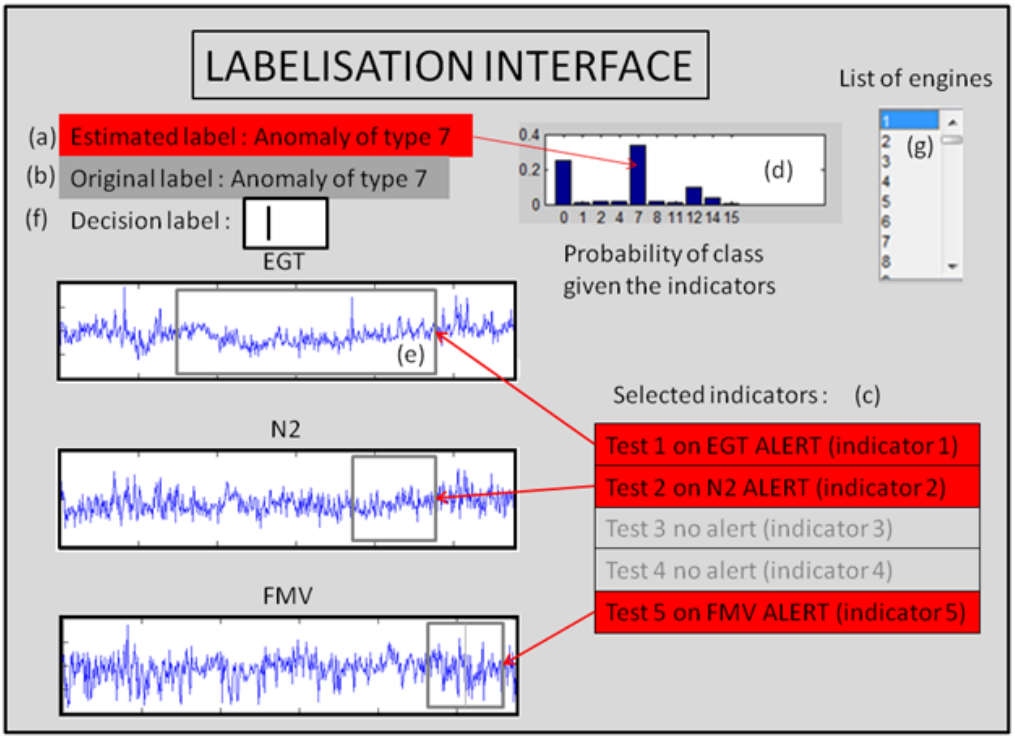


FIGURE *6.2: Exemple d'interface d'analyse et de labellisation.*

Au-delà de l'amélioration des performances de classification obtenues par la méthodologie, l'accent est mis sur l'interprétabilité des résultats par l'homme métier. En effet, pour prendre une décision finale l'opérateur métier a besoin d'avoir confiance. Cette confiance est rendue possible si le résultat est interprétable.

Une bonne représentation visuelle est une solution pour aider à l'interprétabilité des résultats. Jusqu'à présent, peu d'efforts ont été faits pour associer le data mining et la visualisation pour l'opérateur métier. Pourtant comme le titre [2] de Vellido et al. (2011) l'indique, on peut souvent considérer qu'une image bien choisie vaut mieux que mille mots. Ainsi, la visualisation peut être bénéfique à l'extraction d'informations. Aujourd'hui, on assiste, notamment avec la baisse des prix des supports de stockage, à une explosion du nombre de données disponibles. Ces données contiennent des informations intéressantes mais difficilement exploitables. Le challenge est d'en extraire des connaissances.

Les recherches dans ce domaine sont intenses mais sont loin d'être abouties. Cependant, il est intéressant de faire remarquer que les hommes métier auront tendance à préférer les méthodes proposant une visualisation. En effet, même si les résultats

2. Seeing is believing : The importance of visualization in real-world machine learning applications.

peuvent être un peu moins performants, ils sont, par nature, plus facilement interprétables.

Par exemple, on peut détecter rapidement une anomalie sur une représentation bien choisie. De plus, en permettant à l'homme métier d'interagir avec les images, on exploite ses connaissances, ce qui permet d'augmenter la confiance lors d'un processus de décision.

Ainsi, en proposant à l'opérateur métier de modifier un label, il participe à la correction des labels de la base et peut ensuite mesurer directement l'influence de ses corrections sur les performances de la méthodologie.

Les paramètres des méthodes doivent être représentables et compréhensibles à travers la visualisation : les paramètres doivent pouvoir être ajustés à l'aide d'un contrôle visuel. Ainsi, idéalement l'opérateur métier aurait le choix de la méthode utilisée et constaterait visuellement les conséquences de ce choix. Cependant, dans notre cas et dans dans un premier temps, il n'est possible de modifier que le label.

Finalement, cette visualisation peut servir à montrer que la méthodologie utilise le même cheminement de pensée que l'expert et peut donc plus facilement arriver à une acceptation de la méthodologie. Dans cette thèse, on propose une solution permettant de faciliter ce travail à travers une interface et en proposant un processus de labellisation.

### 6.3.1 Les difficultés de mise en œuvre

Les résultats fournis par la méthodologie peuvent être très bons, mais malgré cela, les opérateurs métier ont besoin de comprendre comment la décision a été prise, afin d'être en mesure de confirmer ou non une décision difficile à prendre aujourd'hui. C'est d'ailleurs pour cette raison que, le choix de la visualisation ainsi que la sélection d'indicateurs sont des étapes nécessaires pour que l'opérateur prenne confiance en la méthodologie.

En résumé, il faut bien choisir l'espace de représentation des données pour une compréhension optimale. Il faut que les techniques d'interaction et les algorithmes soient adaptés à la dite interaction. Et comme dans le cas de l'aéronautique, il faut d'excellentes performances, il y a une priorité pour un choix d'algorithmes efficaces avant d'être rapides.

Une mise à jour de la phase d'apprentissage consistant en la sélection des indicateurs et en l'apprentissage du modèle de classification peut se faire parallèlement à l'analyse par l'opérateur métier.

### 6.3.2 Suivi d'une flotte

A un instant donné, présent ou passé, on aimerait connaître l'état d'une flotte. Pour cela, on peut projeter sur une « carte » l'ensemble des moteurs à cet instant. On extrait l'historique disponible pour chacun des moteurs tout en rejetant les données trop anciennes pour être pertinentes selon l'avis des experts. On peut également se restreindre à la dernière intervention sur le moteur (*water-wash*, changement de pièce,

dépose moteur, ...).

### 6.3.3 Perspective

L'interprétabilité peut contribuer au paramétrage des modèles par les experts eux-mêmes : on parle alors de *visual data mining*, cette notion est développée dans Keim et al. (2010).

Ainsi, plutôt qu'une amélioration continuelle des outils utilisant les nouvelles connaissances en analyse de données, l'idée serait d'intégrer le *visual data mining* dans le processus de décision.

## 6.4 Discussions

Avant de conclure, il faut signaler qu'il y a une différence entre les diagnostics et la prise de décision. Jusqu'ici quand on parlait de décision, celle-ci portait sur la présence ou non d'une anomalie, ce qui se rapportait essentiellement à un diagnostic. Pour la prise de décision effective concernant la démarche à suivre, il faudrait rajouter une information sur les coûts afin de les minimiser. En d'autres termes, il n'y a pas de décision sans gestion des connaissances (*knowledge management*).

Une décision ne peut pas être prise trop tard à cause de la maintenance préventive, sauf si un jour on prouve que l'on peut rallonger les délais, mais cela nécessite une garantie de fiabilité des résultats.

Pour l'instant, on reste dans le cadre du diagnostic, et on propose une amélioration du processus de labellisation, de façon à montrer que la méthodologie fonctionne et pour inciter à encore plus nettoyer la base de données en amont.

La figure 6.3 donne un exemple de l'architecture générale.

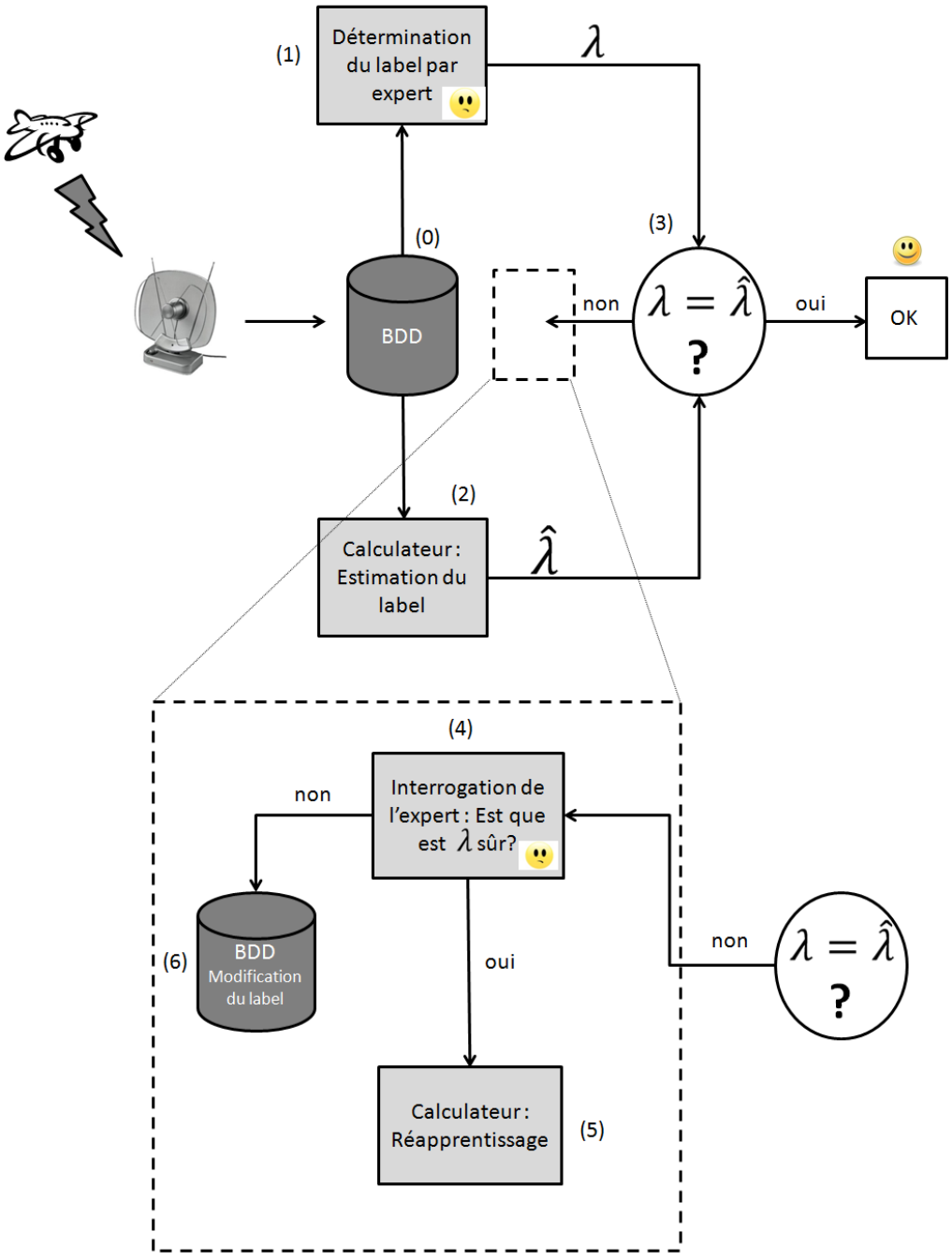


FIGURE *6.3: Résumé du traitement de l'information. (0) Les données sont stockées dans une base de données. (1) L'expert donne des labels. (2) Le modèle estime des labels. (3) Les labels sont comparés. (4) En cas de désaccord, l'expert vérifie son label. (5) Si le modèle a tort → réapprentissage, (6) sinon modification du label à l'aide de l'outil de labellisation.*

## 6.5 Applications génériques

L'architecture de la méthodologie proposée a été pensée pour s'adapter à tout type de données, et peut donc très facilement être exploitée pour répondre à différentes problématiques où l'estimation du label est importante. Dans cette section, on donne des cas réels où la méthodologie peut être utilisée.

### 6.5.1 Fusion des données

Dans cette thèse, on s'est limité à la fusion d'indicateurs correspondant tous à des décisions binaires ou plus précisément à des résultats de tests de rupture faits sur des séries temporelles. En réalité la méthodologie peut fusionner tout type de données sans que ces données soient nécessairement des décisions : il suffit de les convertir au format binaire.

### 6.5.2 Calibration des algorithmes

Une phase essentielle mais loin d'être triviale lors de l'élaboration d'une application de surveillance est la calibration des différents algorithmes. Les experts peuvent vérifier manuellement que les résultats fournis par les algorithmes sont ceux qui sont attendus. Ils peuvent également mettre en place un vecteur des réponses attendues pour obtenir directement une mesure de performance. Cependant, lorsque les résultats sont loin des résultats attendus (pour rappel, on veut limiter le nombre de fausses alarmes et maximiser le taux de bonne détection), il peut arriver que les algorithmes utilisés ne soient pas adaptés, mais surtout qu'ils ne soient pas bien calibrés.

Dans le cas où le vecteur de réponses d'une application de surveillance est disponible, la génération d'un grand nombre d'indicateurs permet de mettre en compétition un très grand nombre d'algorithmes et avec différents types de calibration. Le processus de sélection des indicateurs permet implicitement de souligner les algorithmes les plus pertinents pour optimiser les performances de l'application de surveillance avec les calibrations adéquates.

### 6.5.3 Détection d'anomalie sur un sous-système

Dans le cadre de la surveillance d'un sous-système spécifique du moteur, des indicateurs sont parfois définis par les experts métier pour suggérer si une anomalie est présente à un instant donné. Mais la définition d'une règle permettant de confirmer à partir d'une succession d'indicateurs s'il y a anomalie ou pas, n'est pas toujours une tâche aisée.

Par exemple, lorsque des indicateurs sont sous forme d'un code couleur où typiquement les couleurs utilisées sont le vert, le orange et le rouge (la couleur traduit l'état de santé courant du sous-système surveillé), il faudrait une règle basée sur les dernières couleurs visualisées pour conclure sur l'état réel de ce sous-système.

Cette règle n'est pas toujours évidente à déterminer, cependant, les experts sont capables de déterminer l'état du sous-système considéré à partir d'un historique (en utilisant par exemple les 5 derniers indicateurs courants). Établir des règles expertes donnant une estimation de l'état du sous-système à partir des 5 derniers indicateurs est possible, mais une vérification complète de ces règles peut révéler que des estimations inadéquates peuvent être faites.

Ainsi, une des possibilités offerte par la méthodologie est de proposer aux experts de corriger, à travers une interface, les estimations obtenues par les règles, afin d'obtenir une base d'apprentissage de données correctement labellisées et de créer une règle propre à la méthodologie à partir de cette base, permettant alors de supprimer les incohérences obtenues par les règles expertes.

CHAPITRE 7

# Conclusion et perspectives

## 7.1 Conclusion

Cette thèse a commencé par montrer l'importance de la détection des prémices d'une panne de moteurs pour les constructeurs tels que Snecma, en mettant en avant les conséquences logistiques, financières ou sur la réputation des compagnies aériennes. En effet, une panne non détectée ou détectée trop tard peut engendrer des retards et des coûts de réparation imprévus. Le travail mené dans la thèse a permis de faire un point sur les méthodes employées par Snecma pour détecter les prémices d'une panne, en mettant cependant l'accent sur le *Health Monitoring* des moteurs d'avions. L'analyse de données prélevées durant les vols n'est pas une tâche triviale, c'est même un travail d'experts motoristes. La thèse avait pour but d'assister les experts dans l'exploitation des données, en commençant par un prétraitement de celles-ci pour les rendre compatibles avec les méthodes d'apprentissage automatique, permettant ensuite une aide à la calibration des applications de surveillance et la fusion de leurs résultats pour pouvoir prendre une décision unique. Une fois le prétraitement fait, la mise en place des méthodes d'apprentissage automatique a permis d'atteindre le but recherché.

Cependant, la tentative d'évaluation de la méthodologie proposée sur des données réelles a montré une difficulté d'accès voire une absence d'informations claires et précises (des labels) sur les moteurs. Ainsi pour faciliter cette évaluation, une interface de labellisation a été proposée. Elle est aujourd'hui en cours de développement, et elle a pour but de faciliter la labellisation des moteurs par les experts tout en les aidant à prendre une décision finale en proposant aux experts des visualisations adéquates des résultats obtenus par la méthodologie.

Finalement, tout le travail mené durant la thèse va permettre à un motoriste tel que Snecma à mieux détecter automatiquement les prémices de pannes et, le cas échéant à aider l'expert à faire un diagnostic du moteur à partir de l'analyse des données. Cette assistance est d'autant plus acceptable par l'expert métier que la méthodologie proposée n'est pas sous forme de boite noire, ainsi le savoir expert peut facilement être exploité.

D'un autre côté, la thèse a permis également de montrer que l'on peut classifier des signaux multidimensionnels à partir d'une agrégation de détecteurs simples mais nombreux. Dans cette thèse, ce sont les résultats de tests statistiques élémentaires qui sont utilisés pour résumer chaque signal en un ensemble d'indicateurs binaires. Chaque indicateur binaire est donc le résultat d'un test statistique pour un jeu de paramètres et méta-paramètres donnés. Cependant, le travail mené a permis de se rendre compte de la difficulté à définir précisément les paramètres et méta-paramètres des tests, et

la solution proposée dans cette thèse est d'utiliser les différents tests en faisant varier tous les paramètres et méta-paramètres de façon à engendrer un très grand nombre d'indicateurs binaires.

Une méthode de sélection de variables est alors utilisée pour ne garder que les indicateurs pertinents. Des méthodes de type filtre ont d'abord été proposées avant de montrer que dans le cas de l'utilisation d'un classifieur bayésien naïf, une méthode de type wrapper avec une sélection des indicateurs *forward* suivie d'une *backward* ou inversement, obtenait de meilleurs résultats que celles de type filtre. De plus, il a été constaté que dans ce cas il était plus intéressant d'utiliser un critère de sélection basé sur la probabilité d'erreur de classification plutôt que sur le critère classique utilisant le taux de mauvaise classification.

## 7.2 Perspectives

Un premier travail à faire est donc de finir de développer l'interface de labellisation, dont le but premier est d'obtenir une base de moteurs labellisés par les experts. Cette interface pourra également les aider à comprendre la méthodologie et permettre d'avoir des tests (et donc des indicateurs) un peu plus en phase avec leurs attentes.

De ces indicateurs, un premier jeu sélectionné pourra être confronté aux experts et aux retours d'expérience. D'autres types de tests sur la demande des experts métiers pourront alors facilement être ajoutés.

Un point important à signaler est que le taux de bonne classification n'est pas forcément la mesure la plus adaptée pour évaluer les performances d'un classifieur dans le cadre du *Health Monitoring* des moteurs d'avions. La nature des moteurs d'avions provoquent des données avec des classes déséquilibrées comme signalé dans Japkowicz et Stephen (2002). Et il a été mis en avant dans cette thèse que les fausses alarmes, c'est-à-dire les cas où le moteur s'avère sain alors qu'il y a une alarme le concernant, sont à éviter au maximum. Ainsi, une perspective serait de calibrer les algorithmes d'apprentissage automatique à partir d'un critère plus adapté que le taux de bonne classification.

Finalement, on rappelle que le *Health Monitoring* des moteurs d'avions est un domaine où les coûts sont à considérer car ils peuvent s'élever rapidement, par exemple dans le cas d'une maintenance non programmée. Il est donc important de prendre en compte ces coûts asymétriques dans l'évaluation des performances de classification.

Bien d'autres perspectives sont possibles pour améliorer la sélection des indicateurs ou leurs agrégations, et c'est l'avantage de la méthodologie proposée : elle est divisée en étapes suffisamment simples à comprendre pour que toutes les étapes puissent être améliorées indépendamment les unes des autres.

# Bibliographie


Basseville, Michèle et Nikiforov, Igor V. *Detection of abrupt changes : theory and application*. Prentice-Hall, Inc., Englewood Cliffs, N.J., USA, 1993. ISBN 0-13-126780-9. ISBN : 0-13-126780-9.

Bellas, Anastasios, Bouveyron, Charles, Cottrell, Marie, et Lacaille, Jérôme. Robust clustering of high-dimensional data. Dans *Proceedings of the 20th European Symposium on Artificial Neural Networks*, 2012.

Bellas, Anastasios, Bouveyron, Charles, Cottrell, Marie, et Lacaille, Jérôme. Model-based clustering of high-dimensional data streams with online mixture of probabilistic pca. *Advances in Data Analysis and Classification*, 7(3), p. 281–300, 2013.

Boullé, Marc. Compression-based averaging of selective naive bayes classifiers. *Journal of Machine Learning Research*, 8(7), 2007.

Breiman, Leo. *Classification and regression trees*. CRC press, 1993.

Breiman, Leo. Random forests. *Machine Learning*, 45(1), p. 5–32, October 2001. ISSN 0885-6125. URL http://dx.doi.org/10.1023/A%3A1010933404324.

Breiman, Leo, Friedman, Jerome H, Olshen, Richard A, et Stone, Charles J. *Classification and regression trees*. Wadsworth Statistics/Probability. Chapman and Hall/CRC, 1984. ISBN 978-0412048418. ISBN : 978-0-412-04841-8.

Caruana, Rich et Niculescu-Mizil, Alexandru. An empirical comparison of supervised learning algorithms. Dans *Proceedings of the 23rd international conference on Machine learning*, pages 161–168. ACM, 2006.

Cerf, Raphaël. Une théorie asymptotique des algorithmes génétiques. 1994.

Chandola, Varun, Banerjee, Arindam, et Kumar, Vipin. Anomaly detection : A survey. *ACM Computing Surveys*, 41(3), p. 1–58, Juillet 2009. ISSN 0360-0300. URL http://doi.acm.org/10.1145/1541880.1541882.

Côme, Etienne, Cottrell, Marie, Verleysen, Michel, et Lacaille, Jérôme. Aircraft engine health monitoring using self-organizing maps. Dans *Advances in Data Mining. Applications and Theoretical Aspects*, Perner, Petra (éditeur), volume 6171 de *Lecture Notes in Computer Science*, pages 405–417. Springer Berlin Heidelberg, 2010. URL http://dx.doi.org/10.1007/978-3-642-14400-4_31. ISBN : 978-3-642-14399-1.

Demirci, Seref, Hajiyev, Chingiz, et Schwenke, Andreas. Fuzzy logic-based automated engine health monitoring for commercial aircraft. *Aircraft Engineering and Aerospace Technology*, 80(5), p. 516–525, 2008.

Dietterich, Thomas G. Ensemble methods in machine learning. Dans *Multiple classifier systems*, pages 1–15. Springer, 2000.

Fernández-Delgado, Manuel, Cernadas, Eva, Barro, Senén, et Amorim, Dinani. Do we need hundreds of classifiers to solve real world classification problems ? *Journal of Machine Learning Research*, 15, p. 3133–3181, 2014.

Flandrois, Xavier, Lacaille, Jérome, Masse, Jean-Rémi, et Ausloos, Alexandre. Expertise transfer and automatic failure classification for the engine start capability system. Dans *Proceedings of the 2009 AIAA Infotech@Aerospace (I@A) Conference*, Seattle, WA, April 2009. AIAA (American Institute of Aeronautics and Astronautics). ISBN 978-1-60086-979-2.

Fleuret, François. Fast binary feature selection with conditional mutual information. *Journal of Machine Learning Research*, 5, p. 1531–1555, November 2004.

Frénay, Benoît, Doquire, Gauthier, et Verleysen, Michel. On the potential inadequacy of mutual information for feature selection. Dans *Proceedings of ESANN*, volume 2012, 2012.

Frénay, Benoît, Doquire, Gauthier, et Verleysen, Michel. Is mutual information adequate for feature selection in regression ? *Neural Networks*, 48, p. 1–7, 2013a.

Frénay, Benoît, Doquire, Gauthier, et Verleysen, Michel. Theoretical and empirical study on the potential inadequacy of mutual information for feature selection in classification. *Neurocomputing*, 2013b.

Guyon, Isabelle et Elisseeff, André. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3, p. 1157–1182, March 2003.

Guyon, Isabelle, Gunn, Steve, Nikravesh, Masoud, et Zadeh, L. Feature extraction : Foundations and applications. 2006.

Hastie, Trevor, Tibshirani, Robert, Friedman, Jerome, Hastie, T, Friedman, J, et Tibshirani, R. *The elements of statistical learning*, volume 2. Springer, 2009. ISBN 978-0387848570.

Hegedus, Jozsef, Miche, Yoan, Ilin, Alexander, et Lendasse, Amaury. Methodology for behavioral-based malware analysis and detection using random projections and k-nearest neighbors classifiers. Dans *Proceedings of the Seventh International Conference on Computational Intelligence and Security (CIS 2011)*, pages 1016–1023, Hainan, China, December 2011. IEEE Computer Society. ISBN 978-1-4577-2008-6. ISBN : 978-1-4577-2008-6.

Hmad, Ouadie, Massé, Jean-Rémi, Grall-Maës, Édith, Beauseroy, Pierre, et Mathevet, Agnès. Maturation of detection functions by performances benchmark. application to a phm algorithm. *Chemical Engineering*, 33, 2013.

Japkowicz, Nathalie et Stephen, Shaju. The class imbalance problem : A systematic study. *Intelligent data analysis*, 6(5), p. 429–449, October 2002.

Keim, Daniel A, Kohlhammer, Jörn, Ellis, Geoffrey, et Mansmann, Florian. *Mastering The Information Age-Solving Problems with Visual Analytics*. Florian Mansmann, 2010.

Klein, Renata et Issacharoff, Moshe. Model based approach for identification of gears and bearings failure modes. 2009.

Koller, Daphne et Friedman, Nir. *Probabilistic graphical models : principles and techniques*. The MIT Press, July 2009. ISBN 978-0262013192. ISBN : 978-0-262-01319-2.

Kudo, Mineichi et Sklansky, Jack. Comparison of algorithms that select features for pattern classifiers. *Pattern recognition*, 33(1), p. 25–41, 2000.

Kuncheva, Ludmila I et Whitaker, Christopher J. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine learning*, 51(2), p. 181–207, 2003.

Lacaille, Jérôme et Côme, Etienne. Visual mining and statistics for a turbofan engine fleet. Dans *Proceedings of the IEEE Aerospace Conference*, pages 1–8. IEEE, 2011.

Lacaille, Jérôme, Côme, Etienne, et al. Sudden change detection in turbofan engine behavior. Dans *Proceedings of the The Eighth International Conference on Condition Monitoring and Machinery Failure Prevention Technologies*, pages 542–548, 2011.

Lung-Yut-Fong, Alexandre. *Détection de ruptures pour les signaux multidimensionnels. Application à la détection d'anomalies dans les réseaux.* PhD thesis, Télécom ParisTech, 2011.

Massé, Jean-Rémi, Humeau, Aurore, Lalonde, Pierre, et Alimardani, Armand. Placement of alert thresholds on abnormality scores. *PHM Society*, 2014.

Murphy, Kevin P. *Machine learning : a probabilistic perspective*. MIT press, 2012. ISBN 978-0262018029.

Oakland, John S. *Statistical process control*. Routledge, 2008.

Peng, Hanchuan, Long, Fuhui, et Ding, Chris. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8), p. 1226–1238, August 2005.

Rabenoro, T et Lacaille, J. "procédé d'estimation sur une courbe d'un point pertinent pour la détection d'anomalie d'un moteur et système de traitement de données pour sa mise en oeuvre", 23 2013a. Number FR 13 57252.

Rabenoro, Tsirizo et Lacaille, Jérome. Instants extraction for aircraft engine monitoring. Dans *Proceedings of the AIAA Infotech@Aerospace (I@A) Conference*, Boston, MA, August 2013b. AIAA (American Institute of Aeronautics and Astronautics). ISBN 978-1-60086-979-2.

Rabenoro, Tsirizo, Lacaille, Jérôme, Cottrell, Marie, et Rossi, Fabrice. Anomaly detection based on aggregation of indicators. Dans *Proceedings of 23rd annual Belgian-Dutch Conference on Machine Learning (Benelearn 2014)*, Frénay, Benoît, Verleysen, Michel, et Dupont, Pierre (éditeurs), pages 64–71, Brussels (Belgium), 6 2014a.

Rabenoro, Tsirizo, Lacaille, Jérôme, Cottrell, Marie, et Rossi, Fabrice. Anomaly detection based on indicators aggregation. Dans *Proceedings of the International Joint Conference on Neural Networks (IJCNN 2014)*, pages 2548–2555, Beijing (China), 7 2014b. IEEE. ISBN 978-1-4799-6627-1.

Rabenoro, Tsirizo, Lacaille, Jérôme, Cottrell, Marie, et Rossi, Fabrice. A methodology for the diagnostic of aircraft engine based on indicators aggregation. Dans *Advances in Data Mining. Applications and Theoretical Aspects (Proceedings of the 14th Industrial Conference, ICDM 2014)*, Perner, Petra (éditeur), volume 8557 de *Lecture Notes in Computer Science*, pages 144–158, St. Petersburg (Russia), 7 2014c. Springer International Publishing. ISBN 978-3-319-08975-1.

Rabenoro, Tsirizo, Lacaille, Jérôme, Cottrell, Marie, et Rossi, Fabrice. Search strategies for binary feature selection for a naive bayes classifier. Dans *Proceedings of the XXIIIth European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN 2015)*, Bruges, Belgique, 4 2015.

Ratanamahatana, Chotirat Ann et Gunopulos, Dimitrios. Feature selection for the naive bayesian classifier using decision trees. *Applied Artificial Intelligence*, 17(5-6), p. 475–487, 2003.

Ricordeau, Julien et Lacaille, Jérôme. Application of random forests to engine health monitoring. ICAS, 2010.

Ruta, Dymitr et Gabrys, Bogdan. Classifier selection for majority voting. *Information fusion*, 6(1), p. 63–81, 2005.

Siedlecki, Wojciech et Sklansky, Jack. A note on genetic algorithms for large-scale feature selection. *Pattern recognition letters*, 10(5), p. 335–347, 1989.

Sohn, Hoon, Czarnecki, Jerry A, et Farrar, Charles R. Structural health monitoring using statistical process control. *Journal of Structural Engineering*, 126(11), p. 1356–1363, 2000.

Tsitsiklis, John N et al. Decentralized detection. *Advances in Statistical Signal Processing*, 2, p. 297–344, 1993.

Van der Vaart, Aad W. *Asymptotic statistics*, volume 3. Cambridge university press, 2000.

Vellido, Alfredo, Martín, José D, Rossi, Fabrice, et Lisboa, Paulo JG. Seeing is believing : The importance of visualization in real-world machine learning applications. Dans *ESANN*, volume 11, pages 219–226, 2011.

Wilcoxon, Frank. Individual comparisons by ranking methods. *Biometrics bulletin*, pages 80–83, 1945.

Yang, Jihoon et Honavar, Vasant. Feature subset selection using a genetic algorithm. Dans *Feature extraction, construction and selection*, pages 117–136. Springer, 1998.