Détection de ruptures pour les signaux multidimensionnels. Application à la détection d'anomalies dans les réseaux.

Alexandre Lung-Yut-Fong

ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY

TELECOM ParisTech

2012-ENST-00

EDITE - ED 130

Doctorat ParisTech

T H È S E

pour obtenir le grade de docteur délivré par

TELECOM ParisTech

Spécialité « Signal et Images »

présentée et soutenue publiquement par

Alexandre LUNG-YUT-FONG

le 6 décembre 2011

Détection robuste de rupture pour les signaux multidimensionnels.

Application à la détection d'anomalies dans les réseaux

Directeur de thèse : **Olivier CAPPÉ**
Co-encadrement de la thèse : **Céline LÉVY-LEDUC**

Jury

Mme Michèle BASSEVILLE, Directeur de recherche, UMR CNRS 6074, IRISA	Rapporteur
M. Fabrice ROSSI, Professeur, Université Paris 1 Panthéon-Sorbonne	Rapporteur
M. Igor NIKIFOROV, Professeur, UMR CNRS 6279, Université de Technologie de Troyes	Président
M. Dario ROSSI, HDR, Maître de Conférences, TELECOM ParisTech	Examinateur
M. Jean-Philippe VERT, Professeur, INSERM, Mines ParisTech, Insitut Curie	Examinateur

TELECOM ParisTech
école de l'Institut Télécom - membre de ParisTech

T H È S E

Remerciements

J'ai un peu l'impression qu'il faut que j'apporte grand soin à ces prochaines lignes, qui je n'en doute pas, seront parmi les plus lues de ce manuscrit ;-)

Je voudrais tout d'abord remercier les rapporteurs de cette thèse, les professeurs Nicole Basseville et Fabrice Rossi qui ont eu la patience et la gentillesse de relire ce manuscrit, ainsi que les membres du jury, Igor Nikiforov, qui a accepté de le présider, Dario Rossi et Jean-Philippe Vert, dont j'ai beaucoup apprécié les discussions.

Ce travail n'aurait jamais été possible sans l'aide de mes directeurs, qui m'ont encadré, guidé pendant trois voire quatre années. Malgré ses nombreuses responsabilités, j'ai pu bénéficier de l'esprit critique, de l'expérience et de la vision d'Olivier, qui à plusieurs reprises, n'a pas hésité à rester disponible jusqu'à des heures avancées de la nuit pour que l'on puisse venir à bout de la rédaction d'articles. Céline a quant à elle toujours eu porte ouverte, je ne peux compter les nombreuses après-midi pendant lesquelles j'ai pu bénéficier de sa très grande patience et de sa pédagogie. Son dévouement envers ses doctorants est tel que je reçus des messages de sa part alors qu'elle venait à peine d'accueillir sa première petite fille ! Céline et Olivier m'ont toujours soutenu, même pendant les périodes incertaines, et je ne les remercierai jamais assez d'avoir lu et relu extensivement mon manuscrit dans la dernière ligne droite de sa rédaction.

Merci à toute l'équipe STA de m'avoir accueilli, ainsi qu'à mes collègues, ou plutôt devrais-je dire camarades, doctorants et post-doctorants qui ont beaucoup contribué à la bonne ambiance de travail au sein du labo. Lorsque l'on est en stage pour quelque mois, on s'interroge sur l'éventualité de poursuivre dans la recherche, en thèse ; leur bonne compagnie et bonne humeur ont mis un fort poids dans la balance. À Sarah, Nataliya, Tabea, Julien, Zaïd, Steffen, Olaf, Sylvain, Émilie, Charanpal, Onur, Olivier, Marc, Romaric, Kévin, Laurent et mes colocataires de la DA320 Marine, Anne-Laure, Sylvain, Émilie, Amandine, Jian Feng, Joffrey, un grand merci.

J'ai aussi pu trouver un soutien indéfectible en Louise et Jean-Édouard, sans qui ma vie parisienne aurait été beaucoup plus solitaire. Ces nombreux moments en votre compagnie furent nécessaires et précieux.

Mon parcours ainsi que celui de Chloé ont quasiment été identiques ; c'est

avec grande une stupéfaction que j'appris un beau jour qu'elle s'intéressait à des domaines un peu similaires aux miens ! Elle a ainsi pour moi ouvert la voie dans le monde de la recherche en apprentissage statistique et son témoignage fut déterminant pour que je poursuive dans ce même domaine. Malgré la distance qui nous séparait géographiquement, c'est une personne sur laquelle j'ai toujours pu compter, et une formidable amie.

Je ne peux raisonnablement pas quantifier les très nombreuses discussions avec Marie, certes chronophages, mais néanmoins indispensables. Nous avons partagé musique, blagues carambar, soucis respectifs et encouragements. Si j'ai pu garder un certain équilibre, c'est en grande partie grâce à toi, et je suis aujourd'hui très fier de notre profonde amitié.

Je remercie et embrasse enfin mon frère, Dominique, et mes parents Philippe et Caroline, qui m'ont toujours soutenu, à 10000 km de là, depuis l'Île de la Réunion, mais néanmoins si proches.

Ces travaux ont été financés par le programme « Futur et Ruptures » de l'Institut Télécom par le biais de la Fondation Télécom. Ils se sont quasiment exclusivement appuyés sur des logiciels Libres.

Résumé

L'objectif de cette thèse est de proposer des méthodes non-paramétriques de détection rétrospective de ruptures. L'application principale de cette étude est la détection d'attaques dans les réseaux informatiques à partir de données recueillies par plusieurs sondes disséminées dans le réseau.

Nous proposons dans un premier temps une méthode en trois étapes de détection décentralisée d'anomalies faisant coopérer des sondes n'ayant accès qu'à une partie du trafic réseau. Un des avantages de cette approche est la possibilité de traiter un flux massif de données, ce qui est permis par une étape de filtrage par records. Un traitement local est effectué dans chaque sonde, et une synthèse est réalisée dans un centre de fusion. La détection est effectuée à l'aide d'un test de rang qui est inspiré par le test de rang de Wilcoxon et étendu aux données censurées.

Dans une seconde partie, nous proposons d'exploiter les relations de dépendance entre les données recueillies par les différents capteurs afin d'améliorer les performances de détection. Nous proposons ainsi une méthode non-paramétrique de détection d'une ou plusieurs ruptures dans un signal multidimensionnel. Cette méthode s'appuie sur un test d'homogénéité utilisant un test de rang multivarié. Nous décrivons les propriétés asymptotiques de ce test ainsi que ses performances sur divers jeux de données (bio-informatiques, économétriques ou réseau). La méthode proposée obtient de très bons résultats, en particulier lorsque la distribution des données est atypique (par exemple en présence de valeurs aberrantes).

Abstract

The aim of this work is to propose non-parametric change-point detection methods. The main application of such methods is the use of data recorded by a collection of network sensors to detect malevolent attacks.

The first contribution of the thesis work is a decentralized anomaly detector. Each network sensor applies a rank-based change-point detection test, and the final decision is taken by a fusion center which aggregates the information transmitted by the sensors. This method is able to process a huge amount of data, thanks to a clever filtering step.

In the second contribution, we take into account the dependencies between the different sensors to improve the detection performance. Based on homogeneity tests that we have proposed to assess the similarity between different sets of data, the robust detection methods that we have designed are able to find one or more change-point in a multidimensional signal. We thus obtained robust and versatile methods, with strong theoretical properties, to solve a large collection of segmentation problems: network anomaly detection, econometrics, DNA analysis for cancer prognosis... The methods that we proposed are particularly adequate when the characteristics of the analyzed data are unknown.

Table des matières

Chapitre 1

Introduction

Le 5 décembre 2010, le site internet des relations publiques de la société de paiement en ligne PayPal subit plus de 75 interruptions de service pour une durée totale de près de huit heures. En 2009, le volume de transactions qui était géré par PayPal s'élevait à 71 milliards de dollars. Si le service de transfert d'argent de PayPal n'était pas la cible des attaques, ces chiffres donnent un ordre de grandeur des pertes qui auraient été subies si cela avait été le cas. Trois jours plus tard, c'est au tour des sites internet institutionnels des sociétés Visa et Mastercard de subir une interruption de service. Ces sociétés furent victime d'une attaque par déni de service distribué, revendiquée par les *Anonymous*

Les Anonymous sont un groupe d'individus formé par l'intermédiaire d'un forum de discussion à but récréatif ; ce forum a une très grande audience. Pour diverses raisons (qu'elles soient politiques ou non), les Anonymous ont lancé des campagnes d'attaques vers certains sites internet. En décembre 2010, les Anonymous décident de lancer une action coordonnée vers les sites internet de certaines compagnies bancaires : PayPal, Visa et Mastercard. Plusieurs dizaines de milliers d'individus suivent alors les instructions qui publiées sur le forum de discussion : chacun d'entre eux devait télécharger un programme et l'exécuter à une date précise. Ce programme se contentait d'envoyer quelques paquets de données vers les sites internet cibles de l'attaque. Si un seul ordinateur envoyait une grande quantité de données vers un site internet donné, les administrateurs réseau de ce site seraient en mesure de rapidement détecter l'attaque et de prendre les mesures nécessaires pour qu'il n'y ait pas d'interruption de service. L'attaque des Anonymous est en revanche plus difficile à détecter : les milliers d'individus responsables se situent partout dans le monde ; de plus, la quantité de paquets envoyés individuellement étant très faible, elle est indiscernable en comparaison avec la quantité de données qui arrive à tout instant vers un site internet à grande audience. Or c'est la superposition de l'ensemble de ces flux qui est fatale aux machines cibles d'une telle attaque.

L'opération des Anonymous n'est qu'un exemple parmi un grand nombre d'attaques par DDoS (*Distributed Denial of Service*) qui ont touché de nombreuses sociétés ou organisations gouvernementales.

L'équipe STA du département TSI a participé entre 2006 et 2008 au projet ANR-RNRT Oscar (*Overlay network Security Characterization And Recovery*), en collaboration avec France Télécom. L'objet du projet était d'élaborer des méthodes statistiques de détection d'anomalies dans le contexte de la sécurité des réseaux informatiques. Les partenaires du projet disposaient de mesures pouvant être recueillies à différents endroits d'un réseau, mais ces mesures étaient rassemblées en un lieu unique afin d'être analysées. Une contrainte était imposée : un compte-rendu de présence ou non d'une attaque devait être envoyé périodiquement (toutes les minutes) au superviseur de réseau, la quantité de mémoire allouée à cette tâche étant limitée à la quantité de données recueillies pendant cette période. Une méthode de détection de changement *rétrospective*, c'est-à-dire qui analyse un signal constitué par l'ensemble des observations recueillies dans une fenêtre de taille fixée, s'avérait particulièrement adaptée à cette contrainte. Une alternative aurait été d'élaborer une procédure utilisant une méthode *en ligne*, par exemple utilisant le principe du CUSUM (présenté au chapitre 2) qui a été beaucoup utilisé dans la communauté de la sécurité réseau. Lorsque l'on connaît la forme des observations avant et pendant l'anomalie, le CUSUM permet d'alerter au plus tôt l'administrateur réseau. Malheureusement le trafic réseau est très variable : les séries temporelles formées par les observations peuvent prendre de nombreuses formes, c'est-à-dire être distribuées selon des lois différentes ou être à des niveaux très différents (un flux réseau peut être constitué d'à peine quelques paquets par minute comme compter un débit très élevé). Ainsi, utiliser une méthode paramétrique, c'est-à-dire dans laquelle on a besoin de spécifier à l'avance la forme des observations, n'est pas très appropriée. On pourrait adopter une approche guidée par les données et apprendre en direct les caractéristiques du trafic, mais cela introduirait un délai nécessaire à l'apprentissage, la propriété de détection rapide des méthodes en ligne deviendrait alors caduque.

La contrainte mémoire évoquée ci-dessus révèle un problème inhérent à l'analyse *centralisée* des données : la quantité de données qui transite sur un réseau est tellement importante que l'analyse en devient difficile au niveau de la mémoire utilisée et du temps de calcul. Cette thèse propose des méthodes *décentralisées* de détection de ruptures. Dans ce cadre, les sondes, qui dans les méthodes centralisées se contentaient de recueillir les mesures et de les transmettre à une entité qu'on appelle « collecteur central », traitent localement les informations recueillies avant de les transmettre au collecteur central qui prend la décision finale. Les avantages de ces approches sont doubles : d'une part, en déléguant certains calculs aux entités présentes au sein du réseau, on diminue la quantité de données traitées par

un seul calculateur et transmises entre les sondes et le collecteur central ; d'autre part, on peut envisager d'exploiter des informations liées à la localisation des sondes (par exemple des corrélations spatiales ou encore la structure du réseau) afin d'améliorer les performances de détection.

Nous proposons dans ce manuscrit des méthodes de détection de changements rétrospectifs et non-paramétriques. La première est conçue pour l'application de détection d'anomalies réseau dans un cadre décentralisé. La seconde exploite quant à elle les informations de dépendance spatiale entre les différentes mesures recueillies. Cette dernière est en fait complètement générale et peut s'appliquer à des données provenant de diverses applications : analyse de données économétriques, recherche d'éléments caractéristiques de maladies dans le génome, etc.

Ce document est divisé en deux parties. La partie I est consacrée à la détection décentralisée de ruptures dans les réseaux de données. Dans le chapitre 2 nous introduisons le problème en décrivant le mécanisme d'une attaque par déni de service, qui est caractérisée par une augmentation soudaine du nombre de paquets transmis à la machine attaquée. Nous décrivons quelques méthodes qui ont été proposées dans la littérature pour détecter des anomalies réseau. En particulier, nous décrivons dans le chapitre 2 la méthode du TopRank, qui a été développé dans le cadre du projet Oscar mentionné précédemment. La méthode du TopRank est une méthode de détection centralisée. Aussi, nous avons développé un moyen de décentraliser le TopRank. C'est la première contribution de cette thèse, que nous décrivons dans le chapitre 3. Nous proposons ainsi la méthode *DTopRank*, qui est mise en œuvre dans les sondes locales qui recueillent les données puis au niveau du collecteur central. Les sondes appliquent la méthode du TopRank, c'est-à-dire filtrent les données et appliquent un test de détection de ruptures puis transmettent les données les plus pertinentes (qui correspondent aux entités potentiellement impliquées dans une attaque). Le collecteur central effectue ensuite une étape d'agrégation avant d'appliquer à nouveau un test de détection (qui est une extension de celui du TopRank aux données doublement censurées) afin de déclarer la présence ou non d'une anomalie. Dans cette partie, nous établissons le comportement limite de la statistique de test proposée. Cela nous permet de fixer en pratique un seuil de détection en fonction d'un taux de fausses alarmes désiré.

Dans la partie II, nous proposons des méthodes robustes pour tester l'homogénéité d'échantillons ainsi que pour détecter et estimer la position des changements. Elles s'appliquent à des données multi-dimensionnelles. L'idée de départ était, dans l'application réseau, de prendre les séries temporelles collectées par chacune des sondes comme une dimension d'un signal multivarié. On s'éloigne cependant dans cette partie du contexte de départ de la sécurité réseau pour élaborer des méthodes plus générales. Nous commençons dans le chapitre 4 par dresser un état de l'art des procédures, en majorité non-paramétriques, qui per-

mettent de tester l'homogénéité de deux ou plusieurs échantillons multivariés. La seconde contribution principale de cette thèse est ensuite présentée dans les chapitres 5 et 6. Nous fournissons d'abord, dans le chapitre 5, des extensions au cas multi-dimensionnel de tests classiques utilisant des statistiques de rang, les tests de Mann-Whitney/Wilcoxon et de Kruskal-Wallis. Puis, dans le chapitre 6, les tests d'homogénéité proposés sont réutilisés pour élaborer un test de détection d'un changement ainsi qu'une méthode d'estimation de plusieurs changements. Sur quelques exemples de simulations, nous mettons en évidence les propriétés de robustesse des méthodes proposées. Enfin, dans le chapitre 7, nous évaluons ces méthodes sur des jeux de données provenant de diverses applications. Cela nous permet de montrer leurs forces et leurs faiblesses et de discuter de leurs mise en œuvre pratique.

En annexe A, nous présentons un travail effectué en collaboration avec d'autres membres de l'équipe STA sur un sujet distinct du thème principal de cette thèse. Il s'agit d'une application de la méthode d'estimation de changements multiples que nous avons utilisée pour segmenter des signaux de capteurs de mouvements.

Première partie

Détection d'anomalies réseau

Chapitre 2

La détection de changement dans les réseaux de données

L'objectif de ce chapitre est de présenter le problème de la détection d'anomalies dans les réseaux de données et de proposer des outils permettant de se prémunir de ces anomalies. La plupart du temps, elles sont l'œuvre d'une intention malveillante ; on parle alors de systèmes de détection d'intrusions (*IDS*) pour ces dispositifs de veille. De nombreuses méthodes existent dans la littérature pour la détection d'anomalies. On peut les classer en deux catégories : d'une part les approches utilisant des signatures et d'autre part les méthodes statistiques. Les premières opèrent en comparant certaines caractéristiques de l'activité réseau à une base de données existante de motifs d'attaques connues appelées signatures. Des logiciels utilisant des méthodes à signatures, tels que Snort (Roesch *et al.*, 1999) ou Bro (Paxson, 1999), sont disponibles dans le commerce ou la communauté du logiciel libre. L'inconvénient principal de ces systèmes à signatures est qu'ils ne peuvent détecter que les anomalies déjà identifiées ; bien sûr faut-il aussi constamment mettre à jour cette base de données de signatures, processus qui peut prendre du temps après l'apparition de nouvelles méthodes d'attaques, ce qui rend le système vulnérable aux nouvelles attaques. À l'inverse, les méthodes statistiques s'affranchissent de ces limitations et n'utilisent que les données et leurs caractéristiques pour prendre les décisions de détection. Elles peuvent donc potentiellement détecter n'importe quelle attaque inconnue, au prix toutefois d'un nombre de fausses alarmes plus élevé. Une première classe de méthodes statistiques reposent sur le principe qu'il existe un comportement « normal » des données et qu'une attaque ou anomalie constitue une déviation par rapport à ce comportement. Celui-ci peut être modélisé a priori ou alors appris à partir des données. D'autres méthodes n'ont au contraire besoin d'aucun a priori sur la forme des données et agissent de manière totalement non supervisée. Ce sont ces méthodes statistiques qui font l'objet de cette thèse.

Ces derniers systèmes de détections reposent sur le fait que les anomalies dans le trafic réseau se traduisent par un changement abrupt dans certaines caractéristiques du réseau. On peut ainsi utiliser un test sur les séries temporelles correspondant à chacun des flux rencontrés. Cette manière de procéder permet d'identifier quelles sont les entités qui sont impliquées dans l'anomalie.

En plus du problème en lui-même de la détection d'intrusion, un défi supplémentaire vient s'ajouter : les réseaux font face à une croissance exponentielle de leur trafic. Il est ainsi de plus en plus difficile, notamment en temps de calcul, d'analyser chacun des flux individuels. Des méthodes de réduction de la taille des données deviennent donc indispensables.

Dans la section 2.1, nous décrivons tout d'abord un exemple d'anomalie à détecter qui est due à une attaque de la part d'un individu malveillant. Nous présentons ensuite deux méthodes de détection d'anomalies qui sont mises en œuvre dans la communauté de la sécurité réseau. La première (2.2.1), le CUSUM, est un test séquentiel qui s'applique sur une série temporelle pour détecter un changement. La seconde (2.2.2) est une méthode de détection d'anomalies dans les flux réseau à partir de mesures de trafic global enregistrés par plusieurs capteurs contrôlés par l'administrateur réseau. Cette méthode, qui utilise la technique de l'analyse en composantes principales, effectue une décomposition du trafic en deux sous-espaces, le premier engendré par les r premières composantes principales et le second par les composantes résiduelles. Dans la section 2.3, on ajoute la problématique d'*identification* des entités impliquées dans une anomalie, problématique qui est fortement liée au volume de données analysées ; cette section est ainsi consacrée à deux méthodes permettant d'une part de réduire la dimension des données et d'autre part de résoudre la problématique d'identification. On introduit ainsi les *sketches* dans le paragraphe 2.3.1 puis la méthode du *TopRank* en 2.3.2 qui allie un test de détection rétrospectif et une méthode de filtrage. Le TopRank est au cœur de la méthode de détection décentralisée décrite dans le chapitre 3.

2.1 Anomalies réseau : exemple de l'attaque par déni de service

Un système de détection d'anomalies peut faire face à de nombreux types d'attaques. Nous nous intéressons, dans un premier temps, aux attaques par déni de service.

L'attaque par déni de service par engorgement de paquets SYN (*SYN flooding*) exploite les mécanismes utilisés par le protocole TCP. Le protocole TCP est un protocole de transport de paquets de données permettant d'obtenir une liaison de données fiable entre deux machines. Lorsque deux machines, qu'on appellera *Alice* et *Bob* dans cet exemple, veulent effectuer un échange de données à l'aide

du protocole TCP, elles doivent établir une connexion. Celle-ci s'effectue en trois étapes, illustrées en figure 2.1 appelées « poignée de main en trois temps » (ou *three-way handshake*). Schématiquement, le processus se déroule ainsi :

1. Alice envoie un paquet de type SYN à Bob contenant (entre autres) un numéro de séquence aléatoire ;
2. Bob reçoit ce paquet, enregistre en mémoire une trace de cette connexion en la marquant comme étant « semi-ouverte » et envoie un paquet SYN-ACK d'acquittement à Alice ;
3. Alice reçoit ce paquet SYN-ACK et émet vers Bob un paquet d'acquittement ACK ; les deux hôtes ont ainsi chacun reçu un acquittement de la part de l'autre machine, la communication est établie.

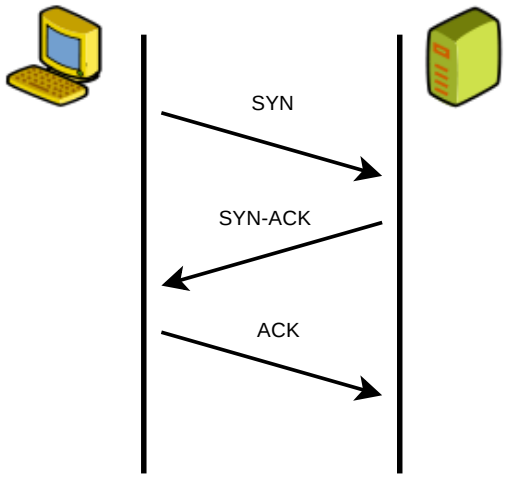


FIGURE 2.1 – Établissement d'une connexion TCP en trois étapes

Une attaque de type *SYN flooding* a pour but d'interrompre le fonctionnement normal d'une machine, celle de Bob, dans ce cas. Une attaquante, qu'on appellera Sasha, va envoyer un grand nombre de paquets SYN contenant des numéros de séquence différents à Bob (voir figure 2.2). Bob, suivant la procédure habituelle, enregistre autant de connexions semi-ouvertes que de paquets SYN reçus et envoie les paquets SYN-ACK correspondants à Sasha qui en revanche ne renvoie pas de paquet ACK. La capacité mémoire de la machine de Bob est limitée, soit de manière physique, soit en nombre de connexions qui peuvent être ouvertes (celle-ci est fixée, et est de l'ordre du millier dans les systèmes d'exploitations récents) ; ainsi lorsque le nombre de connexions semi-ouvertes excède cette capacité, la machine de Bob ne fonctionne plus correctement et Alice ne peut plus établir de connexion avec Bob.

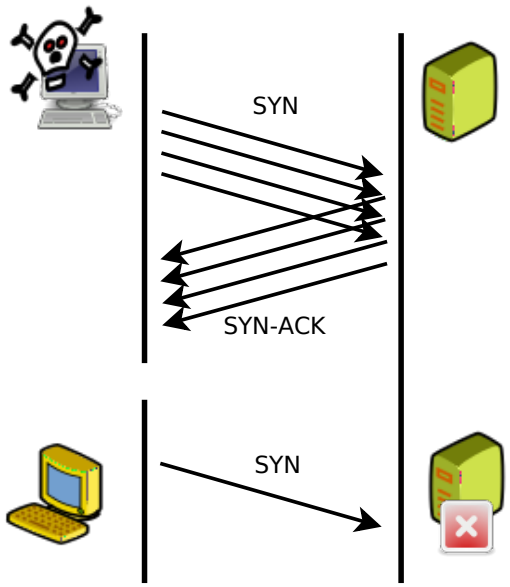


Figure 2.2 – Attaque de type *SYN flooding*

Nous avons décrit le principe de base d'une attaque par *SYN flooding*, il existe de nombreux mécanismes de protection contre ces attaques afin de pouvoir garder un niveau de service acceptable, notamment au niveau du système d'exploitation (temps d'expiration des connexions, limitation du nombre de connexions par hôtes, etc., voir (Eddy, 2007)), mais il existe de nombreuses variantes qui rendent nécessaires des mécanismes de prévention. L'attaquant peut ainsi, aisément, falsifier les entêtes des paquets SYN envoyés en modifiant les adresses IP source ; les vraies machines possédant ces IP n'étant pas accessibles, le processus d'initialisation de connexions ne peut se terminer. L'attaque peut aussi se faire de manière distribuée, et l'on parle de déni de service distribué (ou *DDoS*) : Sasha met à contribution un très grand nombre de machines qu'elle contrôle (soit par collusion avec d'autres attaquants, soit parce qu'elle dispose d'un réseau de machines – a priori innocentes – mais qu'elle a infectées) : chacune d'elle initie de manière parallèle une connexion vers la machine de Bob sans renvoyer de paquets d'acquittement ACK. Le nombre de paquets SYN envoyés par chaque machine peut rester relativement faible, ce qui rend ces attaques individuelles difficilement détectables, mais le grand nombre de machine rend cette attaque distribuée très dangereuse pour la machine de Bob.

L'attaque par engorgement de paquets UDP (*UDP flooding*) est un autre type d'attaque fonctionnant sur un principe similaire à l'attaque précédemment décrite. Elle fait intervenir le protocole UDP et peut potentiellement paralyser un réseau, en plus des machines de celui-ci. En temps normal, lorsqu'une machine s'attend à

recevoir des paquets UDP (par exemple pour un flux multimédia audio ou vidéo), un port UDP d'entrée est ouvert pour y recevoir le flux de données sollicité. Mais lorsqu'une machine reçoit un paquet sur un port non ouvert, celle-ci renvoie un paquet de type ICMP *Destination Unreachable*. Lors d'une attaque par *UDP flooding*, l'attaquant envoie une grande quantité de paquets UDP vers de nombreux ports d'une machine du réseau cible. Ce trafic n'étant pas sollicité, la machine cible renvoie des paquets ICMP. La connexion réseau de la machine devient ainsi indisponible à cause du grand nombre de paquets reçus et envoyés ; par ailleurs, le protocole UDP ne disposant pas de mécanisme de régulation de trafic comme TCP, les liens reliant le réseau au reste du monde peuvent être saturés, dégradant le service pour ses utilisateurs comme pour des machines extérieures voulant accéder à des services de ce réseau.

De nombreux autres types d'attaques ou d'anomalies réseau existent : balayage de ports (*port scanning*) ou d'adresses réseau (*network scanning*), *flash crowds* (Lakhina *et al.*, 2005) ; chacune de ces attaques est caractérisée par une augmentation d'une quantité d'intérêt.

Caractérisation d'une attaque Nous avons jusqu'à présent utilisé le terme d'« anomalie » de manière générique pour désigner un comportement anormal des données. Mais dans la littérature réseau, on peut distinguer deux types d'événements suspects à détecter. Le premier type est l'anomalie proprement dite, dont la détection correspond en fait à de la détection de valeurs aberrantes. Dans ce cadre, on définit un comportement « normal » du trafic, et l'anomalie correspond à quelques points de données qui ne semblent pas distribués de la même manière que le trafic normal. Par exemple, on peut calculer une sphère (en utilisant la distance euclidienne ou de Mahalanobis, par exemple) contenant les points correspondant au comportement normal ; les anomalies se situent alors hors de cette sphère (Shyu *et al.*, 2003). Le deuxième type correspond à une rupture qui est caractérisée par un changement de la distribution des observations après l'instant de rupture. C'est le type d'événements que l'on veut détecter dans cette thèse.

Nous présentons dans ce chapitre quelques méthodes de détection de changements. Au vu de la nature des attaques décrites précédemment, il semble naturel d'étudier les séries temporelles correspondant à un compte de paquets que l'on agrège sur une petite période de temps. Notons ainsi par $N_i(t)$ le nombre de paquets reçus à l'instant t par l'adresse IP i (cet indice i pourra le cas échéant être omis).

2.2 Tests statistiques de données agrégées

2.2.1 Test séquentiel : CUSUM

Un algorithme couramment utilisé (Wang *et al.*, 2002; Siris et Papagalou, 2006; Soule *et al.*, 2005; Tartakovsky *et al.*, 2006) en détection d'anomalies réseau, en particulier pour la détection d'attaques par déni de service, est le *CUSUM* (*CUmulated SUM*, (Basseville et Nikiforov, 1993)), initialement introduit par Page (1954). CUSUM est, contrairement à la plupart des algorithmes présentés dans ce manuscrit, un algorithme *séquentiel*, famille d'algorithmes ayant l'avantage de détecter un changement abrupt avec un délai minimal.

L'algorithme repose sur un test d'hypothèses dans lesquelles on connaît la distribution des données avant et après le changement. Notons le logarithme du rapport de vraisemblance d'une observation $N(t)$ par $s_t = \log p_{\theta_1}(N(t))/p_{\theta_0}(N(t))$, où θ_0 et θ_1 sont les paramètres de la distribution avant et après le changement. s_t prend, en moyenne, des valeurs négatives avant le changement puis des valeurs positives après ; le logarithme du rapport de vraisemblance d'une séquence d'observations $\{N(t)\}_{t=1,\dots,n}$, définie comme la somme

$$S_n = \sum_{i=1}^{n} s_i$$

décroît comme fonction de n avant l'instant de changement et augmente après. La statistique du CUSUM s'écrit

$$g_k = S_k - \min_{1 \le j \le k} S_j$$

et une rupture est détectée si cette statistique dépasse un certain seuil h. La figure 2.3 illustre l'algorithme du CUSUM dans le cas du saut dans la moyenne avec bruit d'observation gaussien. Le choix de la valeur du seuil h doit provenir d'un compromis entre le délai de détection d'une rupture et le taux de fausse alarme.

Le CUSUM ainsi décrit est complètement paramétrique, en particulier la distribution des données après le changement ainsi que les paramètres doivent être spécifiés pour le calcul de la vraisemblance. Pour s'affranchir de cette limitation, plusieurs solutions ont été envisagées. Soule *et al.* (2005) utilisent la méthode du rapport de vraisemblance généralisé et effectuent une estimation du paramètre sur une certaine fenêtre d'observation avant d'appliquer le CUSUM classique ; le désavantage de cette méthode est un sacrifice au niveau du temps de détection. Siris et Papagalou (2006) arrivent à obtenir des données gaussiennes indépendantes – ce qui n'est à leur avis pas le cas pour les comptes de paquets SYN consécutifs – en supprimant les tendances et les phénomènes périodiques à l'aide de méthodes de lissage. Enfin, citons Wang *et al.* (2002) qui utilisent une version non-paramétrique

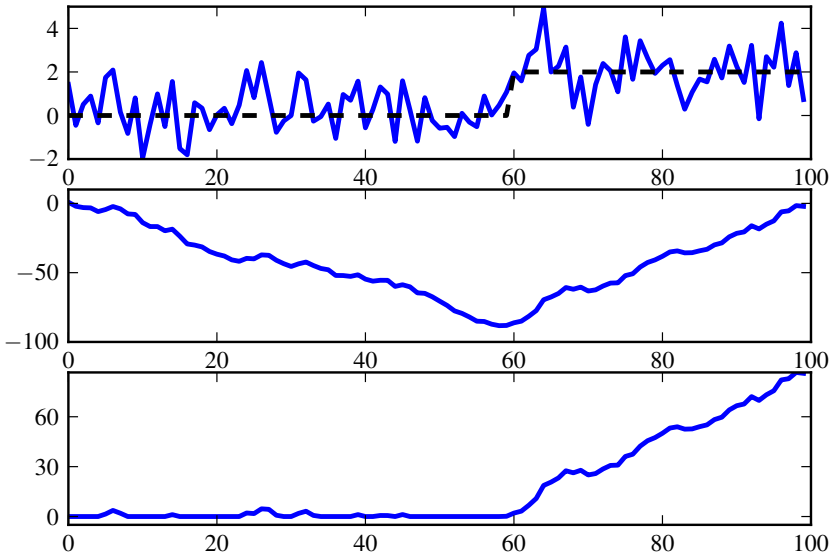


FIGURE 2.3 – Illustration du CUSUM pour des observations gaussiennes. Haut : signal déterministe (pointillés) constant par morceaux, avec un saut dans la moyenne de 2 à la position 60 et signal contaminé avec un bruit standard gaussien en trait plein ; milieu : fonction de log-vraisemblance S_n ; bas : statistique du CUSUM g_k.

de CUSUM introduite par Brodsky et Darkhovsky (1993) ; il n'y a pas d'a priori sur la forme de la distribution des données, mais il faut néanmoins définir une borne supérieure sous laquelle le compte de paquets reste en régime normal ainsi qu'une borne inférieure sur l'intensité des attaques.

2.2.2 Analyse en composantes principales

On s'intéresse maintenant à un cadre quelque peu différent, qui se situe à un niveau d'agrégation supérieur. Nous abordons ainsi l'étude de séries temporelles enregistrées au niveau de liens réseau. Le trafic dans un réseau d'opérateur est constitué de flux Origine–Destination (OD), c'est-à-dire un flux de données transmis d'une entité Origine vers une autre entité Destination. Ces entités peuvent être des adresses IP, des lieux géographiques ou des opérateurs qui échangeraient des données avec d'autres opérateurs. Le chemin emprunté dans le réseau par ces flux OD est déterminé par les tables de routage ; par conséquent, le trafic observé au niveau d'un des liens du réseau est la superposition des flux OD transitant par ce lien. Ce sont les séries temporelles correspondant aux observations au niveau des liens qui sont étudiées dans la suite ; les flux OD ne sont pas directement acces-

sibles (mais partiellement reconstitués par les statistiques obtenues au niveau des liens).

Une méthode courante de réduction de dimension, préalable à une étape de détection d'anomalies, est l'utilisation de l'*Analyse en Composantes Principales* (ACP). L'ACP consiste à déterminer itérativement les axes pour lesquels, si l'on y projette les données, celles-ci auront une variance maximale (pour la première composante principale), la seconde plus grande variance (pour la seconde), etc. La réduction de dimension des données se fait alors en les projetant sur l'espace engendré par un nombre réduit de vecteurs choisis parmi les premières composantes principales. $\mathbf{Y}$ est la matrice centrée (de moyenne nulle) des données, où chacune des ℓ lignes représente un point de données de dimension m (par exemple, dans Lakhina *et al.* (2004), les m dimensions correspondent aux données provenant de m capteurs placés sur les liens du réseau, et l'analyse est faite sur une série temporelle de longueur ℓ). La première composante principale $\mathbf{v}_1$, qui est le vecteur pointant dans la direction de variance maximale de $\mathbf{Y}$, s'écrit

$$\mathbf{v}_1 = \underset{\|\mathbf{v}\|=1}{\operatorname{argmax}} \|\mathbf{Y}\mathbf{v}\| \ .$$

Les composantes principales suivantes se calculent itérativement, on peut écrire la k^{e} composante

$$\mathbf{v}_k = \underset{\|\mathbf{v}\|=1}{\operatorname{argmax}} \left\| \left(\mathbf{Y} - \sum_{i=1}^{k-1} \mathbf{Y}\mathbf{v}_i\mathbf{v}_i' \right) \mathbf{v} \right\| \ .$$

La composante principale $\mathbf{v}_k$ peut aussi être vue comme le vecteur propre associé à la valeur propre λ_K de la matrice de covariance des données $Y'Y$, où les valeurs propres sont telles que $\lambda_1 \geq \cdots \geq \lambda_m \geq 0$.

L'ACP est souvent utilisée comme une méthode de réduction de dimension. Par exemple Shyu *et al.* (2003) analysent des données dont les descripteurs sont diverses caractéristiques de trafic réseau (entêtes du protocole TCP, nombre de paquets échangés, etc.) ; les données sont alors de dimension supérieure à 30. Ils appliquent une méthode de détection d'anomalies sur les composantes principales qui expliquent 50% de la variance totale.

La méthode de Lakhina *et al.* (2004) utilise une décomposition en sous-espaces qui résulte de l'ACP. Cette méthode a pour but de détecter des anomalies (dues à des attaques, défaillances réseau ou autres) au sein d'un ou plusieurs flux Origine–Destination à partir des données agrégées recueillies au niveau des liens réseau. On utilise le fait qu'une anomalie dans un flux OD se propage au niveau de chacun des liens traversés par ce flux. La difficulté réside dans le fait que le trafic d'un flux donné et une anomalie dans ce flux sont complètement noyés dans le trafic total enregistré au niveau des liens qui agrègent chacun un grand nombre de flux OD différents.

Les auteurs appliquent une ACP à la matrice $\mathbf{Y}$ décrite ci-dessus, et distinguent les r premières composantes principales [1] des $m - r$ suivantes. Ils donnent une interprétation de la séparation du trafic en deux sous-espaces : la projection sur le sous-espace engendré par les r premières composantes principales correspond aux variations usuelles et prévisibles du trafic réseau (variations périodiques, notamment quotidiennes) ; la projection sur le sous-espace complémentaire est la composante « anormale » du trafic. C'est sur cette composante qu'est appliqué le schéma de détection d'anomalies : les anomalies présentes dans un flux OD individuel sont visibles sur les projections sur les composantes résiduelles alors qu'elles demeurent noyées par le trafic total quand ce flux est projeté sur les r premières composantes. La statistique de détection d'anomalie est celle de Jackson et Mudholkar (1979), elle est appelée erreur de prédiction quadratique (*Squared Prediction Error* (SPE)) et est définie comme $\|(\mathbf{I} - \mathbf{P}\mathbf{P}')\mathbf{y}\|^2$, $\mathbf{P}$ étant la matrice de dimensions $m \times r$ dont les r colonnes sont les premières composantes principales $(\mathbf{v}_1, \ldots, \mathbf{v}_r)$. Une anomalie est déclarée lorsque le SPE dépasse un certain seuil Q_α qui dépend des valeurs propres $(\lambda_{r+1}, \ldots, \lambda_m)$ associées aux $(m - r)$ composantes résiduelles.

Cette étape de détection est le cas échéant suivie d'une étape d'identification du flux Origine-Destination impliqué dans l'anomalie détectée, une procédure assez lourde nécessitant l'énumération de tous les flux OD possibles et du calcul d'une statistique associée. Si cette procédure est faisable dans le cas où les entités source ou destination agrègent un réseau complet, elle est ne l'est plus si l'on veut identifier finement quelles machines (c'est-à-dire quelles adresses IP particulières) sont impliquées.

2.3 Détection et identification de ruptures

La détection d'anomalies ou de ruptures consiste à appliquer un test ou un algorithme à certaines séries temporelles afin de déterminer la présence d'un événement suspect. Les données à analyser dans les réseaux informatiques sont volumineuses et de grande dimension. Il est ainsi nécessaire de pré-traiter les données afin de réduire le volume à analyser. En agrégeant les séries temporelles correspondant à plusieurs flux de données, on pourrait faciliter l'analyse de l'ensemble du trafic. Cependant on perd dans ce cas la faculté d'identifier quelle entité est concernée par une anomalie.

Nous présentons dans cette section des méthodes qui permettent de conserver la propriété d'identifiabilité tout en étant en mesure de diminuer la quantité de données à analyser.

[1] Les premières composantes sont celles dont la projection contient une déviation supérieure à trois fois l'écart type par rapport à la moyenne.

2.3.1 Réduction de la dimension par agrégation aléatoire (*sketches*)

Les *sketches* sont une technique de représentation résumée de flux de données. Le principe des *sketches* est d'effectuer une agrégation aléatoire d'un grand nombre d'objets vers un ensemble de plus petites dimensions. Dans le cas de notre application réseau, cela permet de réduire le nombre d'objets à examiner de plusieurs dizaines de millions d'adresses IP potentielles à un nombre bien plus petit (par exemple de l'ordre de la centaine dans Lévy-Leduc et Roueff (2009)), ce qui rend possible l'application de méthodes de détection à chacun de ces objets. Les méthodes d'agrégation aléatoire ont par exemple été utilisées pour la détection d'anomalies réseau par Krishnamurthy *et al.* (2003), Li *et al.* (2006) ou Lévy-Leduc et Roueff (2009).

Plus précisément, étant donnée une famille d'objets (par exemple d'adresses IP sources, destination ou paires source–destination), un *sketch* de taille K est construit de la manière suivante. Soit une fonction de hachage $h(\cdot)$ qui à une adresse IP i donnée associe un nombre entier dans $\{1, \dots, K\}$. On agrège alors le compte de paquets de toutes les adresses IP associées à la même valeur de hachage k :

$$X_k(t) = \sum_i \mathbf{1}\left(h(i) = k\right) N_i(t), \quad t = 1, \dots, P$$

sont les séries temporelles étudiées par la suite.

En pratique, on met en œuvre une collection de L *sketches*, c'est-à-dire qu'on utilise une famille de L fonctions de hachage h_ℓ distinctes, $\ell = 1, \dots, L$ pour obtenir $L \times K$ séries temporelles agrégées $X_{\ell,k}$ sur lesquelles on applique un test de détection d'anomalie (étape 1 de la figure 2.4).

En présence d'une anomalie dans la série temporelle concernant une adresse IP i (inconnue), on s'attend à ce que le test de détection soit positif pour les L séries temporelles agrégées $(X_{\ell,h_\ell(i)}(t))_{1\le t\le P}$ (le test appliqué à l'une de ces L séries peut toutefois être négatif puisqu'elle est l'agrégation de séries temporelles correspondant à plusieurs adresses IP). L'adresse IP concernée est alors identifiée en faisant l'intersection des ensembles des adresses ayant obtenu les mêmes valeurs de hachage que i (étape 2 de la figure 2.4). Plus précisément, en désignant par $\mathcal{L}^+_{\ell,k}$ l'ensemble des adresses IP hachées dans la case (ℓ, k) si le test de détection est positif, et l'ensemble vide s'il est négatif, alors l'ensemble des adresses concernées par l'anomalie est

$$\bigcap_{\ell=1}^{L} \bigcup_{k=1}^{K} \mathcal{L}^+_{\ell,k} \,.$$

La famille de fonctions de hachage h_ℓ est choisie afin d'obtenir une faible probabilité de collision (le fait que plusieurs adresses IP soient associées aux mêmes valeurs de sortie des fonctions de hachage, phénomène qui risque d'induire des faux positifs dans le test de détection). Par exemple, la famille de fonctions de

hachage proposée par Thorup et Zhang (2004)[2]

$$h_\ell(x) = 1 + \left(\sum_{j=0}^{3} a_{j,\ell} x^k \mod p \right) \mod K \,,$$

où x est un entier de 32 bits, $p = (2^{61} - 1)$ est un nombre premier de Mersenne et $a_{j,\ell}$ est un entier tiré aléatoirement parmi $\{0, \ldots, p\}$, garantit une probabilité de collision de $1/K^4$ (on dit que c'est une fonction de hachage 4-universelle).

Les valeurs typiques de la taille des *sketches* sont de l'ordre de quelques centaines ($K = 32$ à 1 024 pour Li *et al.* (2006); Krishnamurthy *et al.* (2003)), et de la dizaine pour L.

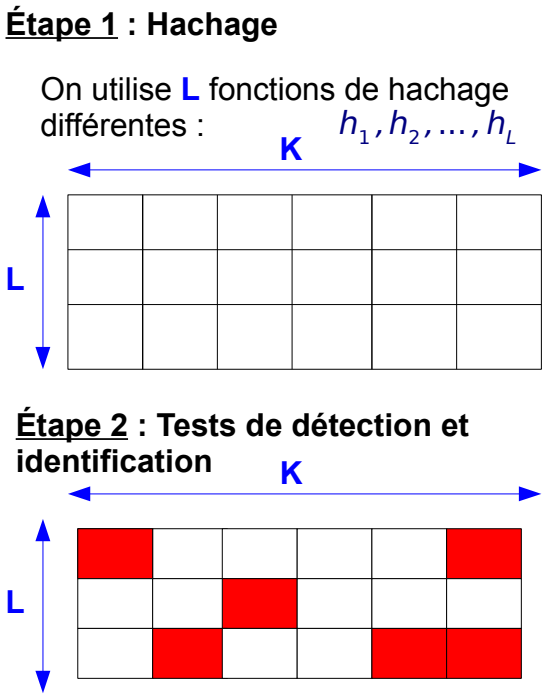


FIGURE 2.4 – Méthode de projection aléatoire (*sketches*). À l'étape 1, on assigne une case de chaque ligne du tableau à chaque adresse IP rencontrée. On agrège les séries temporelles correspondantes. À l'étape 2, on applique le test de détection à la série temporelle agrégée dans chaque case du tableau ; pour chaque ligne, on note quelles IP correspondent aux cases pour lesquelles le test de détection est positif (cases colorées dans l'exemple de la figure) ; on fait l'intersection des ensembles d'IP de chaque ligne pour identifier les IP concernées par les attaques.

[2]Qui ont aussi l'avantage d'être peu coûteuses en temps de calcul.

De nombreux algorithmes de détection ont été utilisés après l'étape de filtrage par les *sketches* : Krishnamurthy *et al.* (2003) utilisent des modèles de prédiction à partir de moyennes glissantes pondérées pour établir le comportement « normal » de la série temporelle ; Li *et al.* (2006) utilisent la méthode de Lakhina *et al.* (2004) que l'on a décrit dans la section 2.2.2 ; Lévy-Leduc et Roueff (2009) comparent, avec la même méthode de détection de changement (celle de TopRank, qui est décrit dans la section suivante), deux méthodes de réduction de dimension : celle utilisant les *sketches* et celle utilisant un filtrage par records qui est décrit dans le paragraphe 2.3.2. Expérimentalement, à taux de réduction de données égaux et en utilisant le même algorithme de détection, la méthode de filtrage par records que nous décrivons ci-dessous utilisée dans *TopRank* produit de meilleures performances de détection que la méthode de projection aléatoire.

2.3.2 Test rétrospectif et réduction de la dimension avec filtrage par records : TopRank

Nous présentons maintenant la méthode du *TopRank*, proposée par Lévy-Leduc et Roueff (2009), qui combine une technique de réduction de dimension par filtrage par records et un test non-paramétrique de détection de ruptures. Contrairement au CUSUM, séquentiel, le TopRank utilise une procédure de détection rétrospective, c'est-à-dire que l'on applique un algorithme sur l'ensemble des données de la fenêtre.

Introduisons tout d'abord un outil qui est utilisé dans le *TopRank* que nous décrivons après.

Modèle de censure aléatoire

Le modèle de censure aléatoire est utilisé lorsqu'une variable aléatoire d'intérêt n'est que partiellement connue. C'est un modèle couramment utilisé dans les essais cliniques ou les enquêtes épidémiologiques.

Soient ainsi $X_1, \ldots, X_n$ des variables aléatoires. Dans les modèles de censure, on considère que l'on ne dispose pas de ces observations mais un certain vecteur multivarié. Soit $C_1, \ldots, C_n$ des variables aléatoires, pour un i donné,

- si l'on observe les variables aléatoires $(\tilde{X}_i, \delta_i)$, où $\tilde{X}_i = \min(X_i, C_i)$ et $\delta_i = \mathbf{1}(X_i \leq C_i)$, lorsque $\delta_i = 0$, on dit que X_i est censuré à droite, on a connaissance d'une borne inférieure de X_i ;
- inversement, si on observe $(\tilde{X}_i, \delta_i)$, où $\tilde{X}_i = \max(X_i, C_i)$ et $\delta_i = \mathbf{1}(X_i \geq C_i)$, lorsque $\delta_i = 0$, X_i est censuré à gauche, on connaît une borne supérieure de X_i ;
- lorsque l'on observe (L_i, R_i) où $-\infty \leq L_i < X_i < R_i \leq \infty$, X_i est alors censuré par intervalle, ou doublement censuré.

Description de la méthode TopRank

Lévy-Leduc et Roueff (2009) proposent une méthode pour la détection de ruptures, dans les cas particuliers où l'on veut détecter des attaques de type déni de service par SYN ou UDP *flooding* ou balayage de ports ou de réseaux. L'objectif, par exemple dans la tâche de détection de DoS est de pouvoir analyser les séries temporelles du nombre de paquets envoyés à tous les hôtes du réseau afin de détecter les machines victimes d'attaques. Le nombre de machines apparaissant dans les traces de données étant très élevé, une diminution de la quantité de données à analyser est nécessaire. La méthode de Lévy-Leduc et Roueff (2009) combine ainsi une étape de filtrage qui permet de diminuer la quantité de données traitées en éliminant celles non relatives à des machines possiblement attaquées – ce processus produit des séries temporelles censurées – et un test de détection de changement qui avait été initialement proposé par Gombay et Liu (2000). Ce test utilise les idées de Gehan (1965) ou Mantel (1967) qui proposent une généralisation du test de Wilcoxon (que l'on présente dans la section 4.4.1) aux données aléatoirement censurées. Soit une fenêtre de P observations. On subdivise cette fenêtre en deux sous fenêtres de taille p_1 et $P - p_1$. Le test de détection de rupture de Gombay et Liu (2000) consiste alors à calculer la statistique permettant de tester l'homogénéité des deux fenêtres pour l'ensemble des valeurs de p_1, puis à en prendre la plus grande valeur. Cette plus grande valeur est alors comparée à un seuil (que l'on détermine à partir de la loi de cette statistique sous l'hypothèse nulle) afin de déterminer la présence d'une rupture dans cette fenêtre d'observations.

La quantité d'intérêt est dans le cas de la détection d'attaques par SYN *flooding* le nombre de paquets de type SYN envoyés à chaque adresse IP i. Considérons ainsi la situation où le trafic de données est analysé dans plusieurs fenêtres d'observations successives d'une certaine durée $P \times \Delta$, où P est un entier indiquant le nombre de subdivisions de la fenêtre d'observation, et Δ la durée de ces subdivisions dans lesquelles on compte le nombre de paquets. On veut prendre une décision concernant la présence d'attaques dirigées vers différentes adresses IP et identifier les adresses en question. On désigne ainsi par $(N_i(t))_{1\leq t\leq P}$ la série temporelle associée au nombre de paquets SYN reçus par l'adresse IP i. La difficulté du problème réside dans le grand nombre d'adresses IP et de séries temporelles à analyser, c'est le point qui justifie l'étape de filtrage du TopRank.

Détaillons ainsi la méthodologie du TopRank qui se déroule en trois étapes.

1. Filtrage par records : Pour chaque indice $t \in \{1, \ldots, P\}$ de la fenêtre d'observation, on garde les adresses IP i des M plus grands $N_i(t)$, que l'on note $i_1(t), \ldots, i_M(t)$ et tels que : $N_{i_1(t)}(t) \geq N_{i_2(t)}(t) \geq \cdots \geq N_{i_M(t)}(t)$. On note $\mathcal{T}(t) = \{i_1(t), \ldots, i_M(t)\}$ ce classement d'adresses IP. Notons que l'on ne garde pour la suite que les éléments de $\mathcal{T}(t)$ ainsi que les valeurs correspondantes $\{N_i(t),\, i \in \mathcal{T}(t),\, t = 1, \ldots, P\}$. On a ainsi créé un tableau de taille $P \times M$ gardant en mémoire

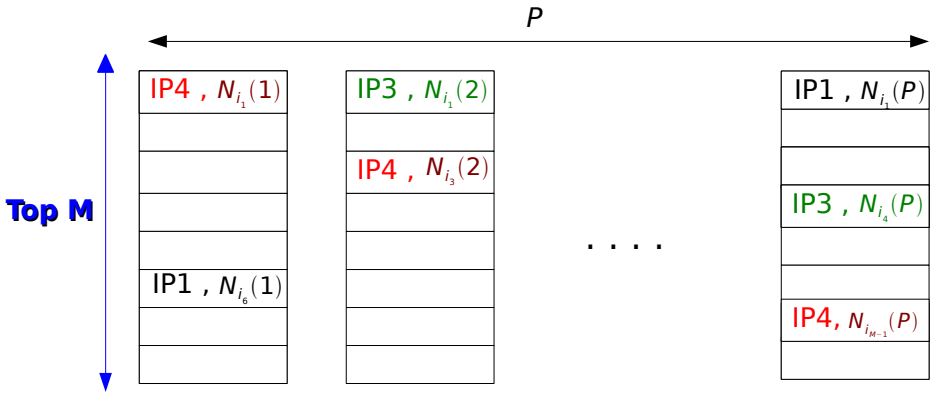


Figure 2.5 – Étape 1 du *TopRank*

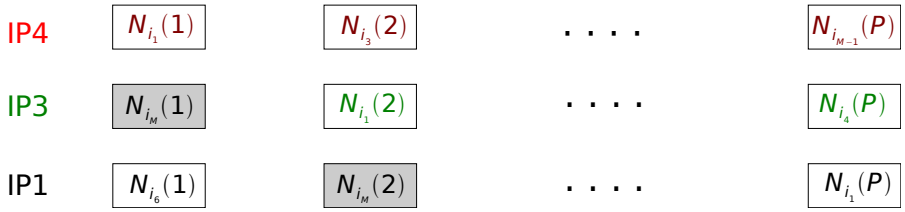


Figure 2.6 – Étape 2 du *TopRank*

les différents classements $\mathcal{T}(t)$ pour $t = 1, \ldots, P$ ainsi que le nombre de paquets associés $N_{i_j}(t)$. Ce tableau et la première étape sont illustrés dans la figure 2.5.

2. Création des séries temporelles censurées : Pour chaque adresse IP i sélectionnée à l'étape précédente ($i \in \bigcup_{t=1}^{P} \mathcal{T}(t)$), on construit la série temporelle $(X_i(t))_{1 \leq t \leq P}$. Cette série est censurée à gauche, puisqu'il se peut qu'à un instant t, i ne soit pas dans l'ensemble $\mathcal{T}(t)$ et que l'on ne dispose donc plus de la valeur $N_i(t)$ correspondante. Dans ce cas, $X_i(t)$ prend alors la valeur $N_{i_M(t)}(t) = \min_{i \in \mathcal{T}_M(t)} N_i(t)$. De manière formelle, la série temporelle censurée $(X_i(t), \delta_i(t))_{1 \leq t \leq P}$ est définie pour tout $t \in \{1, ..., P\}$ par :

$$(X_i(t), \delta_i(t)) = \begin{cases} (N_i(t), 1), & \text{si } i \in \mathcal{T}_M(t) \\ (\min\limits_{j \in \mathcal{T}_M(t)} N_j(t), 0), & \text{sinon.} \end{cases}$$

La valeur de $\delta_i(t)$ indique si la valeur correspondante $X_i(t)$ a été censurée ou non. Observons que par définition, $\delta_i(t) = 1$ implique que $X_i(t) = N_i(t)$ alors que $\delta_i(t) = 0$ implique que $X_i(t) \geq N_i(t)$.

Dans l'exemple de la figure 2.6 qui fait suite à la figure 2.5, pour chaque t, le nombre de paquets reçus par l'adresse `IP4` apparaît toujours dans le top M des adresses recevant le plus de paquets ; la série temporelle correspondante des

$X_{\texttt{IP4}}(t)$ est alors égale à $N_{i_1}(1), N_{i_3}(2), \ldots, N_{i_7}(P)$, la série n'est pas censurée. À l'inverse, pour $i = \texttt{IP1}$, $N_{\texttt{IP1}}(2)$ est plus petit que $N_{i_M}(2)$, la valeur de la dernière ligne. $X_{\texttt{IP1}}(2)$ est alors censurée, et prend pour valeur $N_{i_M}(2)$.

3. Test de détection de rupture : La statistique utilisée à cette étape, proposée par Gombay et Liu (2000), vise à tester les hypothèses suivantes sur les données non-censurées :

- (H_0) : « $(N_i(t))_{1\leq t\leq P}$ sont des variables aléatoires indépendantes et identiquement distribuées »
- (H_1) : « il existe r tel que $(N_i(1), \ldots, N_i(r))$ et $(N_i(r+1), \ldots, N_i(P))$ sont distribuées différemment. »

Soit

$$h(s,t) = \mathbf{1}\left(X_i(s) > X_i(t), \delta_i(s) = 1\right) - \mathbf{1}\left(X_i(s) < X_i(t), \delta_i(t) = 1\right) \ .$$

h est la fonction de score de Gehan (1965) pour comparer deux observations $(X_i(s), \delta_i(s))$ et $(X_i(t), \delta_i(t))$. $h(s,t)$ ne contribue à la statistique définie ci-après uniquement si l'on est sûr de l'ordre relatif des quantités d'intérêt $N_i(s)$ et $N_i(t)$; en particulier lorsque les deux valeurs sont censurées, on n'a pas assez d'informations et la fonction de score n'est pas comptée.

En notant

$$Y_s = \frac{U_s}{\sqrt{\sum_{t=1}^{P} U_t^2}} \ , \quad \text{où} \quad U_s = \sum_{t=1}^{P} h(s,t) \ ,$$

la statistique de test est donnée par

$$W_P = \max_{1\leq t\leq P} \left| \sum_{s=1}^{t} Y_s \right| \ .$$

L'hypothèse nulle (H_0) est alors rejetée pour des valeurs assez grandes de la statistique W_P.

Dans le chapitre 3, nous décrirons une extension de ce test qui s'applique au cas où les données sont *doublement* censurées ainsi qu'un moyen de fixer le seuil de décision.

Chapitre 3

Test de changement pour données censurées dans l'application réseau

3.1 Introduction

La détection de comportements malveillants est une préoccupation importante pour la sécurité des infrastructures de réseaux de données. Dans ce chapitre, on s'intéresse particulièrement au cas des attaques par déni de service distribué que nous avons décrit au chapitre précédent. Dans les réseaux étendus (*Wide Area Networks*), nous avons vu qu'il existait deux grandes problématiques. La première est que la quantité de données transmises sur un réseau et le très grand nombre d'entités impliquées empêchent l'application d'un test de détection sur chacune des séries temporelles rencontrées. D'autre part il y a la problématique de la *décentralisation*. Les données sont souvent recueillies à différents endroits du réseau, et dans les approches centralisées, l'ensemble des données est envoyé pour analyse à un site centralisé, que l'on appelle collecteur central. Les méthodes centralisées ont pour inconvénient de générer un surplus de données, utilisé uniquement pour la supervision et la détection, qui devient non négligeable par rapport à la quantité de données utiles. On cherche donc à élaborer des méthodes dites décentralisées, dans lesquelles les sondes collectant les données mettraient à disposition leurs capacités de calcul. Les sondes traitent ainsi les données avant d'envoyer au collecteur central les données les plus pertinentes, on réduit ainsi le volume de données transmises par rapport aux méthodes centralisées.

Une première approche consiste tout simplement à effectuer un résumé des données, par exemple en effectuant un échantillonage (on n'envoie qu'un paquet sur N). Cette approche a certes l'avantage de limiter les volumes de données échangées entre les sondes et le collecteur, mais un échantillonage excessif réduit grandement les performances des algorithmes de détection au niveau du collecteur.

On peut citer l'approche de Huang *et al.* (2007) qui proposent une méthode dérivée de celle de Lakhina *et al.* (2004) présentée au paragraphe 2.2.2, et qui a pour but de décentraliser le traitement des données dont une partie est effectuée au niveau des collecteurs qui enregistrent les comptes de paquets. Comme dans la version centralisée, l'ACP et le test de détection d'anomalies sont faits au sein d'un collecteur central sur la matrice de données $\mathbf{Y}$. Cependant chacune des colonnes $\mathbf{Y}_i$ de cette matrice est générée au sein des sondes. Une sonde i vérifie en permanence si les données relevées entrent dans un gabarit $[R_i(t) - \delta_i, R_i(t) + \delta_i]$, où $R_i(t)$ est un résumé de $\mathbf{Y}_i$ (moyenne des dernières valeurs de $\mathbf{Y}_i$) et δ_i est un paramètre permettant de contrôler l'erreur introduite par ce processus de décentralisation. Lorsque les données n'entrent plus dans le gabarit, la sonde envoie les informations à jour $\mathbf{Y}_i(t)$ et $R_i(t)$ au collecteur central. Huang *et al.* (2007) proposent une méthode permettant de calculer les δ_i en fonction des erreurs introduites par le filtrage effectué par les sondes. Ils obtiennent une méthode réduisant de 90% la quantité de données échangées entre les sondes et le collecteur central tout en gardant un taux de fausses alarmes à peine affecté de 5%.

La contribution présentée dans ce chapitre est une manière efficace de décentraliser l'algorithme du *TopRank* de Lévy-Leduc et Roueff (2009) et que l'on a présenté au paragraphe 2.3.2. L'algorithme proposé, qu'on appelle *DTopRank* (pour *Decentralised TopRank*), consiste à appliquer localement (au sein d'une sonde) le TopRank et à n'envoyer au collecteur central que les données les plus pertinentes, à savoir les séries temporelles associées aux adresses IP marquées comme étant potentiellement impliquées dans une attaque. Les données envoyées par les différentes sondes sont ensuite agrégées d'une manière spécifique, ce qui nécessite le développement d'un test de rang spécialement adapté aux données doublement censurées. On montre que l'algorithme du *DTopRank* permet d'obtenir des performances équivalentes à celles du TopRank centralisé tout en réduisant la quantité de données échangées.

Dans la section 3.2, nous décrivons l'algorithme du *DTopRank* ; la description proposée de l'étape locale diffère légèrement de celle déjà introduite au paragraphe 2.3.2 : cette version est adaptée à l'extension décentralisée de TopRank ; nous déterminons aussi la distribution limite de la statistique de test proposée sous l'hypothèse nulle, c'est-à-dire en absence d'anomalie. Les performances de l'algorithme proposé sont évaluées sur un jeu de données provenant d'un fournisseur d'accès à internet (section 3.3) puis sur des données synthétiques (section 3.4). Dans les deux cas, DTopRank est comparé au TopRank centralisé ainsi qu'à une méthode de tests multiples utilisant la correction de Bonferroni.

3.2 Description des méthodes proposées

Les données brutes que l'on analyse sont des résumés des flux de paquets qui transitent sur le réseau. Pour chacun des flux, ces résumés contiennent diverses informations, entre autres les adresses IP source et destination du flux, les dates de début et de fin de la communication ainsi que le nombre de paquets échangés. Ce sont des informations qui sont incluses dans le format d'enregistrement Netflow qui est un format standard implémenté par de nombreux constructeurs de matériel réseau.

On note $(N_i(t))_{t\geq 1}$ le nombre de paquets envoyés à la machine d'adresse IP i dans l'intervalle t de longueur Δ de la fenêtre d'observation. La nature de cette quantité dépend du type d'anomalie que l'on veut détecter. Par exemple, pour détecter une attaque d'engorgement par paquets TCP/SYN (*TCP SYN flooding*), $(N_i(t))_{1\leq t\leq P}$ représente la série temporelle du nombre de paquets SYN reçus par l'adresse IP i ; dans le cas d'une attaque d'engorgement UDP (*UDP flooding*), $N_i(t)$ est le nombre de paquets UDP reçus par la machine d'adresse IP i.

L'algorithme *TopRank* centralisé procède à l'analyse des mesures de comptes de paquets globaux. Dans ce chapitre, on considère en revanche un système de surveillance constitué d'un ensemble de K sondes $M_1, \ldots, M_K$; celles-ci mesurent et analysent les séries temporelles observées localement. Un tel traitement décentralisé des informations implique que les paquets envoyés à une adresse IP donnée ne sont pas forcément observés par toutes les sondes ; cela dépend de la position des sondes dans le réseau et de la matrice de routage des paquets. On note ainsi $N_i^{(k)}(t)$ le nombre de paquets envoyés à la machine d'adresse IP i vus par la sonde M_k au temps t. Dans l'approche rétrospective proposée, la détection est réalisée à partir des données observées dans une fenêtre d'observation de longueur $P \times \Delta$ secondes. Le but est de détecter les instants de rupture dans les séries temporelles agrégées $(N_i(t))_{1\leq t\leq P}$ uniquement à partir des séries temporelles locales $\left(N_i^k(t)\right)_{1\leq t\leq P}$ pour $k \in \{1, \ldots, K\}$, tout en minimisant la quantité de données transmises au collecteur central.

3.2.1 Description de la méthode *DTopRank*

L'algorithme *DTopRank* opère à deux niveaux différents : dans chacune des sondes $M_1, \ldots, M_K$, où un traitement local est effectué ; et au niveau du collecteur central où ont lieu une étape d'agrégation de données provenant des sondes puis un test de détection de rupture.

Traitement local

L'étape de traitement local du *DTopRank* peut se subdiviser en quatre sous-étapes qui sont mises en œuvre dans chacune des K sondes ; elles sont décrites ci-après.

Les trois premières sous-étapes sont semblables à l'algorithme du *TopRank* que l'on appliquerait aux suites de comptes locales $\left(N_i^k(t)\right)_{1\leq t\leq P}$. Les deuxième et troisième étapes sont néanmoins légèrement modifiées : on a introduit une valeur de censure à gauche (par valeur inférieure) pour chacune des séries analysées pour pouvoir faire l'agrégation au niveau du collecteur central. Dans cette section, l'exposant $^{(k)}$ qui correspond à l'indice de la sonde locale a été omis afin d'alléger les notations.

1. Filtrage par records : Dans chaque sous-intervalle indexé par $t \in \{1,\dots,P\}$ de longueur Δ secondes de la fenêtre d'observation, on garde les adresses IP i des M plus grands comptes $N_i(t)$, que l'on note $i_1(t),\dots,i_M(t)$ et tels que : $N_{i_1(t)}(t) \geq N_{i_2(t)}(t) \geq \cdots \geq N_{i_M(t)}(t)$. On note $\mathcal{T}(t) = \{i_1(t),\dots,i_M(t)\}$ l'ensemble des adresses IP correspondantes. Notons que l'on ne garde pour la suite que les éléments de $\mathcal{T}(t)$ ainsi que les valeurs correspondantes $\{N_i(t),\, i \in \mathcal{T}(t),\, t = 1,\dots,P\}$.

2. Création des séries temporelles censurées : Pour chaque adresse IP i sélectionnée à l'étape précédente ($i \in \bigcup_{t=1}^{P} \mathcal{T}(t)$), on construit la série temporelle $(X_i(t))_{1\leq t\leq P}$. Cette série est censurée, puisqu'il se peut qu'à un instant t, i ne soit pas dans l'ensemble $\mathcal{T}(t)$ et que l'on ne dispose donc plus de la valeur $N_i(t)$ correspondante. Dans ce cas, $X_i(t)$ prend alors la valeur $N_{i_M(t)}(t) = \min_{i\in\mathcal{T}_M(t)} N_i(t)$. De manière formelle, la série temporelle censurée $(X_i(t),\delta_i(t))_{1\leq t\leq P}$ est définie pour tout $t \in \{1,...,P\}$ par :

$$(X_i(t),\delta_i(t)) = \begin{cases} (N_i(t),1), & \text{si } i \in \mathcal{T}_M(t) \\ (\min\limits_{j\in\mathcal{T}_M(t)} N_j(t),0), & \text{sinon.} \end{cases}$$

La valeur de $\delta_i(t)$ indique si la valeur correspondante $X_i(t)$ a été censurée ou non. Observons que par définition, $\delta_i(t) = 1$ implique que $X_i(t) = N_i(t)$ alors que $\delta_i(t) = 0$ implique que $X_i(t) \geq N_i(t)$. On définit par ailleurs les bornes supérieure $\overline{X}_i(t) = X_i(t)$ et inférieure $\underline{X}_i(t) = X_i(t)\delta_i(t)$ de $X_i(t)$.

Afin de traiter uniquement un nombre fixé de séries temporelles au lieu de toutes celles indicées par les éléments de $\bigcup_{t=1}^{P} \mathcal{T}_M(t)$ (au plus $M \times P$), on choisit S adresses IP en sélectionnant d'abord celles étant apparues en première position, puis celles apparues en deuxième, etc. On obtient ainsi S éléments $i_1(1),\dots,i_1(P)$, $i_2(1),\dots,i_2(P)$, $i_3(1),\dots$ où les $i_k(t)$ ont été définies à la sous étape précédente ; dans le tableau de la figure 2.5, cela correspondrait à sélectionner les IP de la première ligne, puis celles de la seconde, etc. jusqu'à ce que l'on ait sélectionné S adresses différentes.

3. Test de détection de rupture : Lévy-Leduc et Roueff (2009) proposent d'utiliser le test non-paramétrique introduit par Gombay et Liu (2000) pour détecter des ruptures dans les données censurées. Ici, ce test est étendu à la détection de ruptures dans des séries temporelles doublement censurées (par valeurs inférieures et supérieures) afin de pouvoir utiliser la même procédure aussi bien dans les sondes locales qu'au niveau du collecteur central. Ce test, que l'on décrit ci-après, est utilisé sur chacune des séries temporelles créées dans les étapes précédentes. Une p-valeur est ainsi calculée et une petite valeur de celle-ci indique une anomalie potentielle.

Décrivons maintenant le test statistique utilisé. Cette procédure a pour but de tester à partir des observations $(\underline{X}_i(t), \overline{X}_i(t))_{1\leq t\leq P}$ si une rupture existe dans la série temporelle indicée par i. Plus précisément, et en omettant l'indice i par souci de clarté, les hypothèses testées sont

(H_0) : « $(\underline{X}(t), \overline{X}(t))_{1\leq t\leq P}$ sont indépendantes et identiquement distribuées. »

(H_1) : « Il existe r tel que

$$\big((\underline{X}(1), \overline{X}(1)), \ldots, (\underline{X}(r), \overline{X}(r))\big) \text{ et } \big((\underline{X}(r+1), \overline{X}(r+1)), \ldots, (\underline{X}(P), \overline{X}(P))\big)$$

ont des lois différentes. »

Définissons d'abord, pour chaque couple d'indices s, t de $\{1, \ldots, P\}$,

$$h(s,t) = \mathbf{1}(\underline{X}(s) > \overline{X}(t)) - \mathbf{1}(\overline{X}(s) < \underline{X}(t)) \ ,$$

où $\mathbf{1}(E) = 1$ lorsque l'événement E est réalisé et 0 sinon, et

$$Y_s = \frac{U_s}{\sqrt{\sum_{t=1}^{P} U_t^2}} \ , \quad \text{où} \quad U_s = \sum_{t=1}^{P} h(s,t) \ . \tag{3.1}$$

La statistique de test est donnée par

$$W_P = \max_{1\leq t\leq P} |\sum_{s=1}^{t} Y_s| \ . \tag{3.2}$$

Le théorème suivant, dont la démonstration est reportée à la section 3.6, donne, sous des hypothèses relativement faibles, la distribution asymptotique de W_P quand P tend vers l'infini, sous l'hypothèse nulle. Cela permet de calculer une p-valeur associée au test.

Théorème 1. *Soit $(\underline{X}, \overline{X})$ un vecteur aléatoire de $\mathbb{R}^2$ tel que*

$$\mathbb{P}(F(\underline{X}^-) + G(\overline{X}) = 1) < 1 \ , \tag{3.3}$$

où F est la fonction de répartition de $\overline{X}$, G est celle de $\underline{X}$ et $F(x^-)$ est la limite à gauche de F au point x. Soit $(\underline{X}(t), \overline{X}(t))_{1\leq t\leq P}$ des vecteurs aléatoires i.i.d. ayant la même distribution

que $(\underline{X}, \overline{X})$. *Alors, lorsque P tend vers l'infini,*

$$\sup_{0\leq u\leq 1}\left|\sum_{s=1}^{\lfloor Pu\rfloor} Y_s\right| \xrightarrow{d} B^\star := \sup_{0\leq u\leq 1}|B(u)|\,, \tag{3.4}$$

où $\{B(u)\ , 0 \leq u \leq 1\}$ *est un pont brownien,* $\xrightarrow{d}$ *désigne la convergence en loi et* $\lfloor x\rfloor$ *est la partie entière de x.*

Le Théorème 1 est une extension du Théorème 1 de Gombay et Liu (2000), celui-ci était en effet écrit en imposant une hypothèse de continuité sur les variables sous-jacentes, et uniquement pour une censure d'un seul côté.

Remarque 1. Le Théorème 1 donne un moyen de contrôler, pour un nombre d'observations assez grand, le taux de fausses alarmes asymptotique. La seule condition est (3.3), qui est une condition assez faible. En particulier, si les variables aléatoires $\underline{X}$ et $\overline{X}$ ont toutes deux une fonction de répartition continue, (3.3) est vérifiée lorsque $\mathbb{P}(\underline{X} = \overline{X}) > 0$, c'est-à-dire lorsque la probabilité de ne pas être censuré est strictement positive. En effet, $\mathbb{P}(F(\underline{X}) + G(\overline{X}) = 1) = \mathbb{P}(\{F(\underline{X}) + G(\overline{X}) = 1\} \cap \{\overline{X} = \underline{X}\}) + \mathbb{P}(\{F(\underline{X}) + G(\overline{X}) = 1\} \cap \{\overline{X} \neq \underline{X}\}) = \mathbb{P}(\{2F(\overline{X}) = 1\} \cap \{\overline{X} = \underline{X}\}) + \mathbb{P}(\{F(\underline{X}) + G(\overline{X}) = 1\} \cap \{\overline{X} \neq \underline{X}\})$. On peut observer que la première de ces probabilités est plus petite que $\mathbb{P}(2F(\overline{X}) = 1)$. Utilisant le fait que F est continue, $F(\overline{X})$ a une distribution uniforme sur $[0, 1]$, on a donc $\mathbb{P}(2F(\overline{X}) = 1) = 0$. Donc, $\mathbb{P}(F(\underline{X}) + G(\overline{X}) = 1) \leq \mathbb{P}(\overline{X} \neq \underline{X}) = 1 - \mathbb{P}(\overline{X} = \underline{X})$. En pratique, lorsque le nombre d'observations P et le nombre de valeurs non-censurées dans la série temporelle sont assez grands, les p-valeurs déduites du Théorème 1 sont fiables

Remarque 2. Une littérature abondante indique que les mesures de trafic réseau agrégées présentent des phénomènes de mémoire longue (voir par exemple Park *et al.* (2005) et ses références). L'hypothèse selon laquelle les variables aléatoires sont i.i.d. peut ainsi paraître surprenante dans une application d'analyse de trafic réseau. L'examen de flux origine–destination individuels ne permet cependant pas d'exhiber de fortes autocorrélations, en particulier sur des petites périodes d'analyse, ce qui permet de valider l'hypothèse d'indépendance ; voir Susitaival *et al.* (2006) pour un approfondissement sur le sujet.

Remarque 3. En pratique, en utilisant l'autre écriture de U_s faisant intervenir les fonctions de répartition empiriques de $\overline{X}(t)$ et $\underline{X}(t)$ (voir Eq. (3.10) page 58), le calcul des quantités $(\sum_{s=1}^{t} Y_s)_{1\leq t\leq P}$ peut être effectué en un nombre linéaire d'opérations.

À partir de l'équation (3.4), on associe à la statistique du test de détection de rupture la p-valeur $Pval(W_P)$ qui correspond à la probabilité de dépassement du supremum d'un pont brownien (mouvement brownien conditionné à s'annuler en

$t = 0$ et $t = 1$) : pour tout b strictement positif, (voir par exemple Billingsley, 1968, p. 85),

$$Pval(b) = \mathbb{P}(B^\star > b) = 2\sum_{j=1}^{\infty}(-1)^{j-1}e^{-2j^2b^2} \ .$$

4. Sélection des données à envoyer au collecteur central : Chaque sonde M_k choisit les d séries temporelles censurées ayant les plus petites p-valeurs et les transmet au collecteur central. Ainsi, au lieu de recevoir $\sum_{k=1}^{K} D_k$ séries temporelles censurées, ce qui serait le cas dans le cadre d'une approche centralisée, D_k étant le nombre d'adresses IP vues par le k-ème moniteur, le collecteur central ne reçoit au plus que $d \times K$ séries.

Agrégation et test de rupture dans le collecteur central

Au sein du collecteur central, les bornes inférieures et supérieures des séries temporelles agrégées $(\underline{Z}_i(t), \overline{Z}_i(t))_{1\leq t\leq P}$ associées à l'IP i sont ainsi construites de la manière suivante :

$$\underline{Z}_i(t) = \sum_{k\in\mathcal{K}} \underline{X}_i^{(k)}(t) \quad \text{et} \quad \overline{Z}_i(t) = \sum_{k\in\mathcal{K}} \overline{X}_i^{(k)}(t) \ , \tag{3.5}$$

où $(\underline{X}_i^{(k)}(t),\ t = 1,\dots,P)$ et $(\overline{X}_i^{(k)}(t),\ t = 1,\dots,P)$ sont les séries temporelles associées à l'adresse IP i calculées par la sonde M_k et $\mathcal{K}$ est l'ensemble des sondes qui ont effectivement transmis une série temporelle relative à l'adresse i.

Cette méthode d'agrégation a l'avantage d'utiliser une propriété importante des tests dérivés du test de rang de Wilcoxon : l'invariance lorsque les données sont multipliées par un facteur strictement positif (car il y a conservation des ordres relatifs entre les éléments). Lorsque le collecteur central agrège deux séries temporelles identiques, le test retourne une valeur de la statistique qui est égale à celle obtenue à partir des séries temporelles prises individuellement. C'est une propriété qui est désirable dans notre application : si l'on agrège les données provenant de deux sondes différentes mais ayant obtenu exactement les mêmes informations, il est bon que notre méthode ne déclare pas que l'on ait un phénomène beaucoup plus significatif, ce qui aurait été le cas avec un test classique de différence dans la moyenne, par exemple. La série $(\underline{Z}_i(t), \overline{Z}_i(t))_{1\leq t\leq P}$ est bien un encadrement de $(\sum_{k\in\mathcal{K}} X_i^{(k)}(t))_{1\leq t\leq P}$, éventuellement assez pessimiste. L'inconvénient de cette méthode d'agrégation est qu'elle augmente le taux de censure de la série, voir ci-après.

Le test décrit à la troisième étape du traitement local est par la suite appliqué à la série temporelle $(\underline{Z}_i(t), \overline{Z}_i(t))_{1\leq t\leq P}$. Une adresse IP i est alors déclarée comme attaquée au niveau de fausse alarme $\alpha \in (0,1)$ lorsque $Pval(W_P) < \alpha$. L'instant de rupture est le cas échéant estimé par $\hat{r} = \arg\max_{1\leq t\leq P} |\sum_{s=1}^{t} Y_s|$.

Comme indiqué dans la Remarque 1 ci-dessus, le Théorème 1 peut être appliquée à la série temporelle agrégée $(\underline{Z}_i(t), \overline{Z}_i(t))$ si celle-ci n'est pas complètement censurée. Par définition, la série agrégée est plus susceptible d'être censurée que les séries individuelles $(\underline{X}_i^{(k)}(t), \overline{X}_i^{(k)}(t))$ qui sont formées au niveau local. Mais d'autre part, pour une adresse IP donnée i, seules les séries présentant les d plus petites p-valeurs au niveau local sont remontées et agrégées au collecteur central. Celui-ci ne procède qu'à l'agrégation de moins de K séries, et seules celles-ci sont potentiellement significatives. Dans les conditions expérimentales décrites aux sections 3.3 et 3.4 (d prenant ses valeurs dans $\{1, \ldots, 5\}$, $P = 60$ et $M = 10$), le nombre moyen de valeurs non censurées dans les séries agrégées est de 18,7 (sur 60 possibles) et les séries qui sont complètement censurées représentent 0,62% de toutes les séries analysées. Le paramètre qui influence le plus ces valeurs est la profondeur M du tableau de filtrage par records : augmenter M réduit le taux de censure (pour $M = 20$, le nombre moyen de valeurs non censurées devient 31.2 sur 60 et le taux de séries complètement censurées est réduit à 0,017%) mais la quantité de mémoire et le temps de calcul nécessaires à l'étape de filtrage par records augmente avec M. Ainsi, une valeur de $M = 10$ représente un bon compromis ; voir aussi la section 3.4 pour une discussion plus approfondie.

3.2.2 La méthode *BTopRank*

Dans la suite, la méthode *DTopRank* est comparée à une méthode plus simple pour laquelle l'étape d'agrégation dans le collecteur central est remplacée par une procédure de comparaisons multiples, la correction de Bonferroni des p-valeurs déterminées dans chaque sonde. Plus précisément, avec le *BTopRank*, une adresse IP est déclarée comme attaquée au niveau $\alpha \in (0, 1)$ si au moins une des sondes locales a calculé une p-valeur inférieure à α/K, c'est-à-dire si $K(\inf_{1 \leq k \leq K} \mathrm{Pval}_k) < \alpha$, Pval_k étant la p-valeur calculée par la sonde M_k.

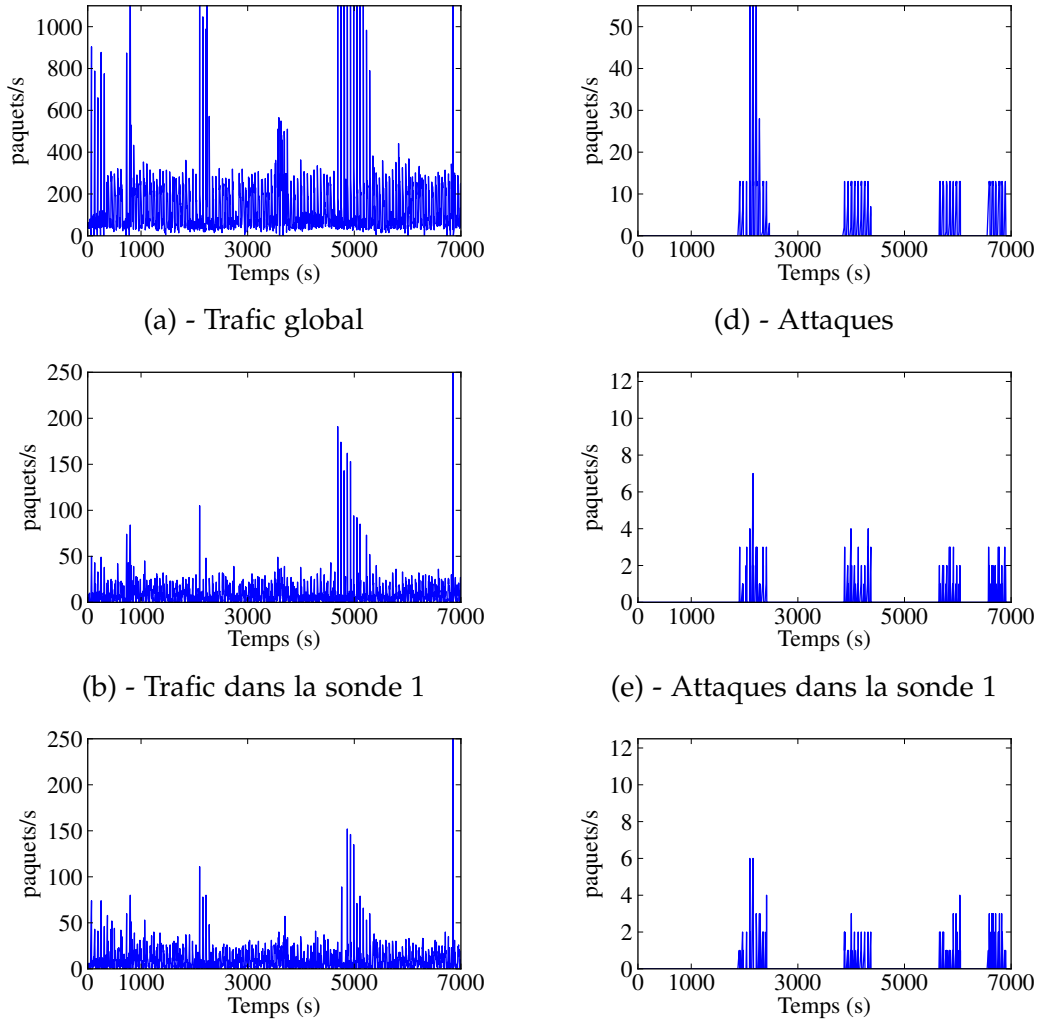
La correction de Bonferroni est la méthode la plus simple pour limiter le taux de fausses alarmes lorsque l'on tente de prendre une décision à partir de tests multiples sur un ensemble de données. En effet, le risque de rejeter à tort l'hypothèse nulle augmente avec le nombre de prises de décision supplémentaires. La correction de Bonferroni consiste alors à diviser le seuil de décision α par le nombre de tests effectués, ici K. Le taux de fausse alarme est contrôlé, mais la méthode devient alors très conservative (en particulier pour K grand), ce qui diminue le taux de détection. On peut se référer par exemple à Dudoit *et al.* (2003) pour d'autres procédures de tests multiples.

3.3 Application à un jeu de données réelles

Dans cette section, nous résumons les résultats expérimentaux obtenus par les algorithmes *DTopRank* et *BTopRank* lorsqu'on les applique à des données de trafic

utilisées par Lévy-Leduc et Roueff (2009) et fournies par un opérateur de télécommunications majeur.

3.3.1 Description des données



(a) - Trafic global

(d) - Attaques

(b) - Trafic dans la sonde 1

(e) - Attaques dans la sonde 1

(c) - Trafic dans la sonde 2

(f) - Attaques dans la sonde 2

FIGURE 3.1 – Nombre global de paquets TCP/SYN échangés (colonne de gauche) et reçus en particulier par les 4 adresses IP attaquées (colonne de droite) dans les données originales (a,d) et celles enregistrées par deux sondes en particulier (b, c, e, f). À noter que les valeurs des données s'étendent sur une échelle 20 fois plus petite sur les figures de la colonne de droite par rapport à celles de gauche.

On considère les données utilisées par Lévy-Leduc et Roueff (2009) dans la section 4 de leur article. Ces données correspondent à un enregistrement de 118 minutes de trafic ADSL et de trafic pair à pair (P2P), auxquels ont été ajoutés des traces d'attaques réseau par engorgement de paquets TCP/SYN (*TCP/SYN flooding attack*). Ces traces d'attaque représentent les événements de référence que les algorithmes proposés doivent détecter. Notons que ces attaques étant superposées à du trafic réel, il n'est pas exclu que ces traces de données contiennent des événements pouvant être considérés comme des attaques mais n'ayant pu être annotés comme telles. Ce jeu de données ne contient pas d'informations de routage ; aussi a-t-il été artificiellement distribué entre plusieurs sondes virtuelles de la manière suivante : les données sont partagées entre $K = 15$ sondes en assignant à chacun des couples Source–Destination l'une des K sondes choisie au hasard. Ainsi, une sonde enregistre tous les flux et données de trafic entre deux adresses IP en particulier. Les résultats montrés ci-après sont obtenus à partir d'un moyennage sur 50 réplications indépendantes de ce processus de routage simulé. Par ailleurs, des méthodes de sous-échantillonnage (en écartant au hasard certains paquets correspondants aux attaques) ont été utilisées afin de moduler l'intensité des attaques. Celle-ci est ainsi diminuée à 25 et 12,5 paquets par seconde afin d'explorer des scénarios d'attaque plus difficiles à détecter.

Le profil du trafic réseau enregistré dans ce jeu de données est représenté dans la figure 3.1. Le nombre total de paquets de type TCP/SYN reçus à chaque seconde par l'ensemble des adresses IP sollicitées est mis en évidence en (a) et le nombre de paquets reçus par les quatre adresses IP destination attaquées en (d). On peut voir que les attaques se produisent autour des instants 2 000 s, 4 000 s, 6 000 s et 6 500 s. Elles constituent au total 33 anomalies (augmentation ou diminution abrupte du signal) de référence à détecter. Les figures 3.1-(b), (c) exhibent le nombre de paquets globalement échangés vus par deux moniteurs différents ; (e), (f) sont les représentations du trafic reçu par les adresses IP attaquées vu par ces mêmes moniteurs.

Le trafic correspondant aux attaques (colonne de droite de la figure 3.1) est complètement caché dans le trafic total (colonne gauche) et est donc difficile à déceler. Notons aussi que 1 006 000 adresses IP destination sont présentes dans ce jeu de données, avec en moyenne 15 000 adresses vues dans chacune des 118 fenêtres d'observation d'une minute. Ainsi, le traitement en temps réel des données serait difficilement possible, même au niveau des sondes locales, sans une étape de réduction de dimension des données (comme le filtrage par records).

3.3.2 Évaluation des performances des méthodes

Dans la suite, le *DTopRank* est utilisé avec les mêmes paramètres que ceux adoptés par Lévy-Leduc et Roueff (2009) pour l'algorithme du *TopRank*. Les fenêtres d'observations d'une minute sont divisées en $P = 60$ sous-intervalles de $\Delta = 1$ se-

conde, le paramètre de filtrage est $M = 10$ et le nombre total de séries analysées par une sonde est $S = 60$. Le paramètre d est fixé à 1 au vu du nombre limité d'attaques attendues dans chaque fenêtre. Le choix de P et de Δ doit résulter d'un compromis sur la durée totale de la fenêtre d'observation ; celle-ci doit être suffisamment longue pour obtenir des décisions statistiquement significatives mais assez courte pour garantir un délai de détection acceptable et pour que les séries temporelles étudiées puissent toujours être considérées comme étant stationnaires en l'absence de rupture. On peut aussi noter que la complexité en temps et en espace de la procédure est proportionnelle à P. L'influence des autres paramètres (d, M et S) est discutée à la fin de la section 3.4.2.

Les figures 3.2 et 3.3 mettent en évidence les avantages de la méthode d'agrégation du *DTopRank* par rapport à la comparaison de tests multiples avec correction de Bonferroni qui sont mis en œuvre dans le dans le *BTopRank*. Les illustrations 3.2-(a), (b) et (c) montrent les séries temporelles locales $(\underline{X}(t),\ t = 1, \ldots, P)$ et $(\overline{X}(t),\ t = 1, \ldots, P)$ associées à une adresse IP attaquée vues par trois sondes différentes ainsi que les p-valeurs associées. La figure (d) montre quant à elle les séries temporelles agrégées $(\underline{Z}(t),\ t = 1, \ldots, P)$ et $(\overline{Z}(t),\ t = 1, \ldots, P)$ définies en (3.5) avec la p-valeur associée. Cette série est le résultat de l'agrégation de séries provenant de 11 sondes différentes où l'adresse IP attaquée a été vue. La p-valeur associée à la série temporelle agrégée est beaucoup plus petite que celles qui sont calculées au niveau local ; ceci est l'illustration que la méthode proposée permet la détection d'attaques qui seraient difficiles à détecter de manière individuelle au niveau local.

Dans la figure 3.3, on affiche le couple p-valeur calculée avec le *DTopRank* — affichée en abscisse — et p-valeur calculée avec *BTopRank* — en ordonnée — pour chacune des adresses IP analysées. Pour les adresses attaquées, le *DTopRank* calcule des p-valeurs plus petites que celles obtenues avec la méthode de Bonferroni ; pour la plus grande part de toutes les autres adresses, les deux quantités sont du même ordre. Un examen des séries temporelles correspondant aux points dont la p-valeur calculée par le *DTopRank* est beaucoup plus petite que celle calculée par *BTopRank* montre que ces séries font aussi apparaître des ruptures. Les adresses IP correspondantes ne sont pas des adresses dont on sait qu'elles ont été attaquées mais appartiennent au trafic de fond, réel ; il est donc fortement possible qu'elles aient subi des anomalies et que celles-ci ne soient pas annotées.

Les algorithmes *DTopRank* et *BTopRank* sont comparés dans deux cas différents et les résultats sont représentés dans la figure 3.4 sous forme de courbes ROC (*Receiver Operating Characteristic*) qui représentent, pour différents seuils de décision, en ordonnée le taux de détection correcte et en abscisse le taux de fausse alarme obtenus à partir de 50 réplications de Monte-Carlo des expériences. Rappelons que 15 000 adresses IP différentes sont traitées ici dans chaque fenêtre d'observation alors que le nombre d'exemples positifs est plutôt faible (de l'ordre de quelques

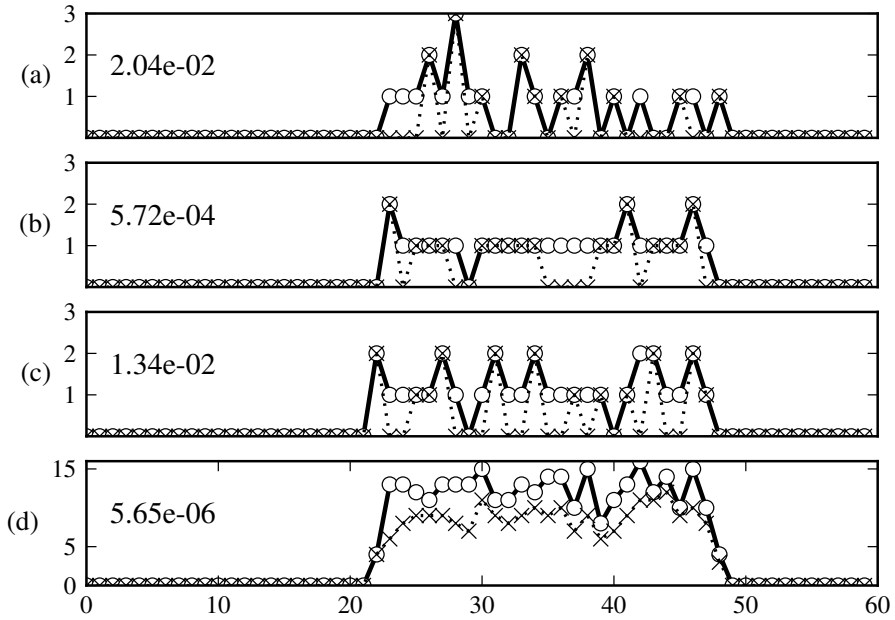


Figure 3.2 – (a), (b), (c) : séries temporelles $(\underline{X}_i^{(k)}(t),\ t = 1,\ldots,60)$, représentée par ('×'), et $(\overline{X}_i^{(k)}(t),\ t = 1,\ldots,60)$ ('∘'), pour 3 valeurs différentes de k ; (d) : $(\underline{Z}_i(t),\ t = 1,\ldots,60)$ ('×') et $(\overline{Z}_i(t),\ t = 1,\ldots,60)$ ('∘').

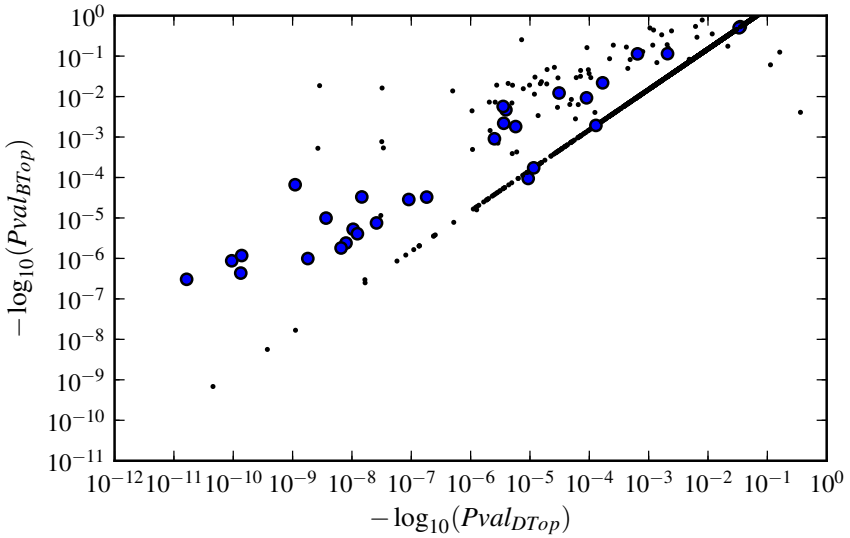


Figure 3.3 – $(\mathrm{Pval}_{\mathrm{DTop}}, \mathrm{Pval}_{\mathrm{BTop}})$ représentées par ('.'), sauf pour les adresses IP identifiées comme attaquées, représentées par ('•').

dizaines sur l'ensemble de la trace de données). À ce titre il n'est donc intéressant que de s'intéresser à des taux de fausses alarmes très faibles et donc à un seuil sur la p-valeur très petit afin de ne pas alerter intempestivement le superviseur de réseau de manière trop fréquente.

Le graphe du haut présente les résultats des expériences comportant des attaques ayant une intensité de 25 paquets SYN par seconde. Dans le graphe du bas, les attaques ont été sous-échantillonées à 12,5 paquets par seconde. Cette figure montre que pour des attaques d'intensité 25 paquets par seconde, les deux algorithmes produisent des résultats similaires. Dans le scénario plus difficile des attaques à 12,5 paquets par seconde, l'algorithme du *DTopRank* produit de meilleures performances de détection que le *BTopRank*.

Nous avons aussi intégré à ces courbes les résultats obtenus grâce à la méthode centralisée du *TopRank*. Le *DTopRank* obtient des résultats très similaires à ceux de l'algorithme centralisé, en particulier à des taux de fausse alarme très faibles, à environ 10^{-4} alors que la quantité de données échangées au sein du réseau est grandement réduite par la décentralisation. En effet, l'algorithme centralisé traite en moyenne 34 000 flux par minute ; dans le cadre du *DTopRank*, les sondes ne transmettent au collecteur central que d séries temporelles constituées des bornes inférieures et supérieures de la vraie série temporelle, ce qui revient à 1 800 valeurs scalaires lorsque $d = 1$ et $K = 15$. Ce sont $34\,000 \times 5$ (dates de début et de fin de la communication, adresses source et destination, nombre de paquets pour chacun des flux) valeurs qui sont transmises dans le cadre centralisé, ce qui fait que la distribution des calculs diminue de deux ordres de grandeur la quantité de données transmises au sein du réseau.

3.4 Application à un jeu de données synthétique

Dans cette section, nous faisons la synthèse de résultats de simulations qui ont été effectuées pour deux raisons spécifiques. Premièrement, la trace de trafic utilisée dans la section 3.3 n'est pas entièrement étiquetée : elle contient des attaques qui sont bien marquées comme telles, mais il est possible que des anomalies supplémentaires soient présentes dans le trafic de fond ADSL et P2P. Cela peut conduire à une surestimation du nombre de fausses alarmes (voir Lévy-Leduc et Roueff, 2009). Deuxièmement, la méthode de décentralisation aléatoire utilisée à la section précédente ne reflète pas la topologie d'un réseau de données réaliste. On considère donc dans cette section des données synthétiques de grandes dimensions correspondant à une minute de trafic contenant une anomalie et qui serait mesurée par 15 sondes positionnées de manière aléatoire au sein d'une topologie réseau plausible.

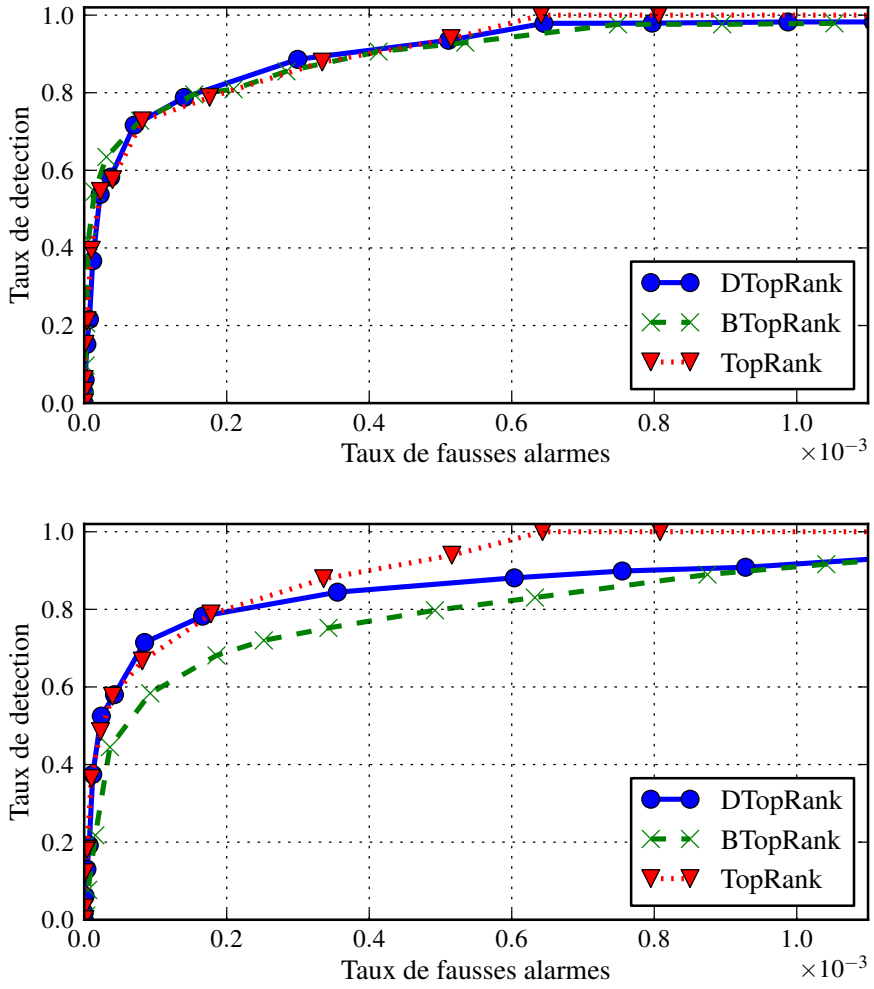


Figure 3.4 – Courbes ROC pour les algorithmes *DTopRank*, *BTopRank* et *TopRank* appliqués aux données contenant des attaques d'intensités 25 SYN/s (haut) et 12.5 SYN/s (bas).

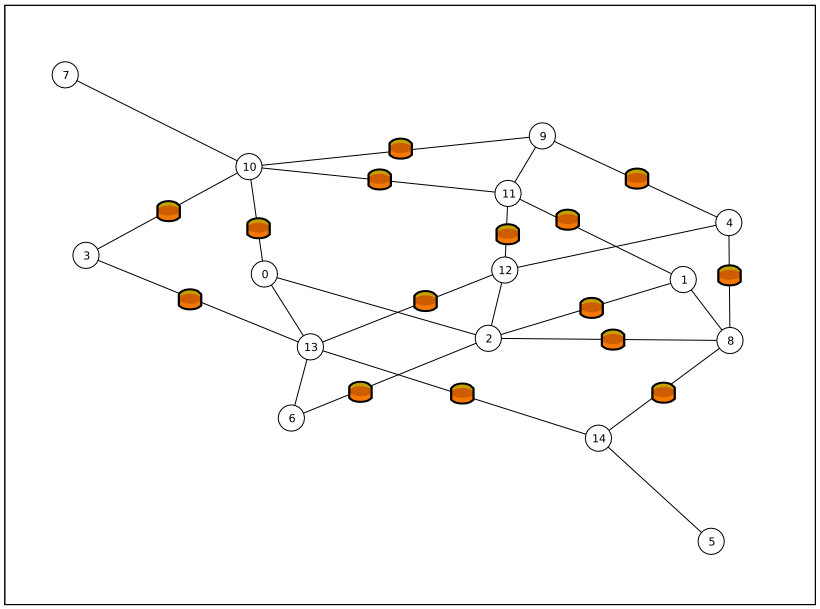


FIGURE 3.5 – Graphe généré : les nœuds du réseau sont représentés par des nombres encerclés et les sondes par des boîtes colorées.

3.4.1 Description des données

On génère une topologie de réseau dans lequel le trafic entre différentes entités situées dans les nœuds du réseau est injecté. Pour cela, on commence par générer un graphe aléatoire de Erdős-Rényi (1959) de 15 nœuds avec une probabilité de création d'arêtes de 0,15. Le graphe ainsi engendré est représenté dans la figure 3.5. Son nombre de nœuds ainsi que leurs degrés (nombre d'arêtes reliées à ce nœud) est similaire à ceux du réseau Abilene qui est souvent considéré dans la littérature sur la détection d'anomalies dans les réseaux, voir par exemple Lakhina *et al.* (2004); Huang *et al.* (2007). Ce graphe est généré une fois et est utilisé dans toutes les réplications des simulations de Monte-Carlo qui vont suivre. Dans chacune de ces réplications, un nœud du graphe est attribué aléatoirement à chacune des $D = 1000$ adresses IP et $K = 15$ sondes sont également positionnées sur 15 des 24 arêtes du graphe. Un exemple de l'une des répartitions possibles des sondes est reproduit dans la figure 3.5.

Les routes entre chaque nœud réseau, c'est-à-dire la liste des arêtes du graphe qui forment un chemin entre chaque nœud, est calculé avec l'algorithme du plus

court chemin de Dijkstra (1959). On utilise ces routes[1] pour déterminer quelles sondes enregistrent le trafic entre deux hôtes du réseau.

Le trafic injecté dans ce réseau est généré de la manière décrite ci-après. Pour un couple Source–Destination (i, j) donné, on modélise comme Lévy-Leduc et Roueff (2009) le trafic de paquets SYN par un processus de Poisson ayant pour intensité $\theta_{i,j}$ correspondant au nombre de paquets SYN par sous-intervalle de la fenêtre d'observation envoyés par i et reçus par j. Dans les applications réseau, des couples Source–Destination différents échangent une quantité très différente de trafic, c'est pourquoi l'intensité du processus sera différente pour chaque couple. Pour prendre en compte cette diversité, nous proposons d'utiliser des réalisations d'une distribution de Pareto comme paramètres de ces différentes intensités. Ainsi un grand nombre d'hôtes du réseau recevront un petit nombre de paquets tandis qu'une petite partie d'entre eux en recevront une grande quantité. L'utilisation de distributions à queue lourde pour la modélisation de trafic réseau est courante dans la littérature, voir Nucci *et al.* (2005) et les références associées.

On génère ainsi une séquence $(\mu_k)_{1\leq k\leq N}$ de N intensités suivant la distribution de Pareto de densité de probabilité

$$\gamma\alpha/(1+\gamma x)^{1+\alpha}$$

pour $x > 0$, avec $\alpha = 2{,}5$ et $\gamma = 0{,}72$. Ces valeurs correspondent approximativement à celles observées dans le trafic réel centralisé qui a été utilisé à la section 3.4. On suppose que les paramètres μ_k sont triés, de sorte que $\mu_1 \geq \cdots \geq \mu_N$.

$(X_{i,j}(t))_{1\leq t\leq P}$ est le nombre de paquets SYN envoyés par i et reçus par j dans chacun des P sous-intervalles de la fenêtre d'observation, i et j étant dans $\{1,\dots,D\}$. Parmi ces N séries temporelles, N_a d'entre elles correspondent au trafic reçu par l'adresse IP attaquée j_0 à laquelle on attribue une position fixe, le nœud 7 qui est situé à la périphérie du réseau (voir la figure 3.5). Ce flux de trafic, qui est transmis par l'adresse IP source i appartenant à un sous-ensemble $\mathcal{I}_a$ de $\{1,\dots,D\}$ choisi aléatoirement, est généré de la manière suivante :

$$\forall i \in \mathcal{I}_a, (X_{i,j_0}(t))_{1\leq t\leq \tau} \overset{iid}{\sim} \text{Poisson}(\theta_{i,j_0}) \ ,$$

et

$$\forall i \in \mathcal{I}_a, (X_{i,j_0}(t))_{\tau< t\leq P} \overset{iid}{\sim} \text{Poisson}(\eta\theta_{i,j_0}) \ ,$$

où η est un nombre strictement positif qui permet de moduler l'intensité du changement, τ est l'instant de rupture et $(\theta_{i,j_0})_{i\in\mathcal{I}_a}$ sont choisis parmi $(\mu_k)_{40N_a\leq k\leq 41N_a}$. Les valeurs de $(\theta_{i,j_0})_{i\in\mathcal{I}_a}$ ont ainsi une valeur approchant de 0,6 (valeur du quarantième centile de la distribution de Pareto de paramètres α et γ choisis ci-dessus, de

[1]Notons que dans cette étude, nous ne prenons pas en compte la capacité des liens du réseau, ce qui requerrait des algorithmes de routage dynamique et une répartition du trafic entre les sondes plus compliquée.

moyenne 0,93, c'est-à-dire que 60% des intensités de tous les flux sont plus élevés que l'intensité de base des flux vers les IP attaquées). Ainsi, l'attaque à détecter est composée de flux provenant de N_a sources qui augmentent de manière multiplicative l'intensité du nombre de paquets envoyés, intensité qui avant le début de l'attaque est noyée dans la masse de la distribution des intensités (proche du quarantième centile). Le reste du trafic de fond (ne correspondant pas aux attaques) est généré comme suit :

$$\forall i \in \{1, \dots, D\},\ j \neq j_0,\ (X_{i,j}(t))_{1 \leq t \leq P} \overset{iid}{\sim} \text{Poisson}(\theta_{i,j}) ,$$

où les $(\theta_{i,j})_{i \in \{1,\dots,D\}, j \neq j_0}$ sont choisis aléatoirement parmi les valeurs restantes de $\mu_k : (\mu_k)_{k \notin [40N_a; 41N_a]}$.

Pour les simulations présentées ci-dessous, on choisit $N = 10100$, $N_a = 100$, $P = 60$, $\tau = 30$ et l'on considère différentes valeurs du paramètre η (1,2; 1,5) afin de moduler la difficulté de détection. Les résultats présentés dans les figures 3.6 et 3.7 correspondent au cas où $d = 1$; l'influence de ce paramètre est discuté dans le paragraphe final de la section 3.4.2. Avec ces paramètres, les attaques de type DDoS contre j_0 font intervenir un grand nombre N_a de sources de trafic provenant de tous les nœuds du réseau. Il est ainsi très difficile au niveau local (dans les sondes) de distinguer le trafic correspondant aux attaques du trafic de fond. Ceci est illustré à la figure 3.6 qui montre d'une part pour chacune des sondes, lorsque $\eta = 1{,}5$, un exemple de séries temporelles formées par le nombre de paquets reçus par la première ("$\times$") et dixième ("$\bullet$") IP la plus sollicitée à chaque sous-intervalle – précisons bien que cette première ou dixième IP peut être différente à chaque instant ; et d'autre part la série temporelle de l'adresse attaquée j_0 ("$\triangleright$") ; le trafic dans les sondes qui n'ont pas enregistré de trafic en direction de j_0 n'est pas affiché sur cette figure. Dans les figures 3.6-(d), (e) et (i), le trafic vers j_0 est enregistré par les sondes mais le nombre de paquets transmis n'est jamais assez élevé pour que j_0 soit sélectionnée à l'étape correspondant au filtrage par records et appartienne à $\{\mathcal{T}_M(t),\ t = 1, \dots, 60\}$. Dans ces sondes, aucun test de rupture n'est ainsi effectué pour j_0. En revanche dans les six autres figures, l'ensemble des étapes locales du *DTopRank* sont exécutées. Un cas particulier est représenté dans la figure 3.6-(a) : la série temporelle correspondant à la sonde située sur l'arête du graphe entre les nœuds 7 et 10 (voir figure 3.5) ; c'est le lien où l'intégralité du trafic destiné à l'adresse attaquée j_0 apparaît.

3.4.2 Performance des méthodes

Les deux méthodes décrites à la section 3.2 sont comparées en calculant leurs taux de détection correcte et de fausse alarme lorsqu'elles sont testées sur 1 000 réplications de Monte-Carlo des expériences décrites à la section 3.4.1. Le graphique de gauche de la figure 3.7 montre les courbes ROC correspondant aux différentes va-

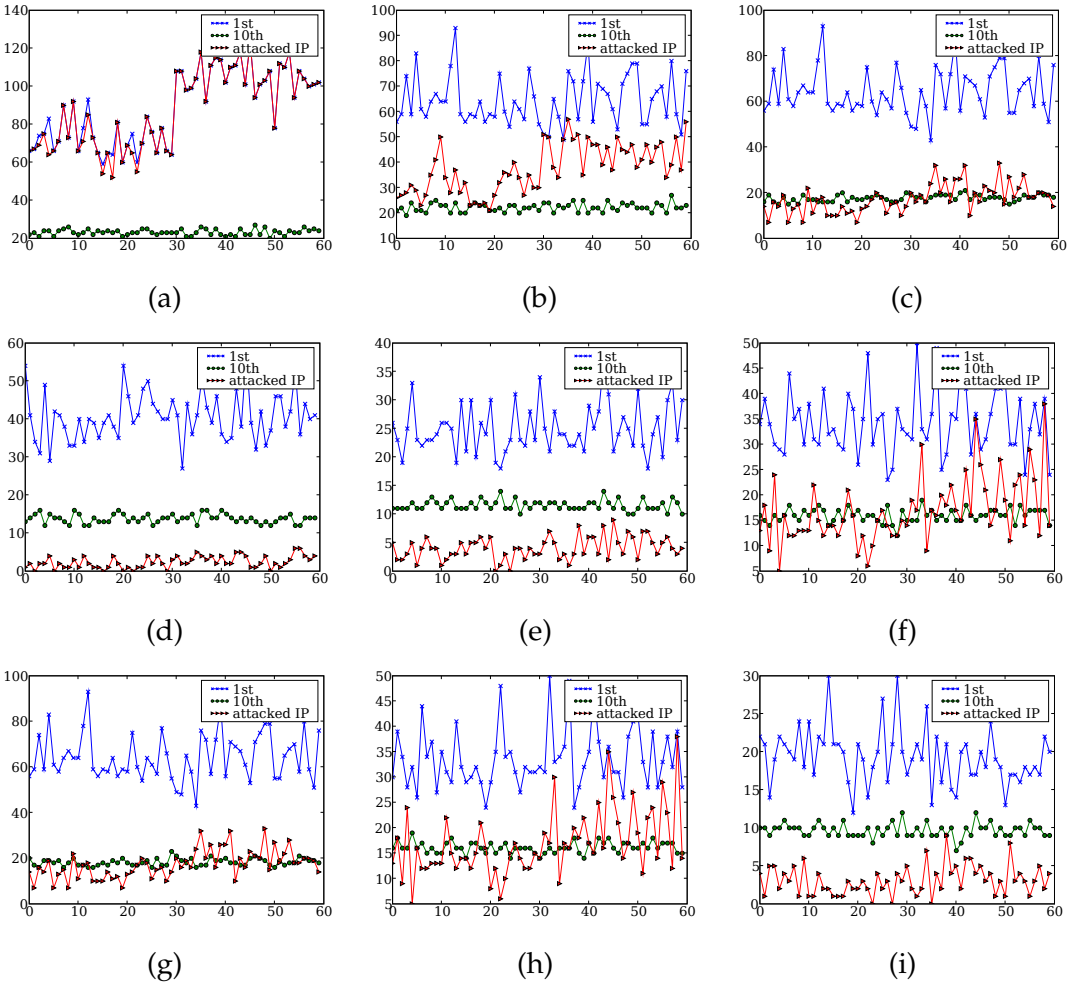


Figure 3.6 – Séries temporelles formées dans 9 moniteurs par les nombres de paquets reçus par la première ("×") et dixième ("•") adresse IP la plus sollicitée à chaque sous intervalle et série temporelle du nombre de paquets reçus par l'adresse attaquée j_0 ("▷").

leurs du paramètre η (1,2 et 1,5) ; les courbes en traits pleins montrent les résultats du *DTopRank* et celles en pointillés ceux du *BTopRank*.

Pour une valeur élevée de η (1,5), les deux méthodes ont de très bons résultats, légèrement meilleurs pour le *DTopRank*, avec de faibles nombres d'attaques manquées et de fausses alarmes. Pour $\eta = 1{,}2$, valeur pour laquelle les attaques sont plus difficilement détectables, si les performances de détection sont naturellement plus basses pour les deux algorithmes, la dégradation des performances est plus marquée pour le *BTopRank* que pour le*DTopRank*.

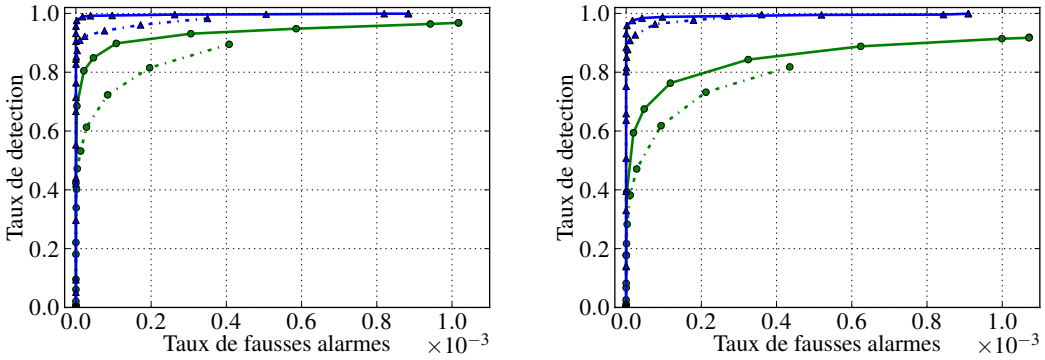


FIGURE 3.7 – À gauche : courbes ROC de *DtopRank* (traits pleins) et *BTopRank* (pointillés) pour $\eta = 1.2$ ("•") et 1.5 ("▲"). À droite : même protocole de simulation mais en interdisant aux sondes d'apparaître sur l'arête 10-7.

Influence de la position des cibles dans la topologie réseau Nous avons par ailleurs constaté que les performances de détection étaient meilleures lorsqu'une sonde était placée sur l'arête entre les nœuds 7 et 10 de la figure 3.5. Dans ce cas, au moins une sonde a en effet accès à toutes les informations sur le trafic qui a pour destination l'adresse IP attaquée qui est située au nœud 7 du graphe. Nous avons reproduit ces expériences de simulation en interdisant l'attribution de cette arête à une sonde ; la partie droite de la figure 3.7 représente les résultats de cette expérience et permet de donner une idée de l'importance de ce phénomène. Pour une répartition de sondes donnée, la performance de détection est ainsi meilleure si la cible des attaques est située à la périphérie du réseau, « derrière » une sonde qui a accès à un maximum d'informations possibles la concernant. Dans le cas contraire, les performances sont quand mêmes appréciables, ce qui est dû à l'agrégation effectuée au niveau du collecteur central de l'information envoyée par les sondes.

Influence des différents paramètres Étudions maintenant l'influence de d sur les performances de *DTopRank*. Dans la figure 3.8, on exhibe les courbes ROC obtenues pour le *DTopRank* utilisé avec plusieurs valeurs de d (d=1, 5, 10) ainsi qu'un nombre variable d'attaques au sein d'une fenêtre d'observation. On peut d'abord faire le constat, d'après la figure supérieure, qu'une valeur de $d = 1$ est optimale lorsqu'une seule attaque se produit mais est moins avantageuse lorsqu'un nombre plus élevé d'attaques apparaît. En effet, le collecteur central ne reçoit dans ce cas qu'une seule série temporelle par sonde, ce qui mène à de faibles taux de détection. Augmenter la valeur de d améliore donc les performances lorsque plusieurs attaques peuvent se produire. Cependant, la figure inférieure révèle un effet de seuil, et $d = 5$ semble être le meilleur compromis, même lorsqu'on s'attend à un nombre d'attaques supérieur. On peut expliquer ce comportement surprenant par le constat que le trafic à destination d'une adresse IP particulière n'est pas forcément visible par toutes les sondes et qu'augmenter d à de plus grandes valeurs ne permet pas d'agréger des séries temporelles supplémentaires qui pourraient améliorer les performances de détection.

Nous avons aussi exploré le rôle des paramètres M (le nombre d'adresses sélectionnées à chacun des P sous-intervalles de la fenêtre) et S (le nombre maximal de séries temporelles analysées) définis dans la section 3.2.1. La figure 3.9 met en évidence l'impact de M alors que l'on fixe S à une valeur constante de 60 ; le nombre d'attaques dans chaque fenêtre ainsi que le paramètre d sont tous deux égaux à 5. Les mêmes expériences ont montré que lorsqu'on modifie la valeur de S à 30 ou 120, la courbe ROC n'est pas affectée. On peut aussi observer sur cette même figure que plus M est grand, meilleures sont les performances, phénomène qui est lié au degré de censure des séries temporelles qui diminue lorsque M augmente. Il ne faut cependant pas oublier que le temps de calcul et l'empreinte mémoire au sein des sondes est proportionnel à M ; aussi le choix de ce dernier paramètre doit-il résulter d'un compromis entre les performances de détection et la consommation de mémoire et de temps de calcul. L'absence d'effet de S peut être expliqué par le fait que seules d valeurs sont au collecteur central par la sonde. Il suffit donc que S soit plus grand que d et de l'ordre de P pour pouvoir enregistrer l'activité des plus grandes valeurs des comptes $N_i(t)$ pour tout $t \in \{1, \ldots, P\}$.

3.5 Conclusion du chapitre

Dans ce chapitre, nous avons proposé une méthode qui permet la détection d'anomalies (plus précisément de ruptures dans un signal) réseau à partir de données collectées dans plusieurs sondes. Elle est par exemple applicable si l'on veut détecter des attaques de type déni de service distribué. Si sa mise en œuvre et ses performances nous semblent satisfaisantes, il reste quelques points qui seraient à explorer.

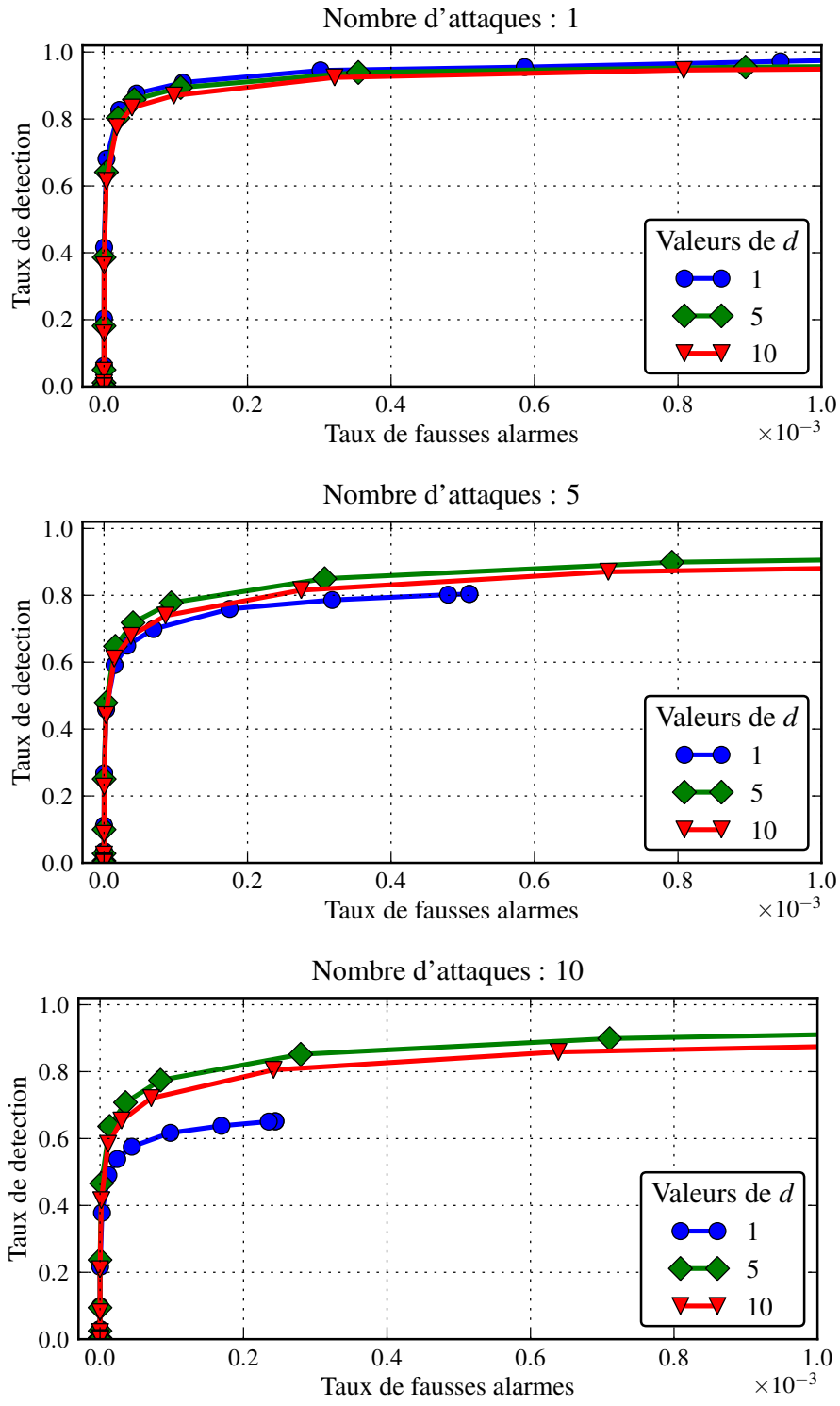


FIGURE 3.8 – Courbes ROC pour le *DTopRank* lorsque le nombre d'attaque par fenêtre d'observation est de 1, 5 et 10 (de haut en bas) et pour des valeurs de d égales à 1 ("•"), 5 ("♦"), 10 ("▼").

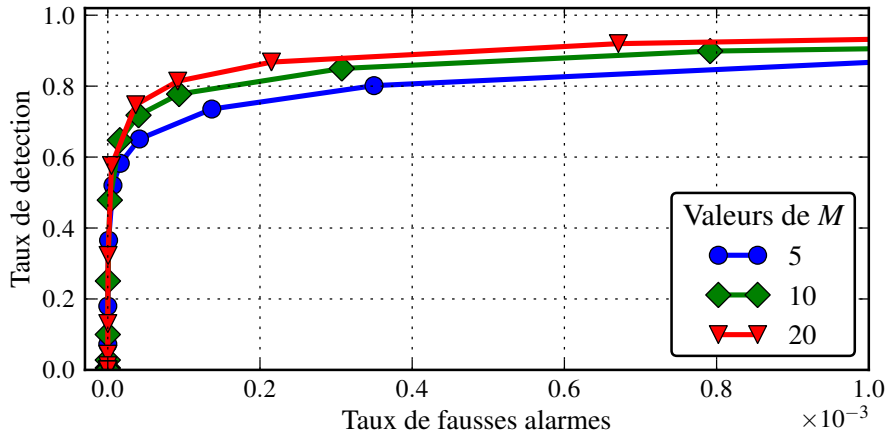


Figure 3.9 – Courbes ROC pour le *DTopRank* pour M prenant pour valeurs 5 ("•"), 10 ("♦") ou 20 ("▼").

Le premier point concerne l'évaluation des performances par les simulations. Si le premier jeu de données est constitué d'exemples réels, une large part reste artificielle (méthode de décentralisation des données, nature des attaques qui ont été injectées à des traces de trafic). Il est en effet très difficile d'obtenir des données étiquetées correspondant au cadre exact de notre étude ; en particulier, nous n'avons pas pu trouver de mesures réseau effectuées en plusieurs points.

D'autre part, le Toprank décentralisé utilise une méthode heuristique pour agréger les données provenant de plusieurs sondes, il prend en compte le fait que les événements qui doivent alerter un opérateur sont une augmentation du nombre de paquets. Cette heuristique est plutôt performante et sa mise en œuvre simple, mais n'exploite pas l'ensemble des informations disponibles. En particulier, la structure de dépendance qui relie les données provenant des différentes sondes est perdue. Ces relations de dépendance pourraient par exemple provenir de la topologie du réseau (les séries temporelles provenant de deux sondes proches vont être très corrélées, par exemple). Dans la section 7.1, nous étudions l'apport éventuel de la prise en compte de ces relations de dépendance.

Enfin, la décentralisation effectuée ici n'est que partielle : un traitement de données est effectué au sein des sondes placées dans le réseau, mais les informations sont transmises à un point de collecte central. Il serait intéressant de pousser plus loin la décentralisation, c'est-à-dire de s'affranchir d'un centre de décision et d'élaborer des méthodes plus collaboratives dans lesquelles les décisions seraient prises conjointement par les sondes.

3.6 Démonstration du théorème 1

La démonstration du théorème 1 s'appuie sur le théorème 24.2 de Billingsley, 1968 :

Théorème 2 (Billingsley, 1968)**.** *Soit $\xi_1, \ldots, \xi_n$ des variables aléatoires échangeables, telles que, lorsque n tend vers l'infini, on a :*

$$\sum_{i=1}^{n} \xi_i \xrightarrow{p} 0, \quad \sum_{i=1}^{n} \xi_i^2 \xrightarrow{p} 1, \quad \max_{1 \leq i \leq n} |\xi_i| \xrightarrow{p} 0 \,. \tag{3.6}$$

Alors,

$$\{ \sum_{i=1}^{\lfloor nt \rfloor} \xi_i \,, 0 \leq t \leq 1 \} \xrightarrow{d} \{ B(t) \,,\ 0 \leq t \leq 1 \} \tag{3.7}$$

quand n tend vers l'infini, où B désigne un pont brownien.

Nous appliquons ce théorème aux variables aléatoires $Y_1, \ldots, Y_P$, définies en (3.1), qui sont échangeables (c'est-à-dire que toutes les permutations de cet ensemble de variables ont la même distribution jointe) puisque $(\underline{X}(i), \overline{X}(i))_{1 \leq i \leq P}$ sont des vecteurs i.i.d. Gombay et Liu (2000) ont brièvement évoqué l'utilisation de ce théorème de Billingsley pour justifier leur résultat. Les hypothèses utilisées dans la démonstration ci-dessous sont cependant moins restrictives que celles utilisées par Gombay et Liu (2000) ; en particulier, nous ne requérons pas la continuité des fonctions de répartition des variables aléatoires $\overline{X}(i)$ et $\underline{X}(i)$, $1 \leq i \leq P$.

Vérifions maintenant les trois conditions de (3.6). Par antisymétrie du noyau h,

$$\sum_{i=1}^{P} U_i = \sum_{i=1}^{P} \sum_{j=1}^{P} h(i,j) = 0 \,,$$

ce qui nous donne la première condition. La deuxième provient de la définition de Y_i :

$$\sum_{i=1}^{p} Y_i^2 = \frac{1}{\sum_{j=1}^{p} U_j^2} \sum_{i=1}^{p} U_i^2 = 1 \,.$$

Pour vérifier la troisième condition, notons F_P la fonction de répartition empirique de $\overline{X}(1), \ldots, \overline{X}(P)$ et G_P celle de $\underline{X}(1), \ldots, \underline{X}(P)$:

$$F_P(t) = P^{-1} \sum_{i=1}^{P} \mathbf{1}(\overline{X}(i) \leq t) \,; \tag{3.8}$$

$$G_P(t) = P^{-1} \sum_{i=1}^{P} \mathbf{1}(\underline{X}(i) \leq t) \,. \tag{3.9}$$

Notons que

$$\begin{aligned}\frac{1}{P}U_i &= \frac{1}{P}\sum_{j=1}^{P}\mathbf{1}(\underline{X}(i) > \overline{X}(j)) - \mathbf{1}(\overline{X}(i) < \underline{X}(j)) \\ &= F_P(\underline{X}(i)^-) - \{1 - G_P(\overline{X}(i))\} \\ &= F_P(\underline{X}(i)^-) - \overline{G}_P(\overline{X}(i)) \,, \end{aligned} \tag{3.10}$$

où $\overline{G}_P(\cdot) = 1 - G_P(\cdot)$. En invoquant le théorème de Glivenko-Cantelli (van der Vaart, 1998, théorème 19.1), on a donc, lorsque P tend vers l'infini, que

$$\begin{aligned}\frac{1}{P}\sum_{j=1}^{P}\left(\frac{1}{P}U_j\right)^2 &= \frac{1}{P}\sum_{j=1}^{P}F_P(\underline{X}(i)^-)^2 - \frac{2}{P}\sum_{j=1}^{P}F_P(\underline{X}(i)^-)\overline{G}_p(\overline{X}(i)) + \frac{1}{P}\sum_{j=1}^{P}\overline{G}_p(\overline{X}(i))^2 \\ &= \frac{1}{P}\sum_{j=1}^{P}F(\underline{X}(i)^-)^2 - \frac{2}{P}\sum_{j=1}^{P}F(\underline{X}(i)^-)\overline{G}(\overline{X}(i)) + \frac{1}{P}\sum_{j=1}^{P}\overline{G}(\overline{X}(i))^2 + o_p(1) \,. \end{aligned}$$

Par la loi des grands nombres et la condition (3.3), on obtient que, lorsque P tend vers l'infini,

$$\frac{1}{P}\sum_{j=1}^{P}\left(\frac{1}{P}U_j\right)^2 \xrightarrow{p} \mathbb{E}[\{F(\underline{X}^-) - \overline{G}(\overline{X})\}^2] > 0 \,. \tag{3.11}$$

Avec (3.10), $P^{-1}|U_i| \leq 2$, $i = 1, \ldots, P$, on a donc

$$\begin{aligned}|Y_i| = \frac{|U_i|}{\sqrt{\sum_{j=1}^{P}U_j^2}} &= \frac{1}{\sqrt{P}}\frac{P^{-1}|U_i|}{\sqrt{P^{-1}\sum_{j=1}^{P}(P^{-1}U_j)^2}} \\ &\leq \frac{1}{\sqrt{P}}\frac{2}{\sqrt{P^{-1}\sum_{j=1}^{P}(P^{-1}U_j)^2}} \,, \; i = 1, \ldots, P \,. \end{aligned} \tag{3.12}$$

En utilisant (3.11), le membre de droite de l'inégalité (3.12) tend vers 0 lorsque P rend vers l'infini, Y_i satisfait donc la troisième condition de (3.6), ce qui conclut cette démonstration.

Deuxième partie

MÉTHODES ROBUSTES DE TESTS D'HOMOGÉNÉITÉ ET DE CHANGEMENT POUR DONNÉES MULTIVARIÉES

Chapitre 4

Tests d'homogénéité et de détection de changements

Nous présentons dans ce chapitre un état de l'art des techniques existantes pour tester l'homogénéité entre plusieurs groupes d'observations ainsi que pour la détection de ruptures. Soit $X = (\mathbf{X}_1, \ldots, \mathbf{X}_n)$ un échantillon d'observations, éléments de $\mathbb{R}^K$ ($K \geq 1$). Dans le cadre des tests d'homogénéité entre deux échantillons, on considère que $(\mathbf{X}_1, \ldots, \mathbf{X}_{n_1})$ sont indépendants et identiquement distribués (i.i.d.) suivant une loi p et que $(\mathbf{X}_{n_1+1}, \ldots, \mathbf{X}_n)$ sont i.i.d. et suivent une loi q. On cherche alors à tester l'hypothèse nulle H_0 d'égalité des distributions : « $p = q$ » ; contre l'hypothèse alternative H_1 : « $p \neq q$ ». Lorsque l'on veut tester l'homogénéité entre plus de groupes, les n observations sont séparées en L échantillons : $(\mathbf{X}_1, \ldots, \mathbf{X}_{n_1})$; … ; $(\mathbf{X}_{n_{L-1}+1}, \ldots, \mathbf{X}_{n_L})$. On teste alors si l'ensemble des observations $(\mathbf{X}_1, \ldots, \mathbf{X}_n)$ est i.i.d., contre l'existence d'un groupe distribué selon une loi différente de celle des autres.

Une rupture est caractérisée par un changement durable dans les caractéristiques du signal étudié, autrement dit, la partie de signal située avant la position (ou l'instant) de rupture sera distribuée suivant une loi différente de celle lui succédant. Dans un cadre rétrospectif, on dispose de l'ensemble des n observations et le test de détection d'une rupture consiste alors à décider s'il y a effectivement une rupture puis, le cas échéant, à estimer sa position. L'approche la plus naturelle pour obtenir un tel test consiste à s'appuyer sur une statistique destinée à tester l'homogénéité. On considère alors que le nombre n_1 d'observations dans le premier échantillon est inconnu, puis on calcule la statistique de test d'homogénéité choisie pour toutes les valeurs possibles de n_1. En prenant leur maximum, on obtient la statistique pour le test de détection de rupture. Cette procédure s'étend aisément à la recherche de plusieurs ruptures dans un signal. Il suffit d'utiliser un test d'homogénéité entre plusieurs (plus de deux) groupes, et de maximiser la statistique sur l'ensemble des partitions possibles du signal en L sous-échantillons.

Un problème supplémentaire apparaît alors : celui de l'estimation du nombre effectif de ruptures.

Dans la section 4.1 nous présentons le cas classique du signal distribué suivant une loi gaussienne multivariée. On veut tester l'homogénéité d'un tel signal ou le partitionner en deux segments homogènes. Le premier cas est abordé avec une technique dérivée du maximum de vraisemblance qui produit la statistique du T^2 de Hotelling. La détection de rupture est effectuée en maximisant cette statistique. Les procédures résultantes sont souvent utilisées en pratique et servent de base à d'autres méthodes (par exemple la méthode du MMD présentée dans la section 4.2.1) ; les performances du test du T^2 de Hotelling et celles de la statistique pour la détection de rupture dérivée nous fournissent également des performances de référence pour évaluer les méthodes proposées. Nous nous concentrons ensuite sur des méthodes non-paramétriques, où l'on ne fait pas d'hypothèse sur la distribution sous-jacente des données. Nous présentons ainsi des méthodes non-paramétriques pour le test d'homogénéité de deux échantillons dans les espaces à grandes dimensions, ainsi que leur extension éventuelle à la détection d'une rupture, avec des méthodes à noyau dans la section 4.2 et des méthodes nécessitant la construction d'un arbre des plus proches voisins dans la section 4.3. Nous abordons ensuite dans la section 4.4 des méthodes robustes utilisant les rangs relatifs des observations, qui constituent les « fondations » des méthodes introduites dans les chapitres suivants. Enfin, en section 4.5, nous détaillons les méthodes utilisées pour l'estimation de changements multiples.

4.1 Test paramétrique de changement dans la moyenne

Nous considérons tout d'abord une méthode à laquelle on se compare généralement lorsque l'on veut tester l'homogénéité de deux échantillons dans un cadre multivarié : on suppose la forme de la distribution sous-jacente des observations connue, et plus particulièrement gaussienne multivariée. Cette problématique est présentée par exemple par Chen et Gupta (2000, chapitre 3). On se place dans le cas où l'on veut tester un changement dans le vecteur de moyenne alors que la matrice de covariance des données est inconnue mais reste identique, qu'il y ait un changement ou non.

Soit une séquence $X = (\mathbf{x}_1, \dots, \mathbf{x}_n)$ de n vecteurs aléatoires i.i.d. de $\mathbb{R}^K$ de paramètres respectifs $(\boldsymbol{\mu}_1, \Sigma), \dots, (\boldsymbol{\mu}_n, \Sigma)$. On veut ainsi tester l'hypothèse nulle

$$(H_0) : \boldsymbol{\mu}_1 = \cdots = \boldsymbol{\mu}_n$$

contre l'alternative

$$(H_1) : \boldsymbol{\mu}_1 = \cdots = \boldsymbol{\mu}_{n_1} \neq \boldsymbol{\mu}_{n_1+1} = \cdots = \boldsymbol{\mu}_n.$$

On examine deux cas, suivant que la valeur de n_1 est connue ou pas. Lorsque n_1 est fixé, le test d'hypothèse correspond à un test d'homogénéité entre deux échantillons. Si n_1 est inconnue, on a affaire à un problème de détection de rupture.

1er cas : n_1 fixé (T^2 de Hotelling) À n_1 fixé, on teste l'homogénéité de deux échantillons de tailles respectives n_1 et $n - n_1$. Dans les modèles paramétriques, ce problème est résolu en calculant le rapport de la vraisemblance sous les deux hypothèses.

Soit δ_{n_1} la différence normalisée entre les deux échantillons

$$\delta_{n_1} = \sqrt{\frac{n_1(n-n_1)}{n}}(\overline{\mathbf{X}}_{n_1} - \overline{\mathbf{X}}_{n-n_1}),$$

avec $\overline{\mathbf{X}}_{n_1} = n_1^{-1}\sum_{i=1}^{n_1}\mathbf{X}_i$ et $\overline{\mathbf{X}}_{n-n_1} = (n-n_1)^{-1}\sum_{i=n_1+1}^{n}\mathbf{X}_i$, et

$$W_{n_1} = \frac{1}{n-2}[\sum_{i=1}^{n_1}(\mathbf{X}_i - \overline{\mathbf{X}}_{n_1})(\mathbf{X}_i - \overline{\mathbf{X}}_{n_1})' + \sum_{i=n_1+1}^{n}(\mathbf{X}_i - \overline{\mathbf{X}}_{n-n_1})(\mathbf{X}_i - \overline{\mathbf{X}}_{n-n_1})'],$$

l'estimateur de la covariance de l'échantillon regroupé. La statistique du T^2 de Hotelling (1931) qui s'écrit

$$T^2_{n_1} = \delta'_{n_1} W^{-1}_{n_1} \delta_{n_1} \ , \tag{4.1}$$

est alors la statistique utilisée pour le test de détection de changement dans la moyenne dans le cas gaussien.

2nd cas : n_1 inconnu (test de changement) Lorsque n_1 est inconnu dans la formulation de l'hypothèse alternative (H_1) ci-dessus, on se ramène alors au problème de détection de changement. La statistique proposée pour un tel test est alors

$$\max_{1\leq n_1\leq n-1} T^2_{n_1} \ , \tag{4.2}$$

l'estimateur de la position du changement est alors $\hat{n}_1 = \text{argmax}_{1\leq n_1\leq n-1} T^2_{n_1}$.

Déterminer la distribution de la statistique (4.2) sous l'hypothèse nulle permettrait de pouvoir fixer un seuil de décision dépendant par exemple d'un taux de fausses alarmes attendu. Il n'existe pas à notre connaissance de forme exacte de cette distribution. Srivastava et Worsley (1986) proposent cependant une approximation de la distribution sous H_0 de $\max_{1\leq n_1\leq n} S_{n_1}$ où

$$S_{n_1} = \frac{T^2_{n_1}}{n-2+T^2_{n_1}} \ ;$$

ce maximum est atteint à la même position que celui de $T^2_{n_1}$. Ils utilisent une procédure de tests multiples pour effectuer cette approximation. Ils améliorent un

premier résultat, très conservatif, qui avait été donné par Vostrikova (1981) et qui fait appel à la correction de Bonferroni.

$$\mathbb{P}(S_{\hat{n}_1} > c) \leq 1 - G_{K,\nu}(c) + q_1 \sum_{r=1}^{n-2} t_r - q_2 \sum_{r=1}^{n-2} t_r^3 \,,$$

où $G_{K,\nu}$ est la fonction de répartition d'une loi bêta de paramètres $K/2$ et $\nu/2$, avec $\nu = (n - K - 1)$, q_1 et q_2 sont des termes multiplicatifs qui dépendent de c, K et n, et $t_r = (1 - \rho_r)^{1/2}$ où ρ_r est la corrélation entre S_r et S_{r+1}. Le résultat de Srivastava et Worsley (1986) prend en compte les fortes dépendances entre les valeurs de la statistique de Hotelling calculée à deux points différents, par exemple $T^2_{n_1}$ et $T^2_{n_1+1}$. Les auteurs indiquent que leur approximation donne aussi des résultats trop conservatifs lorsque la taille de l'échantillon dépasse la quarantaine.

4.2 Méthodes à noyaux

Les méthodes à noyaux (*kernel methods*) sont une classe de méthodes non-paramétriques applicables à des données en grande dimension. Les méthodes à noyau ont été popularisées grâce à leur application aux séparateurs à vaste marge (SVM, ou en anglais *Support Vector Machine*), un algorithme de classification supervisée binaire. Dans le cas où les données sont linéairement séparables, l'algorithme du SVM (Boser *et al.*, 1992) consiste à chercher l'hyperplan séparant les exemples positifs des exemples négatifs qui maximise la marge, c'est-à-dire la distance euclidienne entre cet hyperplan et le point le plus proche de l'ensemble d'apprentissage. Sans entrer dans les détails de l'algorithme permettant de trouver la solution (qui revient en fait à résoudre un problème d'optimisation quadratique), celle-ci peut s'écrire sous la forme $f(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle + b$, où $w, x \in \mathbb{R}^K$, $b \in \mathbb{R}$, $\langle \cdot, \cdot \rangle$ désigne le produit scalaire usuel et K est la dimension de l'espace d'entrée. Le vecteur $\mathbf{w}$ définissant l'hyperplan de séparation s'écrit comme une somme pondérée des vecteurs d'apprentissage $\mathbf{x}_i$: $\mathbf{w} = \sum_i \alpha_i \mathbf{x}_i$, on a ainsi

$$f(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle + b = \sum_i \alpha_i \langle \mathbf{x}_i, \mathbf{x} \rangle + b \,. \tag{4.3}$$

Les solutions de l'équation $f(\mathbf{x}) = 0$ correspondent alors à l'hyperplan de séparation, et le signe de f indique la classe à laquelle appartient un point x.

Dans les problèmes plus complexes où les données des deux classes ne sont pas séparables linéairement, l'idée est d'appliquer une transformation non linéaire φ à ces données pour les transposer dans un espace de plus grande dimension $\mathcal{H}$, appelé espace de Hilbert à noyau autoreproduisant (*Reproducing Kernel Hilbert Space*, ou RKHS) dans lesquelles les images par φ des données d'entrée sont linéairement séparables et d'y appliquer les algorithmes classiques.

Dans l'équation (4.3), l'écriture de f ne fait intervenir que le produit scalaire de $\mathbf{x}$ avec les données d'apprentissage $(\mathbf{x}_i)_{1 \leq i \leq n}$. C'est cette idée qui est utilisée

dans les méthodes à noyau. L'astuce du noyau (*kernel trick*), introduite par Aizerman *et al.* (1964), consiste à utiliser un noyau k, qui est une fonction réelle à deux variables de l'espace d'entrée, pour lequel $k(\mathbf{x}, \mathbf{z})$ est le résultat du produit scalaire $\langle \varphi(\mathbf{x}), \varphi(\mathbf{z}) \rangle$ dans l'espace $\mathcal{H}$. L'utilisation du noyau k permet de se dispenser du calcul explicite des $\varphi(x_i)$. L'utilisation de fonctions noyau particulières suffit à définir implicitement une application φ vers un espace d'arrivée à grande dimension (possiblement infinie). Citons quelques noyaux particuliers, pour $\mathbf{x}$ et $\mathbf{z}$ éléments de l'espace d'entrée $\mathcal{X} \subset \mathbb{R}^K$:

- linéaire : $k(\mathbf{x}, \mathbf{z}) = \mathbf{x}'\mathbf{z}$, qui correspond au produit scalaire canonique de $\mathbb{R}^K$;
- polynomial : $k(\mathbf{x}, \mathbf{z}) = (\mathbf{x}'\mathbf{z} + b)^d$, pour b réel et d entier ;
- gaussien : $k(\mathbf{x}, \mathbf{z}) = \exp(-\|\mathbf{x} - \mathbf{z}\|^2 / 2\sigma^2), \sigma \in \mathbb{R}$.

Ce dernier noyau est généralement le plus utilisé pour des données de type numérique. Dans ce cas, la largeur de bande du noyau σ constitue un hyper-paramètre à régler ; les performances des méthodes à noyaux sont sensibles à ce paramètre. Des noyaux spécifiques existent aussi pour des structures de données non numériques : graphes, arbres, chaînes de caractères. Un panorama des méthodes à noyaux est accessible dans les ouvrages de Schölkopf et Smola (2002) ou de Shawe-Taylor et Cristianini (2004).

De nombreux algorithmes classiques, linéaires, peuvent de la même manière être étendus à des espaces à grande dimension lorsque leur écriture est une combinaison de produits scalaires des variables d'entrée. Certains d'entre eux permettent de traiter le problème du test d'homogénéité et de détection de changement.

4.2.1 MMD

Gretton *et al.* (2007) ont proposé une méthode à noyaux pour le problème à deux échantillons. La statistique proposée, appelée divergence maximale moyenne (*Maximum Mean Discrepancy*, ou MMD) donne une mesure de dissimilarité entre deux distributions calculée à partir d'échantillons tirés de celles-ci. L'idée est similaire à celle utilisée dans le test de Hotelling, c'est-à-dire calculer une différence entre les moyennes des deux échantillons, mais en utilisant l'astuce du noyau pour calculer la moyenne lorsque les points de données sont vus comme des éléments d'un RKHS. Ainsi, étant donnés deux ensembles d'observations $\mathbf{X}_1, \ldots, \mathbf{X}_{n_1}$ et $\mathbf{X}_{n_1+1}, \ldots, \mathbf{X}_n$ éléments de $\mathbb{R}^K$ et distribués selon les lois p et q, l'élément moyenne μ_p associé à p est l'élément du RKHS $\mathcal{H}$ défini par $\mu_p = \mathbb{E}_p[\varphi(X)]$. Le MMD entre les distributions p et q s'écrit

$$\mathrm{MMD}(p, q) = \|\mu_p - \mu_q\|^2_{\mathcal{H}} ;$$

cette valeur est nulle pour la classe de noyaux dit caractéristiques (c'est le cas pour le noyau gaussien) si et seulement si $p = q$. En utilisant les estimateurs empiriques $\hat{\mu}_p = 1/m \sum_{i=1}^{n_1} \varphi(\mathbf{X}_i)$ de μ_p et $\hat{\mu}_q = 1/(n-n_1) \sum_{i=n_1+1}^{n} \varphi(\mathbf{X}_i)$ de μ_q ainsi que le fait que $k(x, x') = \langle \varphi(x), \varphi(x') \rangle$, un estimateur empirique du MMD s'écrit

$$T_{\mathrm{MMD}}(p, q) = \frac{1}{n_1^2} \sum_{i=1}^{n_1} \sum_{j=1}^{n_1} k(\mathbf{X}_i, \mathbf{X}_j) - \frac{2}{n_1(n-n_1)} \sum_{i=1}^{n_1} \sum_{j=n_1+1}^{n} k(\mathbf{X}_i, \mathbf{X}_j) + \frac{1}{(n-n_1)^2} \sum_{i=n_1}^{n} \sum_{j=1}^{m} k(\mathbf{X}_i, \mathbf{X}_j) \ .$$

Cette statistique a une complexité algorithmique quadratique en le nombre d'échantillons, et si la forme exacte de la distribution de la statistique n'est pas connue, les auteurs proposent quelques méthodes d'approximation : une première qui l'approche par une courbe de Pearson à partir des premiers moments empiriques de la distribution de la statistique obtenue ; une deuxième avec une méthode de *bootstrap* ; enfin par le calcul des valeurs propres de la matrice de Gram des données (Gretton *et al.*, 2009).

4.2.2 Analyse discriminante de Fisher à noyaux

Harchaoui *et al.* (2008) proposent un autre test d'homogénéité entre deux échantillons qui est une variante du MMD, le KFDA (*Kernel Fisher Discriminant Analysis*), qui intègre la structure de covariance des données à la statistique de test. Le KFDA est une extension du discriminant de Fisher à un espace de Hilbert à noyau autoreproduisant en utilisant l'astuce du noyau. Dans le cas linéaire, l'idée est de trouver la direction de projection des données pour laquelle on maximise la séparation entre les moyennes des deux échantillons testés tout en minimisant la variance des données de chaque échantillon autour de leur moyenne.

Dans un RKHS, on désigne par $\hat{\Sigma}_p$ l'estimateur empirique de l'opérateur de covariance de l'échantillon X distribué selon p. Si on note $\hat{\Sigma}_W = n_1/n\hat{\Sigma}_p + (n - n_1)/n\hat{\Sigma}_q$ l'estimateur de la matrice de covariance de l'échantillon groupé, alors la statistique du KFDA s'écrit

$$T_{\mathrm{KFDA}}(X, Y) = \frac{n_1(n-n_1)}{n} \left\| (\hat{\Sigma}_W + \gamma I)^{-1/2} (\hat{\mu}_p - \hat{\mu}_q) \right\|_{\mathcal{H}}^2 \ ,$$

où le réel positif γ est un terme de régularisation.

Comparé au MMD, le KFDA obtient généralement de meilleurs performances de détection. Le temps de calcul est cependant plus élevé (inversion de la matrice de covariance) et le terme de régularisation γ introduit un hyper-paramètre supplémentaire à régler.

Harchaoui *et al.* (2009a) étendent la procédure du KFDA au problème de détection rétrospectif d'un changement dans une fenêtre d'observations en maximisant la statistique du KFDA sur l'ensemble des positions possibles de la rupture.

4.2.3 SVM à une classe

Utilisant la même idée du discriminant de Fisher dans un RKHS, Désobry *et al.* (2005) proposent une méthode alternative à celle de Harchaoui *et al.* (2008) comme mesure de dissimilarité entre deux distributions ; leur méthode utilise un SVM à une classe (Scholkopf *et al.*, 2001) pour chacun des deux échantillons à comparer et les auteurs en proposent une interprétation géométrique. L'algorithme du SVM à une classe consiste (Shawe-Taylor et Cristianini, 2004, section 7.1), dans un RKHS $\mathcal{H}$ induit par un noyau k, à calculer l'hyperplan dans $\mathcal{H}$ séparant les échantillons de l'origine avec une marge maximale ρ.

Le noyau k est choisi normalisé de sorte que pour tout élément x de l'espace d'entrée, $k(x, x) = 1$; dans $\mathcal{H}$ les images $\varphi(x)$ se situent donc sur une hypersphère unitaire.

Soit $X = (\mathbf{X}_1, \ldots, \mathbf{X}_n)$ et $Y = (\mathbf{Y}_1, \ldots, \mathbf{Y}_m)$ les deux échantillons dont on veut tester l'homogénéité. On calcule l'hyperplan $\mathcal{W}_X$ correspondant à l'échantillon X. $\mathcal{W}_X$ est caractérisé par sa marge ρ_X et les coefficients $\alpha_{X,i}$ ($1 \leq i \leq n$), qui sont les solutions d'un programme d'optimisation. L'hypersphère unitaire intersecte le vecteur normal à $\mathcal{W}_X$ en un point $\mathbf{c_X}$; soit $\mathbf{p_X}$ un point appartenant à l'intersection entre l'hypersphère unitaire et $\mathcal{W}_X$. On définit de même pour l'échantillon Y les points $\mathbf{p_Y}$ et $\mathbf{c_Y}$.

La statistique T_{DDD} proposée par Désobry *et al.* (2005) se calcule ainsi :

$$T_{\mathrm{DDD}}(X, Y) = \frac{\widehat{\mathbf{c_X c_Y}}}{\widehat{\mathbf{c_X p_X}} + \widehat{\mathbf{c_Y p_Y}}}$$

où $\widehat{\mathbf{ab}} = \arccos(\langle a, b \rangle_{\mathcal{H}})$.

La statistique a le même comportement que le ratio de Fisher : l'arc $\widehat{\mathbf{c_X c_Y}}$ est une mesure de la distance entre les supports des deux échantillons, alors que $\widehat{\mathbf{c_X p_X}}$ mesure la dispersion de l'échantillon X.

Les auteurs utilisent cette mesure de dissimilarité pour concevoir une méthode de détection de changement, appelée *KCD* (*Kernel Change Detection*) : l'algorithme est appliqué à une fenêtre glissante de données, les deux SVM à une classe sont calculés sur les parties droite et gauche de cette fenêtre. Aucune formule donnant la distribution de la statistique n'est proposée, aussi ne peut-on calculer un seuil de détection de manière raisonnée (en fonction de la proportion de fausses alarmes attendues par exemple). De plus, des expériences ont montré (Harchaoui *et al.*, 2009a) que cette méthode donnait des résultats inférieurs à ceux obtenus par d'autres méthodes à noyau.

Notons par ailleurs que le SVM à une classe a été souvent utilisé dans le contexte de la détection d'anomalies (notamment réseau) : le SVM est entraîné sur une portion de données correspondant à un comportement « normal » des

données, et de nouveaux points sont considérés comme étant des anomalies s'ils n'apparaissent pas du côté de l'hyperplan contenant les données d'apprentissage.

4.3 Méthodes à arbres des plus proches voisins

Les algorithmes présentés dans cette section utilisent les relations de distance entre les différents points pour constituer des tests d'homogénéité entre deux échantillons multivariés. Ils ont pour propriété d'être non-paramétriques (sauf pour quelques variantes) et d'être appropriés pour une grande variété d'alternatives (par exemple différence dans la moyenne ou échelles différentes).

4.3.1 Généralisations multivariées du test de Wald-Wolfowitz

Dans le cas unidimensionnel un test d'homogénéité à deux échantillons est le test de Wald et Wolfowitz (1940). Celui-ci consiste à trier les valeurs des observations regroupées des deux échantillons testés dans l'ordre croissant, puis à compter le nombre de « *runs* », le *run* étant une séquence d'observations provenant du même échantillon. On peut voir intuitivement que dans le cas où les deux échantillons diffèrent beaucoup dans leurs moyennes, on obtient de longues séquences et ce nombre de *runs* est petit, ce qui indique que l'hypothèse nulle d'égalité des distributions peut être rejetée. De par l'absence d'une relation d'ordre naturelle dans des observations multivariées, on peut comprendre qu'il est difficile d'obtenir une généralisation de ce test à des dimensions supérieures ou égales à deux.

Friedman et Rafsky (1979) proposent quelques généralisations qui utilisent le concept d'arbre de recouvrement de poids minimal (*minimal spanning tree*, abrégé en MST). Ces méthodes commencent par construire l'arbre de recouvrement de poids minimal sur les données regroupées, c'est-à-dire le graphe complet (aucun point/nœud n'est isolé des autres) qui minimise la somme des poids assignés aux arêtes du graphe, ces poids étant fixés comme étant la distance (euclidienne ou autre) entre les deux nœuds reliés.

Un premier test (« *runs* ») consiste à construire le MST de l'échantillon regroupé, puis à supprimer les arêtes existantes entre les nœuds provenant de groupes différents. La statistique de test est alors le nombre de sous arbres disjoints qui résultent de cet élagage (c'est aussi le nombre d'arêtes supprimées plus un). Les auteurs calculent les deux premiers moments de cette statistique et montrent sa normalité asymptotique (la distribution peut aussi être calculée de manière exacte combinatoirement pour un échantillon de petite taille).

Un second test (appelé *de Smirnov multivarié*) consiste à créer une relation d'ordre entre les différents nœuds de l'arbre. Le nœud ayant la plus grande excentricité, c'est-à-dire le nœud à partir duquel on peut construire un chemin de longueur maximale, est choisi comme racine de l'arbre et est étiqueté comme étant le premier nœud. Les nœuds suivants sont étiquetés successivement lorsqu'ils sont

traversés pour la première fois lors d'un parcours en profondeur de l'arbre, les branches les plus courtes étant traversées en premier (*depth first search with height directed preorder*). Enfin, on compte le nombre de « *runs* » de la même manière que pour le test de Wald-Wolfowitz uni-dimensionnel, cette fois ci en prenant en compte cette nouvelle relation d'ordre utilisant l'ordre de parcours des nœuds. Une variante (*Smirnov radial*) consiste à classer les nœuds en choisissant pour racine le nœud central puis à parcourir l'arbre en largeur.

Le calcul du MST peut se faire rapidement avec les algorithmes classiques de Prim (1957) ou Kruskal (1956)[1] en $\mathcal{O}(n^2)$ opérations ; de récents algorithmes (Chazelle, 2000) obtiennent même une complexité linéaire. Les étapes suivantes s'exécutent en un temps linéaire.

Ces tests sont en théorie puissants contre des alternatives générales. En pratique le test *Smirnov multivarié* est efficace par rapport au test optimal dans le cas gaussien pour de petites dimensions dans le cas des changements dans la moyenne. Le test *Smirnov radial* est quant à lui puissant pour des changements d'échelle. En grande dimension (supérieure à 20) la puissance de ces tests diminue rapidement : lorsqu'on augmente la dimension, les points ont tendance à se rapprocher et les rangs relatifs portent moins d'information. À l'inverse, le test *runs* obtient de mauvaises performances en faible dimension, ce qui est cohérent avec la faible puissance du test de Wald-Wolfowitz en dimension un. Cependant sa puissance augmente avec la dimension, dans le cas des changements dans la moyenne ; il est cependant peu puissant pour les changements d'échelle.

4.3.2 k plus proches voisins

Les méthodes utilisant un algorithme des plus proches voisins (*nearest neighbours*) sont une autre classe de méthodes pour le test d'homogénéité à deux échantillons. Schilling (1986) ou Henze (1988) en font la description. Prenons une norme $\|\cdot\|$ dans l'espace des observations $\mathbb{R}^K$; le k^{e} plus proche voisin d'un point $\mathbf{X}_i$ est le point $\mathbf{X}_j$ tel qu'il existe exactement $k-1$ points $\mathbf{X}_{j'}$ plus proches de $\mathbf{X}_i$ que $\mathbf{X}_j$, c'est-à-dire tels que $\|\mathbf{X}_i - \mathbf{X}_{j'}\| < \|\mathbf{X}_i - \mathbf{X}_j\|$ (les cas d'égalité se produisent avec une probabilité nulle si la distribution des observations est continue). Notons $\mathrm{NN}_i(r)$ le r^{e} plus proche voisin du point $\mathbf{X}_i$ et $I_i(r) = 1$ si $\mathrm{NN}_i(r)$ appartient au même échantillon que $\mathbf{X}_i$ et 0 sinon. Une statistique des plus proches voisins s'écrit de manière générale :

$$T_{\mathrm{NN,k}} = \frac{1}{nk} \sum_{i=1}^{n} \sum_{r=1}^{k} w_i(r) I_i(r) \ ,$$

où $w_i(r)$ est une pondération qui peut dépendre de r et/ou de la position du point $\mathbf{X}_i$.

[1] Joseph Kruskal, frère de William

Dans le cas non pondéré, c'est-à-dire quand les w valent 1, la statistique correspond à la proportion de tous les k plus proches voisins mutuels qui appartiennent au même échantillon. Centrée et avec la normalisation adéquate cette statistique converge vers une loi normale. Schilling (1986) propose aussi une pondération fixe dépendant de r, mais ne constate pas d'amélioration notable par rapport à la version non pondérée ; l'auteur propose aussi une pondération continue dont la valeur dépend de la forme de la distribution supposée des données sous les hypothèses nulle et alternative (et donc du type d'alternative). Cette dernière pondération permet d'obtenir une puissance proche de celle d'un test paramétrique utilisant le maximum de vraisemblance mais le test perd la propriété non-paramétrique. De manière générale, les tests utilisant la méthode des plus proches voisins sont sensibles à la dimension, les résultats asymptotiques donnés sur la distribution de la statistique n'étant plus précis lorsque la dimension devient élevée (supérieure à la dizaine). La complexité algorithmique est cependant assez faible, le calcul de tous les plus proches voisins pouvant se faire en $\mathcal{O}(kn\log(n))$ opérations en utilisant des algorithmes appropriés, comme par exemple celui proposé par Friedman *et al.* (1977), utilisant une structure de données de type *arbre kd*.

À notre connaissance, il n'existe pas de généralisation de ces algorithmes à la détection rétrospective de rupture. On peut en revanche trouver des algorithmes utilisant les algorithmes des plus proches voisins pour la détection d'anomalie. Les k plus proches voisins sont utilisés comme estimateurs de densité ; l'algorithme est appliqué sur les données correspondant au comportement normal. Une anomalie est déclarée lorsqu'un nouveau point se situe hors du support estimé de la distribution. Hero III (2007) ou Zhao et Saligrama (2009) proposent des méthodes utilisant ce principe.

4.4 Méthodes de rang

Nous présentons dans cette section les algorithmes classiques utilisant les rangs pour tester l'homogénéité de deux ou plusieurs groupes de données unidimensionnelles. Le test de Mann-Whitney/Wilcoxon est d'abord présenté, une version modifiée pour prendre en compte la présence de censure a déjà été utilisée pour élaborer le test utilisé pour la détection d'anomalies réseau, au chapitre 3. Ensuite on s'intéresse au test de Kruskal et Wallis qui examine la distribution de plusieurs échantillons ; enfin le test de Wei et Lachin est utilisé pour tester l'homogénéité en présence de censure dans le cas multivarié.

4.4.1 Test de rang de Mann-Whitney/Wilcoxon

Le test de Mann-Whitney/Wilcoxon est un test statistique non-paramétrique permettant de tester si deux échantillons de valeurs scalaires ou ordinales ont la

même loi. Soit ainsi $X_1, \ldots, X_n$ un échantillon de variables aléatoires réelles que l'on subdivise en deux sous-échantillons $X_1, \ldots, X_{n_1}$ et $X_{n_1+1}, \ldots, X_n$ de tailles respectives n_1 et $n - n_1$. Pour tout i, $1 \leq i \leq n$, on note R_i le rang de X_i dans l'échantillon groupé, à savoir $R_i = \sum_{j=1}^{n} \mathbf{1}(X_j \leq X_i)$. La statistique W de Wilcoxon (1945) s'écrit (Lehmann (1975) ou van der Vaart (1998, chapitre 13)) :

$$W_{n_1} = \sum_{j=1}^{n_1} R_j \, . \tag{4.4}$$

La statistique $W_{n_1}^{\mathrm{MW}}$ de Mann et Whitney (1947) qui s'écrit

$$W_{n_1}^{\mathrm{MW}} = \sum_{i=1}^{n_1} \sum_{j=n_1+1}^{n} \mathbf{1}\left(X_i \leq X_j\right) \tag{4.5}$$

est une statistique équivalente à la statistique de Wilcoxon ; en effet,

$$W_{n_1}^{\mathrm{MW}} = W_{n_1} - \frac{1}{2} n_1 (n_1 + 1) \, .$$

L'hypothèse nulle d'homogénéité des deux échantillons est rejetée pour d'« assez grandes » valeurs de W. Le seuil de décision peut être calculé exactement[2] ou en utilisant une approximation de la distribution asymptotique de W. En effet, on peut montrer (van der Vaart, 1998, corollaire 13.8) que

$$\frac{W_{n_1} - \mathbb{E}[W_{n_1}]}{\sqrt{\operatorname{Var} W_{n_1}}} \xrightarrow{d} \mathcal{N}(0, 1)$$

où

$$\mathbb{E}[W_{n_1}] = \frac{1}{2} n_1 (n + 1) \quad \text{et} \quad \operatorname{Var} W_{n_1} = \frac{1}{12} n_1 (n - n_1)(n + 1) \, .$$

Le test de Mann-Whitney/Wilcoxon est consistant pour les alternatives pour lesquelles $\mathbb{P}(X \leq Y) \neq 1/2$, où X et Y sont des variables aléatoires i.i.d. dont on veut tester l'homogénéité. Par exemple pour un changement dans la moyenne, le test est consistant. En effet, soit $\mu \neq 0$; $\mathbb{P}(X \leq X + \mu) = \mathbb{P}(0 \leq \mu) \neq 1/2$. De même pour un changement d'échelle (sans changement de moyenne μ) lorsque la variable aléatoire X est positive : soit $\sigma > 1$, $\mathbb{P}(\mu + X \leq \mu + \sigma X) = \mathbb{P}((\sigma - 1)X \geq 0) = \mathbb{P}(X \geq 0) = 1 \neq 1/2$ (le cas $\sigma < 1$ se traite de la même manière).

Le test de Wilcoxon est une alternative au test de Student qui permet de tester l'égalité des moyennes de variables aléatoires gaussiennes unidimensionnelles, mais impose moins d'hypothèses que ce dernier. En effet, le test de Student suppose la normalité des variables aléatoires testées ainsi que l'égalité de leur variance. Lorsque ces conditions sont réunies, étant dérivé à partir du rapport de

[2] Cela peut être fait en énumérant l'ensemble des valeurs de la statistique W pour les configurations possibles de $1, \ldots, n$ dans deux groupes de n_1 et $n - n_1$ éléments. C'est une opération possible pour de petites valeurs de n et n_1 mais le cardinal de l'ensemble de ces permutations étant $\binom{n}{n_1}$, cela devient impossible lorsque ces quantités augmentent.

vraisemblance, le test est uniformément plus puissant. Le test de Wilcoxon et le calcul de niveau de significativité associé ont l'avantage de ne pas dépendre de la distribution des données sous-jacentes. Le test est par ailleurs plus robuste que le test de Student aux valeurs aberrantes (une comparaison expérimentale de méthodes de rangs à celles utilisant sur le maximum de vraisemblance sous l'hypothèse gaussienne est réalisé en section 6.3.2). Il a aussi été montré (Lehmann, 1975, section 2.4) que sous l'hypothèse gaussienne, le test de Wilcoxon a une bonne efficacité relative par rapport test de Student (que l'on définit comme le rapport entre la taille de l'échantillon nécessaire pour que le test de Wilcoxon obtienne la même puissance que le test de Student sous la même alternative) de 0,955 alors qu'elle devient supérieure à 1 sous d'autres distributions, notamment pour celles à queues lourdes. Ainsi lorsque les données peuvent possiblement dévier de leur distribution présupposée, il devient alors avantageux d'utiliser le test de Wilcoxon plutôt que le test paramétrique, la perte de puissance étant assez minime dans le cas gaussien.

4.4.2 Test de Kruskal-Wallis

Le test de Mann-Whitney/Wilcoxon permet de tester si les distributions de deux échantillons univariés diffèrent. Le test suivant est une variante qui s'intéresse à la distribution d'un nombre plus grand d'échantillons. Supposons que l'on ait L groupes de données à valeurs réelles $X_1, \ldots, X_{n_1}\,; X_{n_1+1}, \ldots, X_{n_2}\,; \ldots; X_{n_{L-1}+1}, \ldots, X_{n_L}$. On convient que $n_0 = 0$ et $n_L = n$. On veut déterminer si ces L groupes de données proviennent de la même distribution. Le test de Kruskal et Wallis (1952) permet de répondre à cette question, à l'aide d'une procédure basée sur les rangs. Une méthode en rapport avec le test de Kruskal-Wallis est l'analyse de la variance (ANOVA) qui s'applique lorsque les observations sont distribuées selon une loi gaussienne.

Soit R_i le rang de X_i parmi l'ensemble des observations et notons

$$\bar{R}_\ell = \frac{1}{n_{\ell+1} - n_\ell} \sum_{i=n_\ell+1}^{n_{\ell+1}} R_i$$

le rang moyen du ℓ^{e} échantillon.

La statistique de Kruskal-Wallis s'écrit

$$H = \frac{12}{n(n+1)} \sum_{\ell=0}^{L-1} (n_{\ell+1} - n_\ell) \left(\bar{R}_\ell - \frac{n+1}{2} \right)^2 . \tag{4.6}$$

Une valeur « assez grande » de H indique que l'on rejette l'hypothèse nulle selon laquelle les L groupes d'observations proviennent d'une même distribution. Rappelons que les rangs R_i ($i = 1, \ldots, n$) prennent pour valeurs l'ensemble des entiers de 1 à n, et ont donc pour valeur moyenne empirique $(n+1)/2$; la statistique

H prend donc de grandes valeurs lorsqu'au moins un des termes de la somme est grand, c'est-à-dire lorsque le rang moyen $\bar{R}_\ell$ au sein d'un groupe s'éloigne suffisamment du rang moyen total $(n+1)/2$.

On peut montrer (van der Vaart, 1998) que sous certaines hypothèses (continuité de la fonction de répartition de X_1 et convergence de $(n_{i+1} - n_i)/n$ vers $t_i \in (0,1)$ lorsque $n \to \infty$), la distribution de la statistique H tend asymptotiquement vers une loi du χ^2 à $L-1$ degrés de liberté sous l'hypothèse nulle, ce qui permet d'élaborer un test à un niveau de fausse alarme α donné.

4.4.3 Statistique de Wei et Lachin

Wei et Lachin (1984) ont proposé une méthode de rangs non-paramétrique pour tester l'égalité de deux distributions multivariées. Leur méthode s'applique aux observations censurées et généralise au cas multidimensionnel le test d'homogénéité de Gehan (1965) qui s'appliquait aux données unidimensionnelles censurées. La statistique de Wei et Lachin est une combinaison de statistiques calculées dans chacune des dimensions. Soit $(\mathbf{X}_1, \ldots, \mathbf{X}_n)$ une série de n observations de $\mathbb{R}^K$, avec $\mathbf{X}_j = (X_{j,1}, \ldots, X_{j,K})'$. On veut tester l'homogénéité des deux échantillons $\mathbf{X}_1, \ldots, \mathbf{X}_{n_1}$ et $\mathbf{X}_{n_1+1}, \ldots, \mathbf{X}_n$.

La variable $X_{j,k}$ n'est pas forcément connue, on considère qu'on observe une valeur censurée, à savoir le couple $(\tilde{X}_{j,k}, \delta_{X,j,k})$ où $\tilde{X}_{j,k} = \min(X_{j,k}, c_{j,k})$, $\delta_{X,j,k} = \mathbf{1}(X_{j,k} = \tilde{X}_{j,k})$ et les $c_{j,k}$ sont des valeurs de censure indépendants des $X_{j,k}$ correspondants.

Pour la dimension k, la statistique marginale s'écrit[3]

$$T_{n,k} = \frac{1}{n^{3/2}} \left[\sum_{j=1}^{n_1} \delta_{X,j,k} \sum_{m=n_1+1}^{n} \mathbf{1}(\tilde{X}_{m,k} \geq \tilde{X}_{j,k}) - \sum_{j=n_1+1}^{n} \delta_{X,j,k} \sum_{m=1}^{n_1} \mathbf{1}(\tilde{X}_{m,k} \geq \tilde{X}_{j,k}) \right] .$$

Les auteurs montrent que sous certaines conditions, le vecteur $(T_{n,1}, \ldots, T_{n,K})'$ converge en distribution vers un vecteur aléatoire gaussien de moyenne nulle et de covariance Σ, ce qui leur permet de proposer

$$(T_{n,1}, \ldots, T_{n,K})' \hat{\Sigma}^{-1} (T_{n,1}, \ldots, T_{n,K}),$$

où $\hat{\Sigma}$ est un estimateur de Σ, comme statistique pour le test d'homogénéité entre les deux échantillons X et Y.

On montre dans le chapitre 5 par le moyen de simulations que la statistique de Wei et Lachin (1984) est erronée, dans le sens où lorsque la taille des deux échantillons testés est déséquilibrée, la distribution de la statistique ne correspond pas

[3] Cette écriture est un cas particulier donné par les auteurs d'une forme plus générale qui comprend une certaine pondération dans chacun des termes des sommes apparaissant dans l'expression de $T_{n,k}$. Suivant les valeurs de ces poids on obtient des statistiques qui étendent celles de Gehan (présentée ici) ou du *log-rank*

à la loi attendue. Nous avons donc cherché à corriger ce défaut en proposant une matrice de renormalisation différente et nous avons étendu le test d'homogénéité à un test de détection de ruptures.

4.5 Stratégies pour l'estimation de changements multiples

Nous nous intéressons dans cette section au problème de l'estimation de plusieurs changements dans la moyenne d'une série d'observations ainsi qu'au problème de l'estimation du nombre de ruptures. Deux familles d'approches sont utilisées dans la littérature, d'une part les approches dites *locales*, qui consistent à appliquer itérativement un algorithme de détection d'une unique rupture à plusieurs parties restreintes des données ; et d'autre part les approches *globales*, où le problème d'estimation des changements revient à un problème d'optimisation sur l'ensemble des observations.

4.5.1 Méthodes « locales »

Parmi les méthodes locales, citons tout d'abord la *segmentation binaire* (un terme plus approprié serait une segmentation *hiérarchique*), mentionnée par exemple par Vostrikova (1981). Les ruptures sont détectées récursivement en appliquant un test de détection d'une rupture dans les sous-segments déterminés à l'étape précédente. L'algorithme 1 présente sous forme de pseudo-code la méthode de segmentation binaire.

Algorithme 1

Entrées: X : données ; α : seuil de détection ; `début` = 1 ; `fin` = N
Sorties: L : liste des positions des ruptures

```
1: fonction SegmentationBinaire(X, début, fin, α)
2:     pvaleur, position = DetectionRupture(X, début, fin)
3:     si pvaleur < α alors
4:         ajouter position à la liste L
5:         SegmentationBinaire(X, début, position)
6:         SegmentationBinaire(X, position, fin)
7:     fin si
8: fin fonction
```

L'algorithme de segmentation binaire a pour principal avantage de ne nécessiter que de savoir traiter le cas de la détection d'une seule rupture. Le nombre de ruptures est automatiquement estimé suivant le seuil de détection α que l'on impose à l'algorithme de détection choisi. Cette méthode est utilisée par exemple par Chen et Gupta (2000) ou Srivastava et Worsley (1986).

Une autre stratégie consiste à calculer une mesure de dissimilarité entre les observations de deux sous-parties d'une fenêtre glissante. C'est une stratégie adoptée

par exemple par Harchaoui *et al.* (2009b) qui utilise l'analyse du discriminant de Fisher à noyaux pour son algorithme de détection. Désobry *et al.* (2005) et Bertrand *et al.* (2011) appliquent leurs algorithmes de détection (utilisant le SVM à une classe pour le premier et un estimateur du maximum de vraisemblance pour le second) sur une fenêtre glissante. Les changements détectés correspondent alors aux maximums locaux significatifs des statistiques calculées. La méthode de la fenêtre glissante a pour avantage une complexité algorithmique linéaire lorsque la taille de la fenêtre glissante est petite par rapport au nombre d'échantillons total.

Mais la complexité moindre des méthodes locales (par rapport aux méthodes globales présentées ci-après) se paie au prix d'une précision diminuée, ces méthodes ne prenant pas en compte l'ensemble des observations pour prendre leurs décisions qui restent locales.

4.5.2 Sélection de modèle par pénalisation du nombre de ruptures

Dans le cadre unidimensionnel, l'objectif est toujours de segmenter une série de n observations $X = (X_1, \ldots, X_n)$ en L segments homogènes, plus particulièrement de détecter des changements dans la moyenne. Plus précisément, le modèle adopté est celui où X est une fonction constante par morceaux contaminée par un certain bruit :

$$X_i = \mu_\ell + \epsilon_i, \quad n^\star_{\ell-1} + 1 \leq i \leq n^\star_\ell, \quad \ell = 1, \ldots, L^\star \,,$$

où $n^\star_0 = 0$ et $n^\star_L = n$, $L^\star$ est le vrai nombre de segments et les $\{\epsilon_t\}_{1 \leq t \leq n}$ sont de moyenne nulle. Ce problème a été entre autres formulé par Yao (1988) ou Lavielle et Moulines (2000), les premiers considérant le bruit comme gaussien alors que les seconds relâchent cette hypothèse et acceptent des familles plus générales de processus à moyenne nulle (par exemple certains processus stationnaires). Bai et Perron (1998) étendent ce cadre aux changements dans les paramètres de régressions linéaires.

Pour un nombre connu de ruptures, l'approche communément utilisée est la minimisation du critère des moindres carrés, c'est-à-dire

$$\min_{1 < n_1 < \cdots < n_{L^\star} = n} \sum_{\ell=1}^{L} \sum_{i=n_{\ell-1}+1}^{n_\ell} \left(X_i - \overline{X}_{n_\ell - 1 : n_\ell}\right)^2 \,,$$

où $\overline{X}_{n_{\ell-1}+1:n_\ell}$ est la moyenne empirique du groupe d'observations $X_{n_{\ell-1}+1}, \ldots, X_{n_\ell}$. Cela revient à chercher la fonction constante par morceaux à L segments correspondant à la meilleure approximation de X.

La détermination du nombre de ruptures se fait en ajoutant un terme de pénalité au programme ci-dessus afin d'éviter un nombre trop grand de segments. Le problème prend ainsi la forme :

$$\min_{1 < n_1 < \cdots < n_L = n} \sum_{\ell=1}^{L} \sum_{i=n_{\ell-1}}^{n_\ell} \left(X_i - \overline{X}_{n_\ell - 1 : n_\ell}\right)^2 + \lambda_n f(L). \tag{4.7}$$

Le choix le plus commun pour la pénalité est celle qui est linéaire en le nombre de changements, *i.e.* $\lambda_n f(L) = \lambda_n L$; Yao (1988) propose ainsi d'utiliser le critère de Schwarz ($\lambda_n = \log(n)$) ; Lebarbier (2005) propose une pénalité de la forme $L(a + b\log(n/L))$, où les constantes a et b sont calibrées à partir des données.

De manière plus générale, la résolution du programme d'optimisation (4.7) afin d'estimer les instants de changements se fait en minimisant une expression du type

$$\sum_{\ell=1}^{L} \Delta(n_{\ell-1}+1 : n_\ell) + \lambda_n f(L)$$

pour les valeurs possibles de L et des segments partitionnant $\{1, \dots, n\}$ – ici $\Delta(n_{\ell-1}+1 : n_\ell) = \sum_{i=n_{\ell-1}+1}^{n_\ell} \left(X_i - \overline{X}_{n_{\ell-1}+1:n_\ell}\right)^2$. Δ est une fonction dite de contraste, ou de coût sur un segment ; le calcul de la minimisation se fait en $\mathcal{O}(Ln^2)$ opérations à l'aide d'un algorithme de programmation dynamique qu'on décrit à la section 6.1.1.

En s'éloignant du modèle gaussien unidimensionel pour se placer dans un espace à grande dimensions, Harchaoui et Cappé (2007) proposent une fonction de contraste à noyau utilisant la statistique du MMD présentée dans la section 4.2.1 On se situe donc dans un cadre non-paramétrique dans lequel les segments que l'on veut obtenir sont homogènes dans leur distribution. La fonction de contraste est alors l'écart moyen des observations vues comme des éléments d'un RKHS à l'élément moyenne dans ce même espace.

4.5.3 Sélection de modèle par l'utilisation de pénalités ℓ_1

Cas unidimensionnel : pénalité sur la variation totale À cause de sa complexité quadratique en le nombre d'échantillons, l'utilisation de l'approche précédente n'est plus possible en un temps limité pour des échantillons de grande taille. Harchaoui et Lévy-Leduc (2010) proposent une variante dont le temps de calcul est de $\mathcal{O}(L_{\max} n \log(n))$, où $L_{\max}$ est une borne supérieure du nombre de changements attendus dans le signal. Le principe est de remplacer la pénalité qui apparaît dans l'équation (4.7), qui est une pénalité de type ℓ_0, par une pénalité ℓ_1 sur la magnitude des sauts (dite également variation totale) c'est-à-dire considérer

$$\min_{U \in \mathbb{R}^n} \|X - U\|^2 + \lambda_n \sum_{i=1}^{n-1} |u_{i+1} - u_i| \ .$$

Ce problème peut être reformulé de la manière suivante :

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^n} \|X - Y\boldsymbol{\beta}\|^2 + \lambda_n \sum_{i=1}^{n} |\beta_i| \ ,$$

où Y est une matrice triangulaire inférieure dont les éléments non nuls sont égaux à 1, et $\boldsymbol{\beta} = (\beta_1, \dots, \beta_n)$ est un vecteur qui contient des valeurs non nulles uniquement aux positions des sauts. On se ramène alors à un problème de sélection

de variables (où chacune des positions de changements potentiels représente une variable) dont la parcimonie est induite par la contrainte sur la norme ℓ_1 de $\boldsymbol{\beta}$; c'est un cadre qui correspond au LASSO (*Least Absolute Shrinkage eStimatOr*) de Tibshirani (1996) qui est résolu grâce à l'algorithme LARS (*Least-Angle Regression*) de Efron *et al.* (2004). En pratique l'algorithme permet de trouver L changements en $\mathcal{O}(Ln\log(n))$ opérations ; les auteurs proposent de chercher un nombre plus grand $L_{\max}$ de changements puis d'appliquer l'algorithme de programmation dynamique sur l'ensemble réduit de $L_{\max}$ changements au lieu de toutes les positions $\{1,\dots,n\}$.

Cas multidimensionnel : *group fused Lasso* La méthode de Harchaoui et Lévy-Leduc (2010), applicable uniquement aux signaux unidimensionnels, est étendue au cadre multidimensionnel par Vert et Bleakley (2010) pour la détection de ruptures de sauts simultanés dans plusieurs dimensions. Les auteurs considèrent le problème de minimisation suivant, pour $X = (\mathbf{X}_1, \dots, \mathbf{X}_n) \in \mathbb{R}^{n\times K}$

$$\min_{U\in\mathbb{R}^{n\times p}} \|X-U\|^2 + \lambda_n \sum_{i=1}^{n-1} \|\mathbf{u}_{i+1,:} - \mathbf{u}_{i,:}\| \ ,$$

où $\mathbf{u}_{i,:}$ représente la i^{e} ligne de la matrice U. Le terme de droite est une pénalité sur la norme 2 des incréments de U. Cette forme de pénalité induit une parcimonie par groupes, c'est-à-dire que les positions des incréments $\mathbf{u}_{i+1,:} - \mathbf{u}_{i,:}$ non nuls vont être situés aux mêmes indices i pour un grand nombre de dimensions, ce qui indique que le signal constant par morceau cible U contient de nombreux changements partagés sur plusieurs dimensions. Une implémentation rapide de l'algorithme de résolution de ce problème d'optimisation est donnée et est effectuée en $\mathcal{O}(nKL)$ opérations pour trouver L ruptures dans les données de dimension K. Dans la section 7.3, nous discuterons des performances de cette approche, comparativement à la méthode d'estimation de changements que nous proposons.

Dans la suite de ce document, nous nous focalisons sur l'utilisation de statistiques de rang marginales en étendant l'idée de Wei et Lachin (1984) pour obtenir des méthodes robustes vis à vis de la distribution des données et de la dimension du problème abordé. Nous nous comparons en termes de performances aux méthodes paramétriques (par exemple dans le cas d'observations gaussiennes) et aux méthodes à noyau. Nous n'avons en revanche pas été en mesure d'aborder les méthodes reposant sur la construction d'arbres des plus proches voisins ; de plus ces méthodes nous semblent peu robustes dans les problèmes à grande dimension.

Nous proposons aussi une méthode pour le problème d'estimation de plusieurs ruptures. Elle sera comparée à des approches locales (segmentation hiérarchique ou par fenêtre glissante) ainsi que globales, par pénalisation du nombre de ruptures.

Chapitre 5

Tests d'homogénéité

Tester l'homogénéité entre plusieurs populations est un problème important dans de nombreux domaines d'applications. Cette tâche permet de valider l'hypothèse selon laquelle les propriétés statistiques d'un échantillon de données sont identiques à celles de tous les autres échantillons constituant le jeu de données complet. Par exemple dans le domaine médical, on peut tester l'homogénéité entre une population ayant reçu un traitement et une autre ayant reçu un placebo afin de déterminer si le traitement produit un effet (l'hypothèse d'homogénéité est rejetée) ou non (les deux populations sont homogènes).

Il existe de nombreuses méthodes pour tester l'homogénéité de deux échantillons : le test de Mann–Whitney/Wilcoxon que nous avons décrit dans la section 4.4 permet de tester l'homogénéité dans le cas univarié et nous avons décrit dans le chapitre précédent quelques méthodes pour le cas multidimensionnel. La contribution apportée dans ce chapitre est la suivante. Nous proposons, dans le cadre multivarié, un test d'homogénéité entre deux échantillons de données puis un test adapté à plus de deux échantillons.

Dans la partie 5.1, nous proposons une méthode pour tester l'homogénéité entre deux échantillons ; elle est inspirée par le test de Wei et Lachin (1984), et peut être vue comme une extension du test de Mann-Whitney/Wilcoxon au cas multivarié. Nous introduisons ensuite dans la section 5.2 une méthode pour tester l'homogénéité de plusieurs groupes de données. Nous évaluons enfin les propriétés des méthodes que nous proposons dans la section 5.3 dans laquelle le test d'homogénéité entre deux échantillons est évalué sur un jeu de données distribué suivant un mélange de gaussiennes. Les deux tests que nous proposons sont utilisés ensuite dans le chapitre 6 pour élaborer des méthodes de détection de changement.

5.1 Test d'homogénéité entre deux échantillons

5.1.1 Présentation de la méthode

Soient $(\mathbf{X}_1, \ldots, \mathbf{X}_n)$ n vecteurs aléatoires de dimension K et $X_{i,k}$ la k^{e} coordonnée de $\mathbf{X_i}$, de sorte que $\mathbf{X}_i = (X_{i,1}, \ldots, X_{i,K})'$.

On se propose de tester l'hypothèse nulle (H_0) « $(\mathbf{X}_1, \ldots, \mathbf{X}_n)$ sont des vecteurs aléatoires i.i.d. » contre l'hypothèse alternative (H_1) « $(\mathbf{X}_1, \ldots, \mathbf{X}_{n_1})$ sont indépendants et identiquement distribués selon $\mathbb{P}_1$; $(\mathbf{X}_{n_1+1}, \ldots, \mathbf{X}_n)$ selon $\mathbb{P}_2$ et $\mathbb{P}_1 \neq \mathbb{P}_2$ ». On suppose ici que, sous (H_0) comme sous (H_1), la distribution des observations n'est pas connue. La statistique de test que nous proposons dans cette section est une extension de la statistique de rang de Mann–Whitney/Wilcoxon aux observations multivariées : on considère le comportement asymptotique conjoint des statistiques de rang calculées à partir de chaque coordonnées des observations. Pour $k \in \{1, \ldots, K\}$, on définit la statistique

$$\mathbf{U}_n(n_1) = (U_{n,1}(n_1), \ldots, U_{n,K}(n_1))'$$

avec

$$U_{n,k}(n_1) = \frac{1}{\sqrt{nn_1(n-n_1)}} \sum_{i=1}^{n_1} \sum_{j=n_1+1}^{n} \left\{ \mathbf{1}(X_{i,k} \leq X_{j,k}) - \mathbf{1}(X_{j,k} \leq X_{i,k}) \right\} . \tag{5.1}$$

On peut remarquer que $U_{n,k}(n_1)$ est liée à la statistique de Mann–Whitney (qui a été décrite dans la section 4.4.1) appliquée aux données $X_{1,k}, \ldots, X_{n,k}$; en effet, en l'absence d'égalité entre deux valeurs $X_{i,k}$ et $X_{j,k}$,

$$\sqrt{nn_1(n-n_1)} U_{n,k}(n_1) = 2W_{\text{MW}} - n_1(n-n_1) ,$$

où W_{MW} est la statistique de Mann–Whitney, décrite à l'équation (4.5).

L'écriture de la statistique (5.1) donnée ci-dessus est utile pour en effectuer l'analyse mathématique ; elle permet aussi de l'étendre à des cas plus généraux (voir par exemple à la section 5.1.3). Mais on peut la réécrire de sorte qu'elle soit plus appropriée que (5.1) d'un point de vue algorithmique comme nous l'expliquons dans la section 5.1.2. On désigne par $R_j^{(k)}$ le rang de $X_{j,k}$ parmi $(X_{1,k}, \ldots, X_{n,k})$, c'est-à-dire $R_j^{(k)} = \sum_{i=1}^{n} \mathbf{1}(X_{i,k} \leq X_{j,k})$. En notant que $\sum_{j=1}^{n} R_j^{(k)} = n(n+1)/2$ et en supposant que les observations $(X_{1,k}, \ldots, X_{n,k})$ sont toutes distinctes, on peut écrire que

$$\begin{aligned} U_{n,k}(n_1) &= \frac{2}{\sqrt{nn_1(n-n_1)}} \sum_{i=1}^{n_1} \left(\frac{n+1}{2} - R_i^{(k)} \right) \\ &= \frac{2}{\sqrt{nn_1(n-n_1)}} \sum_{j=n_1+1}^{n} \left(R_j^{(k)} - \frac{n+1}{2} \right) . \end{aligned} \tag{5.2}$$

Soit $\hat{F}_{n,k}(t) = n^{-1}\sum_{j=1}^{n}\mathbf{1}(X_{j,k} \leq t)$ la fonction de répartition empirique de la k^{e} coordonnée ; on a $\hat{F}_{n,k}(X_{i,k}) = R_i^{(k)}/n$, qui peut donc s'interpréter comme un rang normalisé. Définissons maintenant la matrice de covariance empirique $\hat{\Sigma}_n$; son élément d'indice (k, k') s'écrit

$$\hat{\Sigma}_{n,kk'} = \frac{4}{n}\sum_{i=1}^{n}\{\hat{F}_{n,k}(X_{i,k}) - 1/2\}\{\hat{F}_{n,k'}(X_{i,k'}) - 1/2\},\ 1 \leq k, k' \leq K\,. \tag{5.3}$$

La statistique de test que nous proposons pour le test d'homogénéité entre deux échantillons est définie par

$$S_n(n_1) = \mathbf{U}_n(n_1)'\hat{\Sigma}_n^{-1}\mathbf{U}_n(n_1)\,. \tag{5.4}$$

Le théorème 3 ci-après, dont la démonstration est donnée dans la section 5.4.1, nous fournit le comportement asymptotique de la statistique de test $S_n(n_1)$ lorsque l'hypothèse (H_0) est vraie.

Théorème 3. *Soient $\mathbf{X}_1, \ldots, \mathbf{X}_{n_1}, \mathbf{X}_{n_1+1}, \ldots, \mathbf{X}_n$ des vecteurs aléatoires i.i.d. à valeurs dans $\mathbb{R}^K$ tels que pour tout k de $\{1, \ldots, K\}$, la fonction de répartition F_k de $X_{1,k}$ est une fonction continue. Supposons que $n_1/n \to t_1 \in (0,1)$ et que la matrice de covariance Σ définie par*

$$\Sigma_{kk'} = 4\,\mathrm{Cov}\left(F_k(X_{1,k}); F_{k'}(X_{1,k'})\right),\ 1 \leq k, k' \leq K \tag{5.5}$$

est définie positive. Alors la statistique de test $S_n(n_1)$ définie à l'équation (5.4) *converge en distribution vers une loi du χ^2 à K degrés de liberté.*

Le théorème 3 montre que le test proposé est bien normalisé par rapport à la dimension K des données et à la taille des deux échantillons n_1 et $n - n_1$. Son comportement limite sous (H_0) ne dépend pas de la distribution des données. Par construction, il est aussi invariant par transformation monotone des coordonnées de $\mathbf{X}_i$.

La matrice Σ, qui est la matrice de covariance asymptotique du vecteur $\mathbf{U}_n(n_1)$, est égale, à une constante près, à la matrice de corrélation de Spearman de $\mathbf{X}_i$ (Lehmann, 1975; van der Vaart, 1998) qui est une mesure robuste de dépendance entre coordonnées. Une condition suffisante pour que Σ soit inversible est qu'il n'existe aucune combinaison linéaire des $F_k(X_{1,k})$ qui ne soit presque sûrement égale à une constante.

On peut vérifier aisément que les éléments diagonaux de Σ sont tous égaux à $1/3$ et que $\Sigma_{k\ell} = \Sigma_{\ell k} = 0$ lorsque les coordonnées k et ℓ sont indépendantes. Il s'avère par ailleurs que les éléments diagonaux de $\hat{\Sigma}_n$ convergent très rapidement vers $1/3$; nous n'avons pas observé d'amélioration significative dans les performances de l'algorithme lorsque nous avons voulu prendre en compte cette information, c'est-à-dire en calculant la matrice de corrélation empirique des rangs normalisés dont on multiple ensuite les éléments par $1/3$.

Le théorème 3 permet de définir le taux asymptotique de fausses alarmes associé à la statistique de test $S_n(n_1)$. De plus, le test est consistant (*i.e.* sa puissance tend vers 1) pour les alternatives pour lesquelles le test de Wilcoxon/Mann–Whitney est consistant sur au moins une coordonnée, c'est-à-dire lorsqu'il existe k appartenant à l'ensemble d'entiers $\{1, \dots, K\}$ tel que $\mathbb{P}(X_{k,1} \leq X_{k,n}) \neq 1/2$. On a vu dans le paragraphe 4.4.1 que cette condition est remplie pour des changements dans la moyenne ou des changements d'échelle multiplicatifs lorsque les variables sont à valeurs positives. Il est important de noter que dans le cas multivarié, il suffit que cette condition soit remplie pour un sous ensemble des coordonnées (au moins une) pour que le changement soit détectable ; on illustre ce comportement dans un exemple à la section 5.3.1.

Le résultat donné par le théorème 3 est asymptotique ; aussi avons-nous effectué des simulations de Monte-Carlo afin de vérifier la précision de ce résultat pour une taille d'échantillon finie. En simulant des données ayant des coordonnées indépendantes distribuées selon une loi gaussienne [1]nous avons constaté que la distribution de la statistique $S_n(n_1)$ pouvait être considérée comme suffisamment « proche » de la loi limite – ceci est mesuré à l'aide d'un test de Kolmogorov-Smirnov au niveau 1% – lorsque la taille de l'échantillon n est au moins 8 fois plus grand que la dimension K. Ainsi, pour $K = 20$, il suffit de $n = 210$ observations ; pour $K = 100$, il faut $n = 840$.

La statistique de test (5.4) que l'on propose est inspirée de celle de Wei et Lachin (1984) que nous avons décrit à la section 4.4.3 ; ils proposent une méthode adaptée au cas où les données sont possiblement censurées à droite. La démonstration de leur résultat s'appuie cependant sur une interprétation différente de la matrice Σ ; de celle-ci dérive une matrice de normalisation qui n'est pas identique à $\hat{\Sigma}_n$ comme définie en (5.3). La démonstration que l'on propose (voir à la section 5.4.1) s'appuie sur une méthode classique pour étudier des U-statistiques, la décomposition de Hoeffding. La statistique de Wei et Lachin (1984) diffère donc de $S_n(n_1)$ et s'avère être biaisée dans les cas où $n_1 \neq n/2$, c'est-à-dire lorsque les deux échantillons dont on veut comparer les distributions n'ont pas la même taille ; ce phénomène est visible sur les illustrations du bas de la figure 5.1 qui représentent les histogrammes des statistiques (5.4) et de Wei et Lachin, calculées sur des données de dimension 10. Cette normalisation erronée est particulièrement problématique lorsque l'on s'intéresse au problème de détection de rupture, dont la statistique est construite à partir des statistiques d'homogénéité, le rapport n_1/n pouvant prendre toute valeur entre 0 et 1.

[1]Notons que par construction, le test est invariant par rapport à la distribution utilisée pour les simulations.

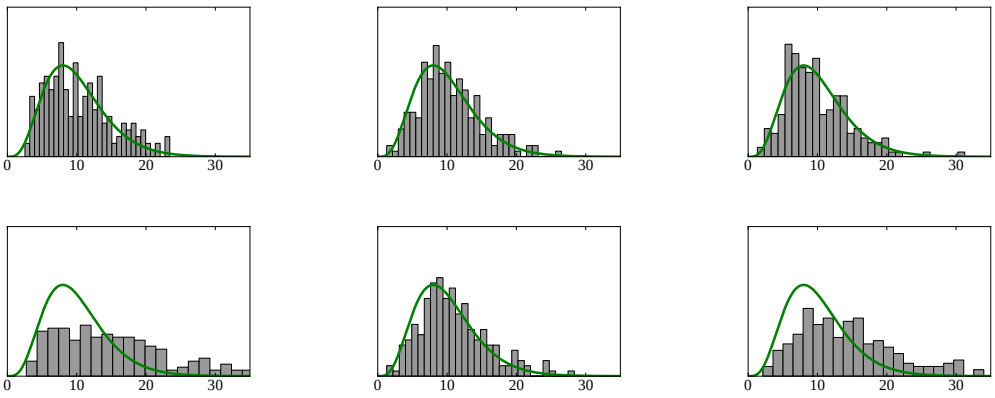


FIGURE 5.1 – Histogrammes des statistiques $S_n(n_1)$ (ligne supérieure) et de Wei et Lachin (ligne inférieure) comparées à la densité de probabilité d'une loi du χ^2_{10}, en fonction du rapport n_1/n. Les statistiques sont calculées à partir d'échantillons de taille $n = 200$ tirés d'une distribution gaussienne standard multivariée de dimension 10.

5.1.2 Implémentation

Comme nous l'avons évoqué précédemment, le vecteur $(U_{n,k}(n_1))_{1\leq k\leq K}$ doit être calculé à partir des rangs comme indiqué en (5.2). Ainsi, $(U_{n,k}(n_1))_{1\leq k\leq K}$ peut être calculé en $\mathcal{O}(Kn\log(n))$ opérations en faisant appel à un tri afin de calculer les rangs. Le calcul de $\hat{\Sigma}_n$, qui fait aussi appel aux rangs normalisés, s'effectue en $\mathcal{O}(K^2n)$ opérations et son inversion en $\mathcal{O}(K^3)$. Notons que, si la statistique de test doit être calculée pour une partition différente des données, ni les rangs, ni la matrice $\hat{\Sigma}_n$ ne doivent être recalculés. Le nombre d'opérations supplémentaires pour calculer par exemple $\hat{S}_n(n_1+1)$ est donc très limité.

Dans certaines situations, il est possible que l'estimateur empirique $\hat{\Sigma}_n$ soit mal conditionné, rendant ainsi le calcul de son inverse impossible. La matrice est mal conditionnée par exemple dans le cas où les coordonnées de $\mathbf{X}_1$ sont fortement dépendantes. Dans le cadre de l'application réseau décrite dans les chapitres précédents, ce cas peut arriver lorsque deux sondes topologiquement proches (par exemple placées à deux arêtes consécutives) enregistrent quasiment le même trafic. Dans le cas limite où deux d'entre elles, par exemple, sont dupliquées, Σ est alors une matrice de rang $K-1$. Pour contourner ce problème, Wei et Lachin (1984) ont suggéré l'addition d'un terme strictement positif aux éléments diagonaux de $\hat{\Sigma}_n$. Nous proposons de régulariser $\hat{\Sigma}_n$ en utilisant une approximation de la pseudo-inverse de Moore-Penrose. Ainsi, si $\hat{\Sigma}_n = USU'$ désigne la décomposition en valeurs singulières de $\hat{\Sigma}_n$, avec $S = \mathrm{diag}(s_1,\ldots,s_K)$ la matrice diagonale contenant les valeurs propres de Σ, alors la pseudo inverse $\Sigma_n^\dagger$ est définie comme

étant $U'\,\mathrm{diag}(s_1^\dagger,\dots,s_K^\dagger)U$, où $s_i^\dagger = s_i^{-1}\mathbf{1}(s_i > \epsilon)$, ϵ étant un seuil fixé strictement positif. Dans le cadre asymptotique, le résultat du théorème 3 est modifié, et la statistique $S_n(n_1)$ est comparée aux quantiles de la distribution $\chi^2_{K'}$, où K' est le nombre de valeurs non nulles de $s_i^\dagger$. Comme déjà mentionné auparavant, certains termes de $\hat{\Sigma}_n$ convergent très rapidement et la matrice est rarement mal conditionnée, même lorsque n n'est que légèrement plus grand que K. Mais dans le cas où cela arrive, la variante régularisée utilisant la pseudo-inverse est efficace, en particulier lorsque les coordonnées sont très dépendantes, par exemple lorsqu'il existe une relation quasi déterministe entre plusieurs coordonnées.

5.1.3 Données discrètes, manquantes ou censurées

Le théorème 3 requiert la continuité des fonctions de répartition F_k de chacune des coordonnées ; à ce titre ce résultat n'est pas directement applicable aux variables discrètes ou lorsque qu'une même valeur peut être prise à plusieurs instants. Cependant, sans cette hypothèse de continuité, le résultat reste valide si l'on redéfinit Σ de la manière suivante :

$$\Sigma_{kk'} = \mathbb{E}\left[\{F_k(X_{1,k}^-) + F_k(X_{1,k}) - 1\}\{F_{k'}(X_{1,k'}^-) + F_{k'}(X_{1,k'}) - 1\}\right] , \qquad (5.6)$$

où $F_k(x^-)$ désigne la limite à gauche de la fonction de répartition au point x. Par ailleurs, il faut aussi remplacer (5.2) par

$$U_{n,k}(n_1) = \frac{2}{\sqrt{nn_1(n-n_1)}} \sum_{j=n_1+1}^{n} \left\{ R_j^{(k)} - \frac{n + \sum_{i=1}^{n} \mathbf{1}(X_{i,k} = X_{j,k})}{2} \right\} .$$

Le cas des données censurées ou manquantes peut aussi être traité à l'aide d'une autre extension de la statistique de test proposée. On introduit ainsi les valeurs de censure inférieure $\underline{X}_{i,k}$ et supérieure $\overline{X}_{i,k}$, de sorte que $\underline{X}_{i,k} \le X_{i,k} \le \overline{X}_{i,k}$. Ici, un cas d'inégalité stricte indiquerait une valeur censurée ; le cas des données manquantes est pris en compte en écrivant $\underline{X}_{i,k} = -\infty$ et $\overline{X}_{i,k} = +\infty$. La statistique marginale modifiée s'écrit

$$U_{n,k}(n_1) = \frac{1}{\sqrt{nn_1(n-n_1)}} \sum_{i=1}^{n_1} \sum_{j=n_1+1}^{n} \left\{ \mathbf{1}(\overline{X}_{i,k} \le \underline{X}_{j,k}) - \mathbf{1}(\overline{X}_{j,k} \le \underline{X}_{i,k}) \right\} . \qquad (5.7)$$

Le théorème 3 est alors vrai pour une matrice Σ dont les éléments s'écrivent

$$\Sigma_{kk'} = \mathbb{E}\left[\{\overline{F}_k(\underline{X}_{1,k}) + \underline{F}_k(\overline{X}_{1,k}^-) - 1\}\{\overline{F}_{k'}(\underline{X}_{1,k'}) + \underline{F}_{k'}(\overline{X}_{1,k'}^-) - 1\}\right] , \qquad (5.8)$$

où $\overline{F}_k$ est la fonction de répartition de $\overline{X}_{1,k}$ et $\underline{F}_k$ est celle de $\underline{X}_{1,k}$.

5.2 Test d'homogénéité entre plusieurs groupes de données

5.2.1 Statistique de test

Dans cette section, nous présentons une extension de la procédure introduite précédemment au cas où l'on veut tester l'homogénéité entre plusieurs groupes de données multivariées. De manière similaire, la statistique de test construite utilise une combinaison de statistiques marginales inspirées de la statistique de Kruskal-Wallis décrite à la section 4.4.2.

Considérons l'hypothèse nulle selon laquelle L groupes de vecteurs aléatoires de $\mathbb{R}^K$ $\mathbf{X}_1, \ldots, \mathbf{X}_{n_1}$; $\mathbf{X}_{n_1+1}, \ldots, \mathbf{X}_{n_2}$; $\ldots$; $\mathbf{X}_{n_{L-1}+1}, \ldots, \mathbf{X}_{n_L}$ sont i.i.d. Dans cette section, on utilise la convention selon laquelle $n_0 = 0$ et $n_L = n$.

Pour $j \in \{1, \ldots, n\}$ et $k \in \{1, \ldots, K\}$, on désigne comme précédemment par $R_j^{(k)}$ le rang de $X_{j,k}$ parmi les $(X_{1,k}, \ldots, X_{n,k})$, c'est-à-dire $R_j^{(k)} = \sum_{i=1}^{n} \mathbf{1}_{\{X_{i,k} \leq X_{j,k}\}}$. Pour $\ell \in \{0, \ldots, L-1\}$, on définit le rang moyen du ℓ^{e} groupe dans la k^{e} coordonnée comme étant $\bar{R}_\ell^{(k)} = (n_{\ell+1} - n_\ell)^{-1} \sum_{j=n_\ell+1}^{n_{\ell+1}} R_j^{(k)}$. Considérons la statistique de test suivante :

$$T(n_1, \ldots, n_{L-1}) = \frac{4}{n^2} \sum_{\ell=0}^{L-1} (n_{\ell+1} - n_\ell) \bar{\mathbf{R}}_\ell' \, \hat{\Sigma}_n^{-1} \, \bar{\mathbf{R}}_\ell \,, \tag{5.9}$$

où l'on définit le vecteur $\bar{\mathbf{R}}_\ell$ par $\bar{\mathbf{R}}_\ell = \left(\bar{R}_\ell^{(1)} - (n+1)/2, \ldots, \bar{R}_\ell^{(K)} - (n+1)/2 \right)'$, et où $\hat{\Sigma}_n$ est la matrice définie à l'équation (5.3) :

$$\hat{\Sigma}_{n,kk'} = \frac{4}{n} \sum_{i=1}^{n} \{\hat{F}_{n,k}(X_{i,k}) - 1/2\} \{\hat{F}_{n,k'}(X_{i,k'}) - 1/2\}, \; 1 \leq k, k' \leq K \,.$$

Une « assez grande » valeur de la statistique (5.9) va ainsi indiquer que l'hypothèse nulle est rejetée, c'est-à-dire que tous les groupes de données ne partagent pas la même distribution sous-jacente. Le seuil de décision est fixé grâce au résultat du théorème 4 ci-dessous. Une justification intuitive de cette statistique est la même que dans le cas unidimensionnel : la quantité $(\bar{R}_\ell^{(k)} - (n+1)/2)$ qui est une composante du vecteur $\bar{\mathbf{R}}_\ell$ représente l'écart entre le rang moyen $\bar{R}_\ell^{(k)}$ de la composante k du groupe ℓ et la quantité qui aurait été attendue si on avait tiré aléatoirement $(n_{\ell+1} - n_\ell)$ entiers parmi $\{1, \ldots, n\}$. Lorsque ce groupe ℓ, en particulier sa k^{e} composante est distribuée différemment des autres groupes de données, cette quantité est grande et contribue à augmenter la valeur de la statistique T. Inversement, sous l'hypothèse nulle, le terme $(\bar{R}_\ell^{(k)} - (n+1)/2)$, normalisé correctement, est asymptotiquement distribué selon une distribution normale.

5.2.2 Cas particuliers

Observons que la statistique (5.9) est une extension au cas multivarié du test classique de Kruskal-Wallis qui s'applique aux données unidimensionnelles. En effet, lorsque $K = 1$, (5.9) s'écrit

$$T(n_1, \ldots, n_{L-1}) = \frac{12}{n^2} \sum_{\ell=0}^{L-1} (n_{\ell+1} - n_\ell) \left(\bar{R}_\ell^{(1)} - (n+1)/2 \right)^2 , \tag{5.10}$$

équation dans laquelle on a remplacé $\hat{\Sigma}_{n,11}$ par $\Sigma_{11} = 4\,\mathrm{Var}(F_1(X_{1,1})) = 4\,\mathrm{Var}(\mathcal{U}) = 1/3$, $\mathcal{U}$ désignant une variable aléatoire uniforme sur $[0, 1]$.

Par ailleurs, dans le cas où l'on ne s'intéresse qu'à deux groupes de données, *i.e.* lorsque $L = 2$, (5.9) se réduit à la statistique de test pour le problème à deux échantillons proposé à la section 5.1. En effet, en utilisant l'écriture (5.2), $T(n_1)$ peut être réécrite comme ceci

$$\begin{aligned} T(n_1) = \frac{nn_1(n-n_1)}{n^2 n_1} \mathbf{U}_n(n_1)' \hat{\Sigma}_n \mathbf{U}_n(n_1) + \frac{nn_1(n-n_1)}{n^2(n-n_1)} \mathbf{U}_n(n_1)' \hat{\Sigma}_n \mathbf{U}_n(n_1) \\ = \mathbf{U}_n(n_1)' \hat{\Sigma}_n \mathbf{U}_n(n_1) = S_n(n_1) , \end{aligned}$$

où $S_n(n_1)$ est définie en (5.4).

5.2.3 Comportement asymptotique de la statistique

Le théorème 4 ci-après, démontré dans la section 5.4.2, permet de décrire le comportement limite de la statistique de test $T(n_1, \ldots, n_{L-1})$ sous l'hypothèse nulle.

Théorème 4. *Soient $(\mathbf{X}_i)_{1 \le i \le n}$ des vecteurs aléatoires i.i.d. à valeurs dans $\mathbb{R}^K$ tels que pour tout $1 \le k \le K$, les fonctions de répartition F_k de $X_{1,k}$ sont des fonctions continues. Si, pour $\ell = 0, \ldots, L-1$, il existe $t_\ell \in (0, 1)$ tel que $(n_{\ell+1} - n_\ell)/n \to t_{\ell+1}$ lorsque n tend vers l'infini, alors la statistique $T(n_1, \ldots, n_{L-1})$ définie en* (5.9) *satisfait*

$$T(n_1, \ldots, n_{L-1}) \xrightarrow{d} \chi^2\left((L-1)K\right) \text{ , as } n \to \infty , \tag{5.11}$$

où $\chi^2((L-1)K)$ désigne la distribution du χ^2 à $(L-1)K$ degrés de liberté.

5.3 Simulations numériques

Dans cette section, nous décrivons les résultats de simulations numériques qui illustrent certains aspects du test d'homogénéité entre deux échantillons précédemment décrit. On désigne ce test par *MultiRank-H* dans la suite.

5.3.1 Illustration du test d'homogénéité de deux échantillons

Pour faire l'évaluation du test d'homogénéité entre deux échantillons introduit à la section 5.1, on considère deux distributions de données différentes. La première distribution est un mélange de deux gaussiennes bi-dimensionnelles ; ces deux gaussiennes ont le même vecteur de moyenne, égal à $(0, 0)$, et leurs matrices de covariances sont les matrices diagonales dont les termes sont égaux à $d_1 = (4, 0.2)$ et $d_2 = (0.2, 4)$, respectivement. La seconde distribution a les mêmes caractéristiques, sauf le vecteur de moyenne qui est dans ce cas égal à $(0.5, 0.5)$. La figure 5.2-(a) montre, sous la forme d'un nuage de points, un exemple de données tirées suivant les deux distributions.

Les méthodes évaluées sont appliquées sur une fenêtre d'observations de $n = 100$ points divisée en deux échantillons de longueur $n/2 = 50$. Sous l'hypothèse nulle, les observations des deux échantillons sont des réalisations de la même distribution ; sous l'hypothèse alternative, chacun des deux échantillons est une réalisation de l'une des deux distributions décrites ci-dessus. Le MultiRank-H est dans ce cadre comparé à trois autres approches. La première, appelée *Maximum Mean Discrepancy* (*MMD*) a été proposée par Gretton *et al.* (2007) ; elle est décrite dans la section 4.2.1. C'est un test non-paramétrique utilisant une méthode à noyaux. Il est utilisé dans cette simulation avec un noyau gaussien dont la bande passante σ est choisie, comme suggéré par les auteurs ainsi que par Harchaoui *et al.* (2009a), comme étant la médiane des distances entre tous les échantillons pris deux à deux. La seconde approche est le test classique du T^2 de Hotelling (Chen et Gupta, 2000, p. 67) que nous avons décrit dans la section 4.1 et qui est optimal dans le cas où les données sont des gaussiennes multivariées (ce qui n'est pas le cas ici). La troisième méthode (« LR ») est celle obtenue en utilisant le test du rapport de vraisemblance en supposant a priori que l'on connaît la structure du modèle (mélange de deux gaussiennes bi-variées) et dont les paramètres sont estimés à l'aide de l'algorithme Espérance-Maximisation (Dempster *et al.*, 1977). Cette dernière approche est optimale dans ce contexte mais est la seule qui fasse appel à une connaissance a priori sur la distribution des données. Ces méthodes sont comparées à l'aide de courbes ROC qui sont obtenues en moyennant 1 000 réplications de Monte Carlo des données. Les expériences, dont les résultats sont représentés à la figure 5.2-(b), montrent que les performances de MultiRank-H sont très similaires à celles du test du rapport de vraisemblance et sont meilleures que les deux autres approches. Le MMD obtient ici de meilleurs résultats que le T^2 de Hotelling, ce qui s'explique par la nature non gaussienne des données. La figure 5.2-(c) correspond à un cas plus difficile pour lequel les données sont identiques au cas précédent à ceci près que des vecteurs gaussiens i.i.d. de dimension 8 et de variance $2{,}5^2$ ont été adjoints aux données originales pour former un vecteur de dimension 10. Les changements éventuels de distribution n'affectent cependant que les deux coordonnées décrites précédemment. De même ici, MultiRank-H obtient des performances comparables

à celles du rapport de vraisemblance, toujours optimal dans cas. On peut aussi noter que le MMD manque de robustesse face à cette transformation des données et est même dominé par le test de Hotelling.

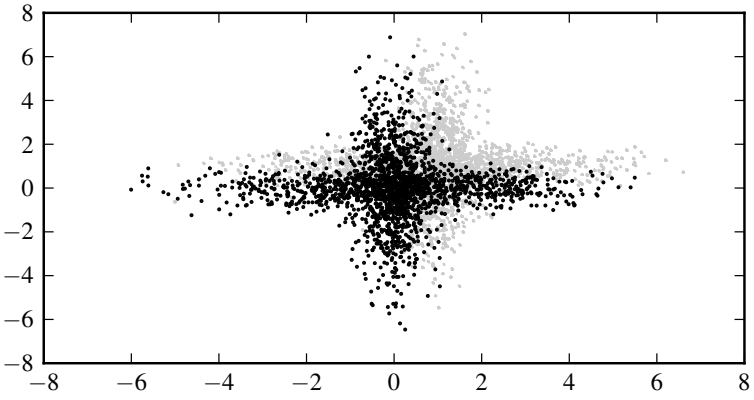
On illustre aussi une situation où le changement n'est pas dans le vecteur de moyenne, mais dans les matrices de covariance des données. Le premier jeu de données est distribué identiquement à celui ci-dessus, c'est-à-dire un mélange de deux gaussiennes bi-dimensionnelles de moyenne $(0, 0)$ et de matrices de covariances diagonales respectives d_1 et d_2. Le second jeu de données conserve les mêmes caractéristiques, hormis pour les matrices de covariance qui sont respectivement $Q_{\pi/4} d_1 Q'_{\pi/4}$ et $Q_{\pi/4} d_2 Q'_{\pi/4}$, où $Q_{\pi/4} = \begin{pmatrix} cos(\pi/4) & -\sin(\pi/4) \\ \sin(\pi/4) & \cos(\pi/4) \end{pmatrix}$. Cette distribution alternative correspond à une rotation d'un angle $\pi/4$ de la distribution précédente, des observations générées selon ces deux distributions sont représentées dans la figure 5.3-(a).

Les courbes ROC représentées à la figure 5.3-(b) sont éloquentes : alors que la méthode du rapport de vraisemblance voire le MMD permettent de différencier les deux distributions, le MultiRank-H en est incapable (la courbe ROC correspondant à MultiRank-H se confond quasiment avec la diagonale qui correspond à la courbe ROC de la décision aléatoire). On peut tenter d'expliquer ce comportement à l'aide de la figure 5.4. On constate que les deux distributions sont symétriques et de moyenne nulle, et que marginalement, les deux distributions diffèrent par leur variance. Dans chacune des dimensions, le test de Wilcoxon n'est pas en mesure de détecter de tels changements. La statistique (5.4) du MultiRank-H, composée de tests marginaux similaires au test de Wilcoxon ne peut donc détecter de différences entre de telles distributions.

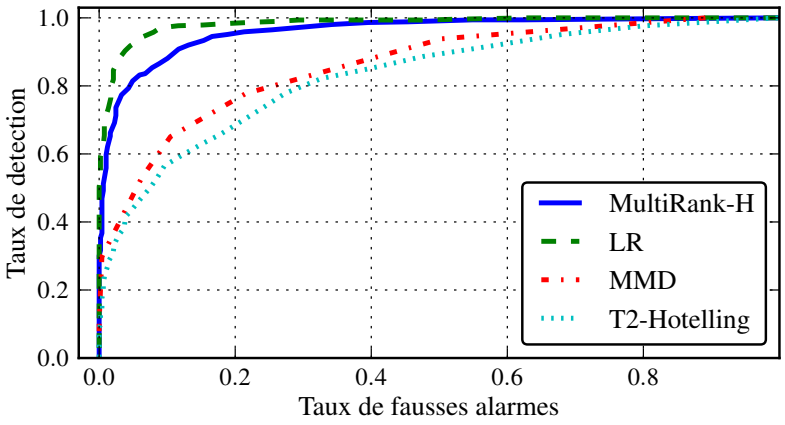
On peut remarquer que le MMD, lorsqu'il est utilisé avec un noyau gaussien isotrope, possède une propriété d'invariance par rotation, c'est-à-dire que si l'on fait tourner l'ensemble des données, on obtiendra exactement la même valeur de la statistique qu'avant la rotation. C'est une propriété que ne possède pas MultiRank-H. Par exemple, si l'on prend le cas bi-dimensionnel, s'il existe une différence dans la moyenne dans une seule coordonnée, en faisant pivoter les données, on retrouvera une différence dans la moyenne dans les deux coordonnées, et donc dans les statistiques calculées dans chacune des coordonnées par MultiRank-H.

5.3.2 Conclusion

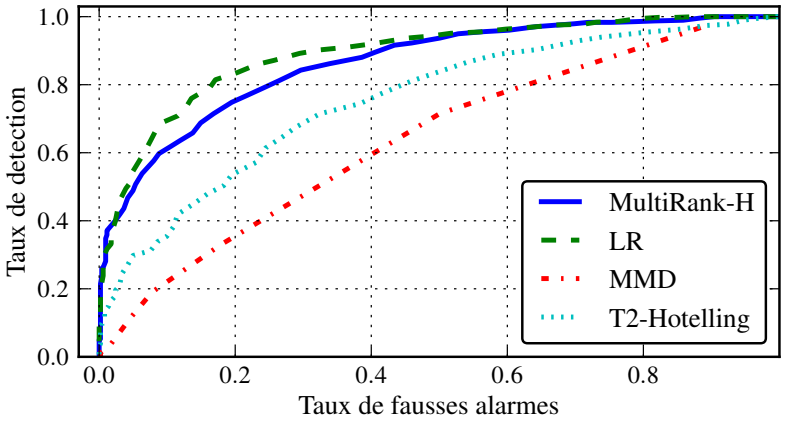
Au cours de ces expériences, nous avons pu mettre en évidence quelques propriétés intéressantes du test d'homogénéité que l'on a proposé dans ce chapitre. C'est un test, qui sans information a priori sur la forme de la distribution sous-jacente des données, permet de tester de manière robuste si deux échantillons diffèrent dans leur moyenne ; cette différence peut n'avoir lieu que dans un nombre li-



(a)



(b)



(c)

FIGURE 5.2 – Changement dans la moyenne. (a) Exemple d'observations tirées selon les deux lois ; (b) Courbes ROC pour les statistiques MultiRank-H, MMD, T^2 de Hotelling et LR ; (c) idem que (b) avec un bruit gaussien de dimension huit ajouté aux signaux originaux.

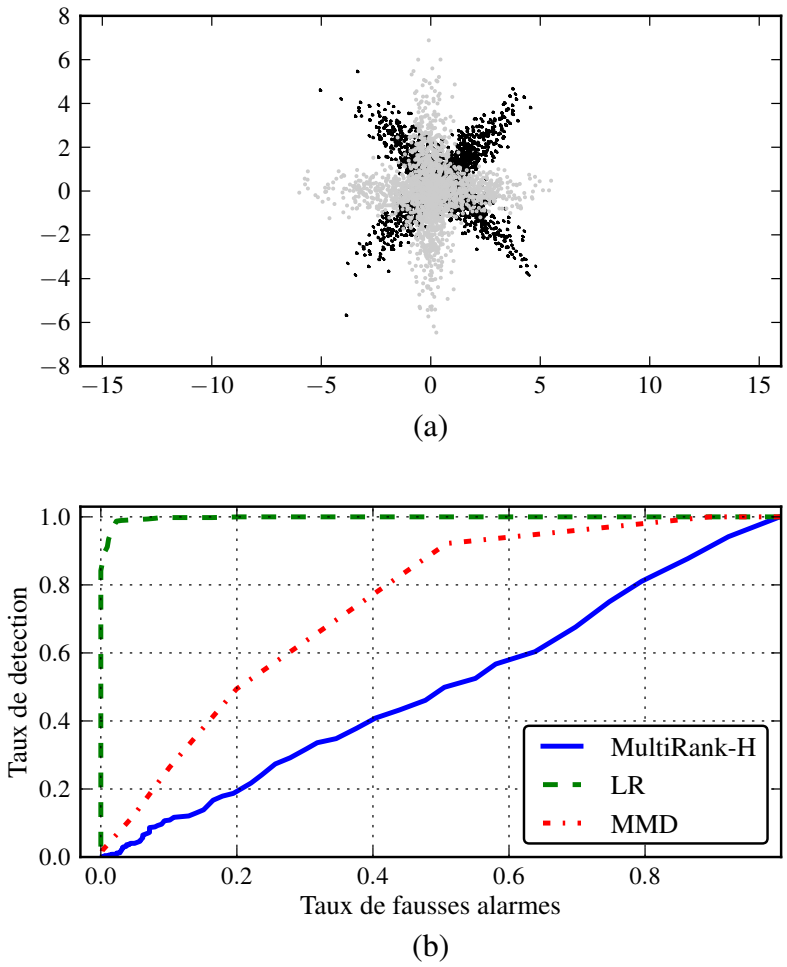


Figure 5.3 – Distribution pivotées. (a) Exemple d'observations distribuées suivant les deux lois ; (b) Courbes ROC pour les statistiques MultiRank-H, MMD et LR ;

mité de coordonnées. Étant une méthode de rang, MultiRank est invariant aux transformations monotones des coordonnées. Cette propriété permet de s'affranchir d'opérations de pré-traitement sur les données, comme une renormalisation (par exemple pour ramener les données toutes les coordonnées dans un intervalle $[-1, 1]$ ou $[0, 1]$, ou alors pour obtenir une variance unitaire), ou une transformation de type logarithmique (utilisée par exemple en économétrie, dans l'application présentée dans la section 7.2).

Nous avons enfin pu exhiber un cas où le test ne permet pas de différencier deux distributions centrées qui ne diffèrent que dans leur variance, en particulier lorsque l'une se déduit de l'autre par rotation. Cette propriété peut être indésirable, par exemple lorsque l'on cherche à tester n'importe quel changement de

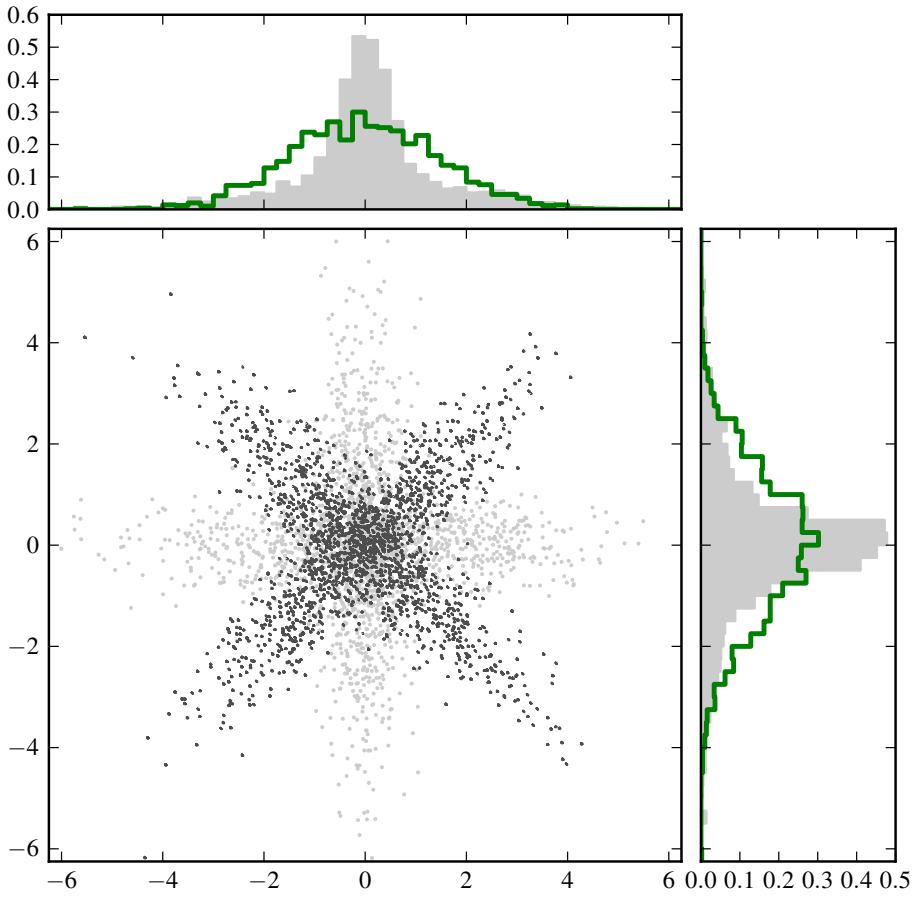


FIGURE 5.4 – Nuage de points des deux distributions pivotées et histogrammes marginaux (histogrammes pleins pour la distribution dont les matrices de covariance sont d_1 et d_2 ; contour des histogrammes pour la distribution pivotée).

distribution. On peut cependant trouver des applications où cette propriété est bienvenue ; par exemple en classification d'objets dont la représentation peut subir des rotations ou des homothéties.

Les méthodes de détection ou d'estimation de ruptures que nous proposons dans le chapitre suivant utilisent les tests d'homogénéité présentés ici. Nous verrons dans le chapitre 6 que les tests de détection de ruptures auront les mêmes propriétés de robustesse et d'invariance que les tests d'homogénéité.

5.4 Démonstrations

5.4.1 Démonstration du théorème 3

Cette démonstration s'appuie sur la décomposition de Hoeffding de $U_{n,k}(n_1)$ pour chaque k de $\{1,\dots,K\}$. Pour des détails supplémentaires sur la décomposition de Hoeffding, on peut se référer aux chapitres 11 et 12 de van der Vaart (1998). Pour chaque k de $\{1,\dots,K\}$, soit $h_{1,k}(y) = \int h(x,y)dF_k(x)$ et $\tilde{h}_{1,k}(x) = \int h(x,y)dF_k(y)$, où h est définie par $h(x,y) = \mathbf{1}(x \leq y) - \mathbf{1}(y \leq x)$. Par continuité de F_k, on a $h_{1,k}(y) = 2F_k(y) - 1$ et $\tilde{h}_{1,k}(x) = 1 - 2F_k(x)$. La décomposition de Hoeffding de $U_{n,k}(n_1)$ peut ainsi s'écrire $U_{n,k}(n_1) = \hat{U}_{n,k}(n_1) + R_{n,k}(n_1)$, où

$$\hat{U}_{n,k}(n_1) = \frac{n_1}{\sqrt{nn_1(n-n_1)}} \sum_{j=n_1+1}^{n} h_{1,k}(X_{j,k}) + \frac{n-n_1}{\sqrt{nn_1(n-n_1)}} \sum_{i=1}^{n_1} \tilde{h}_{1,k}(X_{i,k}) \,, \quad (5.12)$$

$$R_{n,k}(n_1) = \frac{1}{\sqrt{nn_1(n-n_1)}} \sum_{i=1}^{n_1} \sum_{j=n_1+1}^{n} [h(X_{i,k}, X_{j,k}) - \tilde{h}_{1,k}(X_{i,k}) - h_{1,k}(X_{j,k})] \,. \quad (5.13)$$

On montre tout d'abord que $U_{n,k}(n_1) = \hat{U}_{n,k}(n_1) + o_p(1)$ en montrant que la variance de $R_{n,k}(n_1)$ tend vers 0 quand n tend vers l'infini. En utilisant le fait que $\mathbb{E}[U_{n,k}(n_1)] = \mathbb{E}[\hat{U}_{n,k}(n_1)] = 0$, on a

$$\begin{aligned} \mathrm{Var}[R_{n,k}(n_1)] &= \mathrm{Var}[U_{n,k}(n_1) - \hat{U}_{n,k}(n_1)] \\ &= \mathbb{E}[U_{n,k}^2(n_1)] + \mathbb{E}[\hat{U}_{n,k}^2(n_1)] - 2\mathbb{E}[U_{n,k}(n_1)\hat{U}_{n,k}(n_1)]. \end{aligned}$$

Par indépendance des $(X_{i,k})_{1\leq i\leq n}$, on obtient que

$$\mathbb{E}[\hat{U}_{n,k}^2(n_1)] = \frac{n_1^2}{nn_1(n-n_1)} \sum_{j=n_1+1}^{n} \mathbb{E}[h_{1,k}(X_{j,k})^2] + \frac{(n-n_1)^2}{nn_1(n-n_1)} \sum_{i=1}^{n_1} \mathbb{E}[\tilde{h}_{1,k}(X_{i,k})^2]. \quad (5.14)$$

Avec l'égalité

$$\mathbb{E}[h_{1,k}(X_{i,k})^2] = 4\mathbb{E}[(F_k(X_{1,k}) - 1/2)^2] = 4\,\mathrm{Var}(\mathcal{U}) = 1/3 \,, \quad (5.15)$$

où $\mathcal{U}$ est distribuée selon un loi uniforme sur $[0,1]$, on a, d'une part, que

$$\mathbb{E}[\hat{U}_{n,k}^2(n_1)] = \frac{n_1^2(n-n_1)}{3nn_1(n-n_1)} + \frac{(n-n_1)^2 n_1}{3nn_1(n-n_1)} = 1/3\,. \tag{5.16}$$

D'autre part

$$\begin{aligned}\mathbb{E}[U_{n,k}^2(n_1)] = {} & \frac{1}{nn_1(n-n_1)} \sum_{i=1}^{n_1} \sum_{j=n_1+1}^{n} \mathbb{E}[h(X_{i,k}, X_{j,k})^2] \\ & + \frac{1}{nn_1(n-n_1)} \sum_{1\le i\neq i'\le n_1} \sum_{j=n_1+1}^{n} \mathbb{E}[h(X_{i,k}, X_{j,k}) h(X_{i',k}, X_{j,k})] \\ & + \frac{1}{nn_1(n-n_1)} \sum_{i=1}^{n_1} \sum_{n_1+1\le j\neq j'\le n} \mathbb{E}[h(X_{i,k}, X_{j,k}) h(X_{i,k}, X_{j',k})]\,. \end{aligned} \tag{5.17}$$

Étudions maintenant séparément les trois termes de la partie droite de l'équation (5.17). En utilisant le fait que les variables $(X_{i,k})_{1\le i\le n}$ sont i.i.d. on a

$$\begin{aligned}&\frac{1}{nn_1(n-n_1)} \sum_{i=1}^{n_1} \sum_{j=n_1+1}^{n} \mathbb{E}[h(X_{i,k}, X_{j,k})^2] \\ &\qquad = \frac{n_1(n-n_1)}{nn_1(n-n_1)} \mathbb{E}[h(X_{1,k}, X_{n_1+1,k})^2] \to 0, \text{ quand } n\to\infty\,. \end{aligned} \tag{5.18}$$

Ensuite, par continuité de F_k, on a

$$\begin{aligned}&\frac{1}{nn_1(n-n_1)} \sum_{1\le i\neq i'\le n_1} \sum_{j=n_1+1}^{n} \mathbb{E}[h(X_{i,k}, X_{j,k}) h(X_{i',k}, X_{j,k})] \\ &= \frac{(n_1^2-n_1)(n-n_1)}{nn_1(n-n_1)} \int (2F_k(y)-1)(2F_k(y)-1)\,dF_k(y) = \frac{n_1(n_1-1)(n-n_1)}{3nn_1(n-n_1)}\,. \end{aligned} \tag{5.19}$$

Enfin, avec un argument similaire, le dernier terme de la partie droite de l'équation (5.17) est égal à $n_1(n-n_1)(n-n_1-1)/(3nn_1(n-n_1))$. En combinant (5.18) et (5.19), on obtient enfin que

$$\mathbb{E}[U_{n,k}^2(n_1)] \to 1/3, \text{ quand } n\to\infty. \tag{5.20}$$

Puisque $\mathbb{E}[U_{n,k}(n_1)\hat{U}_{n,k}(n_1)] \to 1/3$, quand $n\to\infty$, on déduit de (5.16) et de (5.20) que $\mathrm{Var}[R_{n,k}(n_1)] \to 0$; on a donc $U_{n,k}(n_1) = \hat{U}_{n,k}(n_1) + o_p(1)$, quand n tend vers l'infini.

Grâce au théorème central limite multivarié, on a $(U_{n,1}(n_1), \ldots, U_{n,K}(n_1))' \to \mathcal{N}(0,\Sigma)$, où le $(k,k')^{\text{e}}$ élément de Σ est donné par

$$\Sigma_{kk'} = \lim_{n\to\infty} \mathbb{E}[\hat{U}_{n,k}(n_1)\hat{U}_{n,k'}(n_1)]\,.$$

En utilisant le fait que les $(X_{i,k})_{1\leq i\leq n}$ sont i.i.d., on obtient que

$$\begin{aligned}\mathbb{E}[\hat{U}_{n,k}(n_1)\hat{U}_{n,k'}(n_1)] &= \frac{4n_1^2}{nn_1(n-n_1)}\sum_{j=n_1+1}^{n}\mathbb{E}[\{F_k(X_{j,k})-1/2\}\{F_{k'}(X_{j,k'})-1/2\}] \\ &+ \frac{4(n-n_1)^2}{nn_1(n-n_1)}\sum_{i=1}^{n_1}\mathbb{E}[\{F_k(X_{i,k})-1/2\}\{F_{k'}(X_{i,k'})-1/2\}] \\ &= 4\,\mathrm{Cov}\left(F_k(X_{1,k}), F_{k'}(X_{1,k'})\right)\ .\end{aligned}$$

On a donc $\Sigma^{-1/2}(U_{n,1}(n_1),\ldots,U_{n,K}(n_1))' \xrightarrow{d} \mathcal{N}(0,\mathrm{Id}_K)$. Puisque $\hat{\Sigma}_n \xrightarrow{p} \Sigma$, on déduit que, d'après le théorème de Slutsky, $\hat{\Sigma}_n^{-1/2}(U_{n,1}(n_1),\ldots,U_{n,K}(n_1))' \xrightarrow{d} \mathcal{N}(0,\mathrm{Id}_K)$, ce qui conclut cette démonstration.

5.4.2 Démonstration du théorème 4

Avec $R_j^{(k)} = \sum_{i=1}^n \mathbf{1}(X_{i,k}\leq X_{j,k})$, on obtient que

$$\begin{aligned}\bar{R}_\ell^{(k)} - \frac{n+1}{2} &= \frac{1}{n_{\ell+1}-n_\ell}\left(\sum_{j=n_\ell+1}^{n_{\ell+1}}\sum_{i=1}^{n}\mathbf{1}(X_{i,k}\leq X_{j,k})\right) - \frac{n+1}{2} \\ &= \frac{1}{n_{\ell+1}-n_\ell}\sum_{j=n_\ell+1}^{n_{\ell+1}}\sum_{\substack{i=1\\ i\neq j}}^{n}\left[\mathbf{1}(X_{i,k}\leq X_{j,k})-1/2\right]\ . \quad (5.21)\end{aligned}$$

Soit $h(x,y)=\mathbf{1}(x\leq y)$, $h_{1,k}(y)=\int \mathbf{1}(x\leq y)\mathrm{d}F_k(x)$ et $h_{2,k}(x)=\int\mathbf{1}(x\leq y)\mathrm{d}F_k(y)$. Par continuité de F_k : $h_{1,k}(y)=F_k(y)$ et $h_{2,k}(x)=1-F_k(x)$. Avec la notation

$$\mathcal{R}_\ell^{(k)} = \frac{(n_{\ell+1}-n_\ell)^{1/2}}{n}(\bar{R}_\ell^{(k)}-(n+1)/2)\ ,$$

la décomposition de Hoeffding permet d'écrire

$$\begin{aligned}&\mathcal{R}_\ell^{(k)} = \\ &\frac{(n_{\ell+1}-n_\ell)^{1/2}}{n}\left[\frac{n-1}{n_{\ell+1}-n_\ell}\sum_{j=n_\ell+1}^{n_{\ell+1}}(h_{1,k}(X_{j,k})-\frac{1}{2}) + \frac{n_{\ell+1}-n_\ell-1}{n_{\ell+1}-n_\ell}\sum_{i=1}^{n}(h_{2,k}(X_{i,k})-\frac{1}{2})\right] \\ &+\frac{(n_{\ell+1}-n_\ell)^{1/2}}{n}\left[\frac{1}{n_{\ell+1}-n_\ell}\sum_{j=n_\ell+1}^{n_{\ell+1}}\sum_{\substack{i=1\\ i\neq j}}^{n}\left\{h(X_{i,k},X_{j,k})-h_{1,k}(X_{j,k})-h_{2,k}(X_{i,k})+\frac{1}{2}\right\}\right] \\ &\qquad\qquad \stackrel{\text{def}}{=} \mathcal{R}_{\ell,1}^{(k)}+\mathcal{R}_{\ell,2}^{(k)}+\mathcal{R}_{\ell,3}^{(k)}\ . \quad (5.22)\end{aligned}$$

Notons que $\mathcal{R}_{\ell,3}^{(k)} = o_p(1)$, quand n tend vers l'infini, puisque on peut prouver que $\mathrm{Var}(\mathcal{R}_{\ell,3}^{(k)}) = \mathrm{Var}[\mathcal{R}_\ell^{(k)} - (\mathcal{R}_{\ell,1}^{(k)}+\mathcal{R}_{\ell,2}^{(k)})] \to 0$, quand n tend vers l'infini. L'équation

(5.22) peut donc être réécrite

$$\mathcal{R}_\ell^{(k)} = \frac{n-1}{n(n_{\ell+1}-n_\ell)^{1/2}} \sum_{j=n_\ell+1}^{n_{\ell+1}} (F_k(X_{j,k}) - \frac{1}{2}) + \frac{n_{\ell+1}-n_\ell-1}{n(n_{\ell+1}-n_\ell)^{1/2}} \sum_{i=1}^{n} (\frac{1}{2} - F_k(X_{i,k})) + o_p(1) \ .$$

Puisque $\sum_{i=1}^{n}(1/2 - F_k(X_{i,k})) = \sum_{p=0}^{L-1}\sum_{j=n_p+1}^{n_{p+1}}(1/2 - F_k(X_{j,k}))$, on a

$$\begin{aligned}\mathcal{R}_\ell^{(k)} = {} & \frac{n-(n_{\ell+1}-n_\ell)}{n(n_{\ell+1}-n_\ell)^{1/2}} \sum_{j=n_\ell+1}^{n_{\ell+1}} (F_k(X_{j,k}) - 1/2) \\ & - \frac{n_{\ell+1}-n_\ell-1}{n(n_{\ell+1}-n_\ell)^{1/2}} \sum_{\substack{p=0 \\ p\neq\ell}}^{L-1} \sum_{j=n_p+1}^{n_{p+1}} (F_k(X_{j,k} - 1/2)) + o_p(1) \\ & \stackrel{\text{def}}{=} U_k(n_\ell, n_{\ell+1}) + o_p(1) \ .\end{aligned}$$

On peut observer que, pour ℓ fixé, $0 \leq \ell \leq L-1$, et pour k, k' tels que $1 \leq k \leq K$ et $1 \leq k' \leq K$, on a, pour n tendant vers l'infini,

$$4\,\mathrm{Cov}(U_k(n_\ell, n_{\ell+1}), U_{k'}(n_\ell, n_{\ell+1})) = \Sigma_{kk'} \left[\left(1 - \frac{(n_{\ell+1}-n_\ell)}{n}\right)^2 + \sum_{\substack{p=0 \\ p\neq\ell}}^{L-1} \frac{(n_{\ell+1}-n_\ell-1)^2(n_{p+1}-n_p)}{n^2(n_{\ell+1}-n_\ell)} \right] \to (1-t_{\ell+1})\Sigma_{kk'} \ , \tag{5.23}$$

où l'on a utilisé le fait que

$$\sum_{\substack{p=0 \\ p\neq\ell}}^{L-1} (n_{p+1}-n_p) = n - (n_{\ell+1}-n_\ell).$$

De la même manière, pour k, k' fixés de $\{1, \ldots, K\}$ et $\ell \neq \ell'$ de $\{0, \ldots, L-1\}$, on obtient que, pour n tendant vers l'infini,

$$4\,\mathrm{Cov}(U_k(n_\ell, n_{\ell+1}), U_{k'}(n_{\ell'}, n_{\ell'+1})) \to -\sqrt{t_{\ell+1}t_{\ell'+1}}\Sigma_{kk'} \ . \tag{5.24}$$

Soit

$$\bar{R}_n = 2\left(\frac{(n_1-n_0)^{1/2}}{n}\bar{R}_0', \ldots, \frac{(n_L-n_{L-1})^{1/2}}{n}\bar{R}_{L-1}'\right)' \ .$$

On déduit de (5.23), (5.24) et du théorème central limite multivarié que

$$\bar{R}_n \xrightarrow{d} \mathcal{N}(0, \Theta \otimes \Sigma) \ , n \to \infty \ ,$$

où Σ est la matrice de dimensions $K \times K$ définie en (5.5), $\otimes$ est le produit de Kronecker et $\Theta = \mathrm{Id}_L - \sqrt{t}\sqrt{t}'$ avec $\sqrt{t} = (\sqrt{t_1}, \ldots, \sqrt{t_L})'$. Donc,

$$\bar{R}_n^{\Sigma} \xrightarrow{d} \mathcal{N}(0, \Theta \otimes \mathrm{Id}_K) \, , n \to \infty \, ,$$

où

$$\bar{R}_n^{\Sigma} = 2 \left(\frac{(n_1 - n_0)^{1/2}}{n} \Sigma^{-1/2} \bar{R}_0', \ldots, \frac{(n_L - n_{L-1})^{1/2}}{n} \Sigma^{-1/2} \bar{R}_{L-1}' \right)' \, .$$

Puisque $\hat{\Sigma}_n \xrightarrow{p} \Sigma$, lorsque n tend vers l'infini, on a la même convergence lorsqu'on remplace Σ par $\hat{\Sigma}_n$. Puisque $\sum_{\ell=0}^{L-1}(n_{\ell+1} - n_\ell)/n = 1$, $\sum_{\ell=1}^{L} t_\ell = 1$, la matrice t a donc pour valeur propre 0 avec multiplicité 1 (avec espace propre associé engendré par $\sqrt{t}$) et 1 avec multiplicité $L - 1$. Les valeurs propres de $\Theta \otimes \mathrm{Id}_K$ sont donc 0, avec multiplicité K, et 1 avec multiplicité $(L-1)K$, ce qui conclut la démonstration.

Chapitre 6

Estimation et détection de ruptures

On s'intéresse maintenant au problème de l'estimation et de la détection de ruptures. Dans le cas où les données sont gaussiennes et que l'on cherche un changement dans la moyenne, Chen et Gupta (2000) présentaient un test qui consistait à faire varier la position de la frontière séparant la fenêtre de données en deux sous-fenêtres et à calculer la statistique du T^2 de Hotelling dans chacune des configurations. La statistique pour le test de détection de changement était alors tout simplement la valeur maximale obtenue. Si celle-ci dépassait un certain seuil, alors un changement était déclaré.

C'est cette démarche que nous utilisons dans ce chapitre pour concevoir un nouveau test de détection de changement applicable aux données multidimensionnelles. Celui-ci s'appuie sur le test d'homogénéité que nous avons proposé au chapitre précédent et il conserve les propriétés inhérentes aux tests de rang (indépendance vis à vis de la distribution sous-jacente des données, robustesse vis à vis de la contamination par des valeurs aberrantes, etc.). Un des problèmes à résoudre réside dans la manière dont on peut fixer le seuil au-delà duquel on peut décider systématiquement de la présence d'une rupture.

Dans une première section, nous introduisons une méthode d'estimation de ruptures multiples qui s'appuie sur la généralisation multivariée du test de Kruskal et Wallis. Puis dans la section 6.2, nous donnons un résultat asymptotique dans le cas où l'on cherche au plus un changement dans la fenêtre de données qui nous permet de fixer un seuil décision et d'évaluer le niveau de significativité des ruptures détectées. Enfin nous évaluons les méthodes introduites et mettons en évidence quelques unes de leurs propriétés à l'aide de simulations numériques.

6.1 Estimation de ruptures multiples

6.1.1 Nombre de ruptures connu et programmation dynamique

On suppose dans un premier temps que le nombre de ruptures $L-1$ est connu. Nous proposons de nous appuyer sur la statistique décrite dans la section 5.2 pour déterminer les positions des $L-1$ frontières $n_1, \ldots, n_{L-1}$ entre les différents segments. Ces positions, inconnues, sont estimées en maximisant la statistique $T(n_1, \ldots, n_{L-1})$, définie en (5.9), par rapport à $n_1, \ldots, n_{L-1}$:

$$(\hat{n}_1, \ldots, \hat{n}_{L-1}) = \underset{1<n_1<\cdots<n_{L-1}<n}{\operatorname{argmax}} T(n_1, \ldots, n_{L-1}) \, . \tag{6.1}$$

On veut ainsi trouver la partition de la fenêtre de n échantillons en L sous segments qui maximise la statistique T du test d'homogénéité, les sous-segments étant délimités par les positions $n_1, \ldots, n_{L-1}$.

En pratique, la maximisation de (6.1) est très coûteuse en temps de calcul : c'est un problème combinatoire dont la complexité augmente exponentiellement avec le nombre de ruptures et la taille de l'échantillon. Cependant, la matrice $\hat{\Sigma}_n$ étant commune à chacun des segments, il s'avère que la statistique $T(n_1, \ldots, n_{L-1})$ a une structure additive. Il est donc possible d'utiliser une technique de programmation dynamique – dont l'origine remonte à Bellman (1961) – pour effectuer la maximisation. On peut ainsi se référer à l'algorithme classique de programmation dynamique pour la segmentation qui a été décrit par Kay (1993) et utilisé par exemple par Bai et Perron (2003).

Ainsi, dans notre cas, si l'on utilise la notation

$$\Delta(n_\ell + 1 : n_{\ell+1}) = (n_{\ell+1} - n_\ell) \bar{\mathbf{R}}'_{n_\ell+1:n_{\ell+1}} \, \hat{\Sigma}_n^{-1} \, \bar{\mathbf{R}}_{n_\ell+1:n_{\ell+1}} \, , \tag{6.2}$$

où

$$\bar{\mathbf{R}}_{n_\ell+1:n_{\ell+1}} = \left(\bar{R}^{(1)}_{n_\ell+1:n_{\ell+1}} - (n+1)/2, \ldots, \bar{R}^{(K)}_{n_\ell+1:n_{\ell+1}} - (n+1)/2 \right)'$$

et $\bar{R}^{(k)}_{n_\ell+1:n_{\ell+1}} = (n_{\ell+1} - n_\ell)^{-1} \sum_{j=n_\ell+1}^{n_{\ell+1}} R_j^{(k)}$, on peut alors écrire

$$I_L(p) = \max_{1<n_1<\cdots<n_{L-1}<n_L=p} \sum_{\ell=0}^{L-1} \Delta(n_\ell + 1 : n_{\ell+1}) \, . \tag{6.3}$$

On a alors l'égalité suivante, aussi appelée équation de Bellman :

$$I_L(p) = \max_{n_{L-1}} \left\{ I_{L-1}(n_{L-1}) + \Delta(n_{L-1} + 1 : p) \right\} \, . \tag{6.4}$$

Soient $E \in \{2, \ldots, n\}$ et ℓ un entier. La quantité $I_\ell(E)$ représente la plus grande valeur de la statistique T (à la normalisation près qui n'est pas importante pour l'estimation des positions) calculée sur la partie des données allant de l'indice 1 à

l'indice E pour une partition à ℓ segments. L'équation (6.4) indique donc que pour trouver la meilleure segmentation de la partie des données indexées de 1 à p en L segments, il suffit de connaître les meilleures partitions à $L-1$ segments des données indexées de 1 à j, pour j variant de $L-1$ à p. La résolution du problème initial, à savoir le calcul de $I_L(n)$, s'effectue alors récursivement.

Ainsi, pour résoudre le problème d'optimisation (6.1), on procède comme suit :

1. calcul de $\Delta(i:j)$ avec (6.2) pour tout couple (i,j) tel que $1 \leq i < j \leq n$. En particulier, on définit $I_1(j) = \Delta(1:j)$, pour $j = 1, \ldots, n$;
2. pour $\ell = 1, \ldots, L-1$ et $p = \ell, \ldots, n$, calcul de

$$I_{\ell+1}(p) = \max_{\ell \leq n_\ell \leq p-1} \{I_\ell(n_\ell) + \Delta(n_\ell + 1 : p)\}$$

   et de

$$V_{\ell+1}(p) = \operatorname*{argmax}_{\ell \leq n_\ell \leq p-1} \{I_\ell(n_\ell) + \Delta(n_\ell + 1 : p)\} \ ;$$

3. les positions des ruptures sont estimées grâce à une récursion inverse :

$$\begin{cases} \hat{n}_L = V_L(n) \\ \hat{n}_\ell = V_{\ell+1}(\hat{n}_{\ell+1}) \text{ pour } \ell = (L-1), \ldots, 1. \end{cases}$$

Au vu de la nature récursive de l'algorithme, notons que pendant le calcul de $I_L(n)$ on obtient aussi les meilleures partitions des données à $2, 3, \ldots, L-1$ segments, ce qui est une propriété intéressante lorsque l'on examine le cas où le nombre de ruptures est inconnu. Une stratégie pour déterminer le nombre de ruptures est alors de calculer un grand nombre de ruptures puis d'en trouver le nombre optimal grâce à une procédure a posteriori. Nous en décrivons une dans la section suivante.

6.1.2 Nombre de ruptures inconnu

L'estimation du nombre de ruptures est un problème généralement difficile. L'ajout d'une pénalité à un critère d'attache aux données est l'une des méthodes couramment utilisées (Lavielle et Moulines, 2000; Lebarbier, 2005; Lavielle, 2005). On peut aussi évoquer l'utilisation d'a priori sur la position des ruptures lorsqu'un point de vue bayésien est adopté (Ruanaidh et Fitzgerald, 1996; Fearnhead, 2006), point de vue que l'on n'a pas eu l'occasion d'étudier dans le cadre de cette thèse ; ou encore d'une pénalité associée à la norme ℓ_1 (Harchaoui et Lévy-Leduc, 2010) ou ℓ_1 par blocs (Vert et Bleakley, 2010).

Nous proposons une méthode utilisant une heuristique de pente pour résoudre ce problème, variante d'idées utilisées par exemple dans Lavielle (2005), alternatives aux critères utilisant des pénalités de type AIC ou BIC, qui sont en

pratique peu performantes et ont tendance à fortement sur-estimer le nombre de segments, comme expliqué par Lavielle (2005) dans le cas unidimensionnel. La méthode proposée est fondée sur le principe qu'en présence de $S^\star \geq 1$ ruptures (pour $L = S^\star + 1$ segments), si l'on trace $I_L(n)$ en fonction de L, pour $L = 1, \ldots, L_{\max}$, le graphe qui en résulte peut se décomposer en deux parties : une première, pour $L = 1, \ldots, L^\star$ pour laquelle le critère augmente rapidement ; et une seconde, $L = L^\star, \ldots, L_{\max}$ où la statistique subit une croissance beaucoup plus faible. Pour chaque valeur de L ($L = 2, \ldots, (L_{\max} - 1)$), on calcule donc une régression linéaire par la méthode des moindres carrés pour la partie avant et après L ; le nombre estimé de ruptures est la valeur de L pour laquelle la somme des carrés des résidus calculés sur chacune des parties est minimale (voir la Figure 6.1).

Le cas $L = 1$ est traité séparément ; la procédure que nous venons de décrire n'est appliquée que lorsque $T(\hat{n}_1)$ a une valeur significative, au sens où la p-valeur associée au test d'une unique rupture, obtenue selon la méthode décrite dans la section 6.2, est assez petite.

Il est à noter que le choix de la valeur maximale $L_{\max}$ – qui doit être fixée avec une connaissance a priori du problème à un multiple (deux ou trois fois par exemple) du nombre de segments attendus par le statisticien – a une influence sur le nombre de segments estimés : augmenter ce paramètre aura tendance à diminuer l'erreur de régression de la partie droite du graphe de $I_L(n)$ et donc d'augmenter la « tolérance » à une erreur sur la partie de gauche du graphe, augmentant ainsi la valeur de L estimée.

6.2 Évaluation du niveau de significativité du test dans le cas de la rupture unique

L'élaboration d'algorithmes d'estimation de positions des ruptures est une chose, une autre est d'avoir des outils permettant de mesurer le degré de pertinence d'une partition donnée en plusieurs segments. Un premier pas dans cette direction serait de pouvoir caractériser le comportement de $T(\hat{n}_1, \ldots, \hat{n}_{L-1})$ sous l'hypothèse nulle d'homogénéité de l'ensemble des données. La phase d'optimisation sur l'ensemble des configurations de ruptures possibles rend cette tâche difficile. Une approche possible consiste à procéder à des expériences de Monte Carlo ou à utiliser des techniques de *bootstrap*, si un échantillon représentatif des données est disponible, afin d'estimer la distribution de la statistique. Dans cette section, nous montrons que dans le cas où $L = 2$, c'est-à-dire lorsque l'on ne cherche qu'une unique rupture, il est possible d'obtenir une bonne approximation asymptotique de la p-valeur du test.

Pour ce faire, on considère dans cette section une variante de la statistique de test utilisée en (6.1). Les conséquences pratiques de l'utilisation de cette statistique

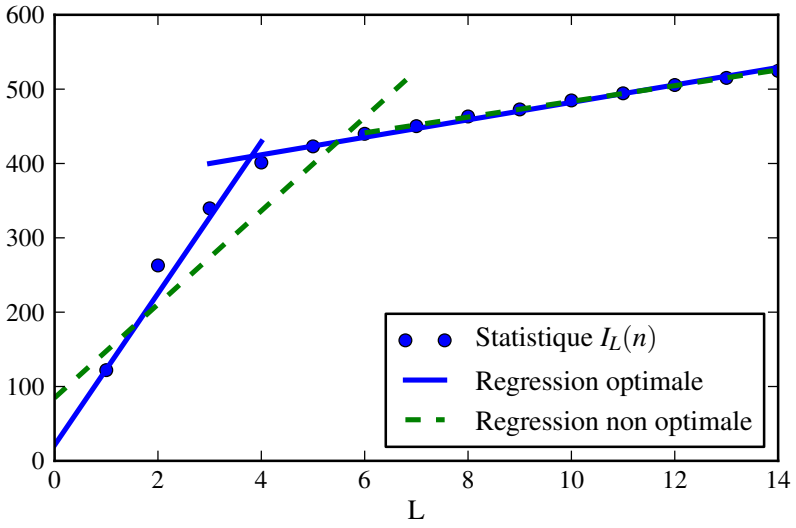


FIGURE 6.1 – Estimation du nombre de ruptures. Dans le cas illustré dans cette figure, le vrai nombre de segments est $L^\star = 4$ (régression correspondante représentée en trait plein) ; un mauvais modèle, ici $L = 6$ est également représenté en pointillés

modifiée plutôt que de $T(\hat{n}_1)$ seront examinées peu après le théorème 5 qui est le résultat principal de cette section.

Soit ainsi $\mathbf{V}_n(n_1) = (V_{n,1}(n_1), \ldots, V_{n,K}(n_1))'$ le vecteur dont les éléments sont

$$V_{n,k}(n_1) = \frac{1}{n^{3/2}} \sum_{i=1}^{n_1} \sum_{j=n_1+1}^{n} \left\{ \mathbf{1}(X_{i,k} \leq X_{j,k}) - \mathbf{1}(X_{j,k} \leq X_{i,k}) \right\}, \; k = 1, \ldots, K \,, \quad (6.5)$$

et définissons

$$\tilde{S}_n(n_1) = \mathbf{V}_n(n_1)' \hat{\Sigma}_n^{-1} \mathbf{V}_n(n_1) \,. \quad (6.6)$$

Notons que $\mathbf{V}_n$ ne diffère de $\mathbf{U}_n$ dont les éléments avaient été définis en (5.1) que par la normalisation, qui est maintenant indépendante de n_1. La matrice $\hat{\Sigma}_n$ est en revanche identique au cas précédent. Considérons maintenant la statistique

$$W_n = \max_{1 \leq n_1 \leq n-1} \tilde{S}_n(n_1) \,. \quad (6.7)$$

Le théorème suivant, dont la démonstration est donnée en section 6.4, permet de donner la p-valeur asymptotique de W_n sous l'hypothèse nulle correspondant à l'absence de rupture au sein des observations.

Théorème 5. *Soit $(\mathbf{X}_i)_{1 \leq i \leq n}$ un vecteur aléatoire i.i.d. à valeurs dans $\mathbb{R}^K$ tel que pour tout k, la fonction de répartition F_k de $X_{1,k}$ est une fonction continue et tel que la matrice*

Σ *de dimensions* $K \times K$ *définie en* (5.5) *est inversible. Alors*

$$W_n \xrightarrow{d} \sup_{0<t<1} \left(\sum_{k=1}^{K} B_k^2(t) \right), \text{ quand } n \to \infty, \tag{6.8}$$

où d désigne la convergence en distribution et $\{B_k(t),\ t \in (0,1)\}_{1 \le k \le K}$ *sont des ponts browniens indépendants.*

Pour déterminer la p-valeur $P_{\text{val}}(W_n)$ associée à la statistique (6.8), on peut utiliser le résultat suivant de Kiefer (1959) :

$$\begin{aligned} P_{\text{val}}(b) &= \mathbb{P}\left(\sup_{0<t<1} \left(\sum_{k=1}^{K} B_k^2(t) \right) > b \right) \\ &= 1 - \frac{4}{\Gamma(\frac{K}{2}) 2^{\frac{K}{2}} b^{\frac{K}{2}}} \sum_{m=1}^{\infty} \frac{(\gamma_{(K-2)/2,m})^{K-2} \exp[-(\gamma_{(K-2)/2,m})^2]/2b}{[J_{K/2}(\gamma_{(K-2)/2,m})]^2}, \end{aligned} \tag{6.9}$$

où J_ν est la fonction de Bessel de première espèce, $\gamma_{\nu,m}$ est le m^{e} zéro de J_ν et Γ est la fonction Gamma. En pratique, seul un petit nombre de termes de la série doit être calculé. Par exemple, pour une dimension de $K = 40$, les p-valeurs sont calculées à partir d'une trentaine de termes de la série.

Nous avons exploré de manière empirique la convergence de la statistique W_n vers sa distribution limite. On utilise ainsi le test de Kolmogorov Smirnov qui permet de tester si un échantillon suit bien une loi donnée. Pour une taille d'échantillon n et une dimension K données, on calcule la statistique de test W_n pour 1 000 tirages des données distribuées suivant une loi normale standard. Pour des valeurs de K différentes, on cherche donc quelle est la taille de l'échantillon minimale pour que la distribution de la statistique soit considérée comme assez proche de la distribution asymptotique définie par (6.9), c'est-à-dire à partir de quelle valeur de n l'hypothèse nulle (du test de Kolmogorov Smirnov) n'est pas rejetée, le niveau du test étant fixé à 5 %. Comme illustré à la figure 6.2, la taille de l'échantillon requise augmente linéairement avec la dimension K des données : n doit être environ 8 fois plus grand que la dimension. Ce fait est par ailleurs illustré à la figure 6.3

6.2.1 Normalisation de la statistique de test

Comme indiqué au début de la section 6.2, la normalisation de $\mathbf{V}_n$ (qui est $n^{-3/2}$) diffère de celle de $\mathbf{U}_n$ ($n^{-3/2}(n_1/n \times (n-n_1)/n)^{-1/2}$) d'un facteur multiplicatif dépendant de n_1 ; la statistique W_n diffère ainsi de $T(\hat{n}_1)$. D'expérience, et d'après les courbes ROC de la figure 6.4 [1], en utilisant $\mathbf{U}_n$ à la place de $\mathbf{V}_n$ dans la définition de W_n, c'est-à-dire utiliser $T(\hat{n}_1)$, on obtient une statistique aux capacités de

[1] Ces courbes ROC, ainsi que les résultats présentés dans la figure 6.5 sont obtenus en évaluant les deux statistiques dans le protocole expérimental présenté au début de la section 6.3 qui suit.

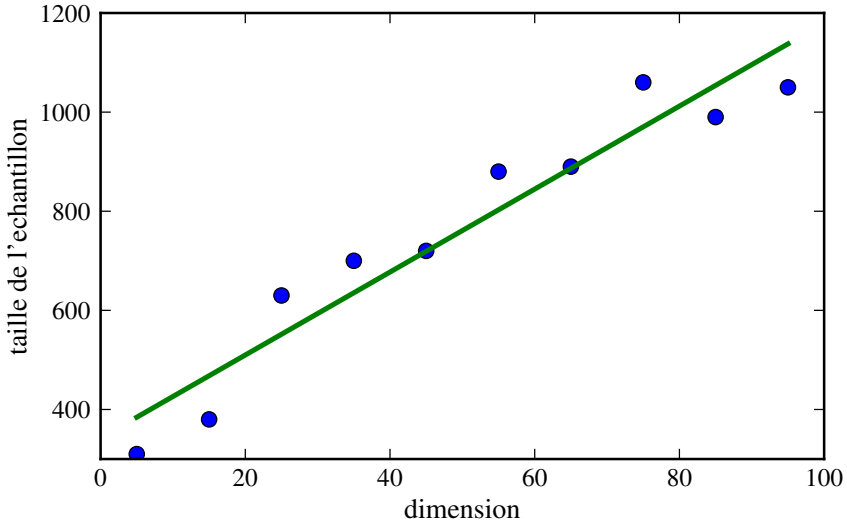


FIGURE 6.2 – Taille minimum de l'échantillon en fonction de la dimension pour que la distribution de la statistique W_n soit « assez proche » de la distribution asymptotique.

détection et de localisation similaires lorsque la rupture éventuelle se situe vers le milieu de la fenêtre d'observation (entre $n/4$ et $3n/4$). Pour des ruptures situées près du début ou de la fin la fenêtre, $T(\hat{n}_1)$ obtient une meilleure puissance de détection, au prix toutefois d'une légère augmentation du taux de fausses alarmes, celles-ci ayant lieu près des bords de la fenêtre.

Ces fausses alarmes correspondent à des cas où n_1 est proche 0 ou n, ce qui fait « exploser » le terme de normalisation. On retrouve ce phénomène lorsqu'on examine la répartition des positions estimées par $T(\hat{n}_1)$ sous H_0 que l'on représente dans l'illustration du haut de la figure 6.5. On constate que la répartition de la position estimée du maximum est uniforme sur la fenêtre, mais avec cependant un nombre non négligeable d'estimations au niveau des bords de la fenêtre. A contrario, avec W_n, les estimations des positions se concentrent plutôt vers le milieu de la fenêtre.

L'avantage de l'utilisation de la normalisation dépendant de n_1 est mis en évidence dans l'illustration du bas de la figure 6.5. Pour W_n (gauche) et $T(n_1)$ (droite), elle représente la répartition des positions estimées des maxima de la statistique en présence d'un changement, situé respectivement au $1/8$, $1/6$, $1/4$ et à la moitié de la fenêtre. Si W_n estime correctement la position de la rupture lorsque celle-ci est située à la moitié de la fenêtre, les positions estimées deviennent approximatives quand la rupture n'est plus à la moitié de la fenêtre. En revanche, la grande majorité des estimations obtenues par la statistique $T(n_1)$ sont correctes.

En procédant de la même manière que dans la démonstration en section 6.4,

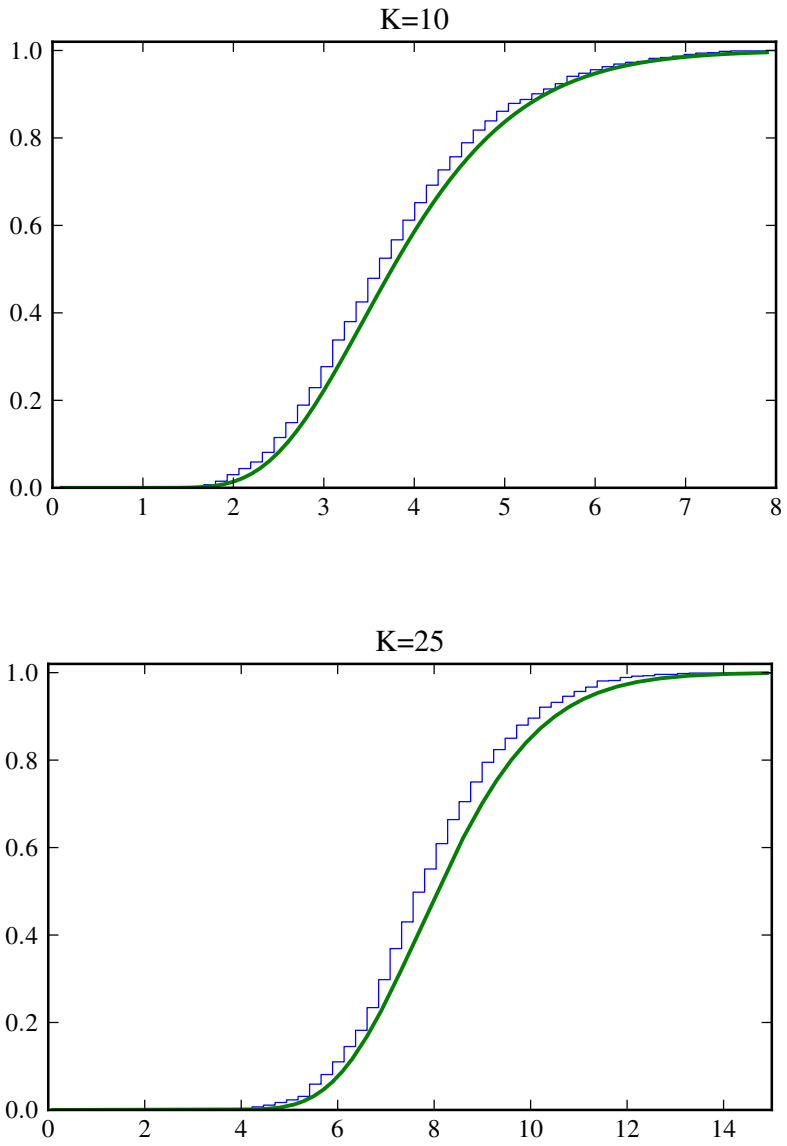


Figure 6.3 – Histogrammes cumulatifs de 1000 valeurs de la statistique W_n sous H_0 comparés à la fonction de répartition théorique calculée avec (6.9), pour une dimension des données de $K = 10$ (haut) et $K = 25$ (bas).

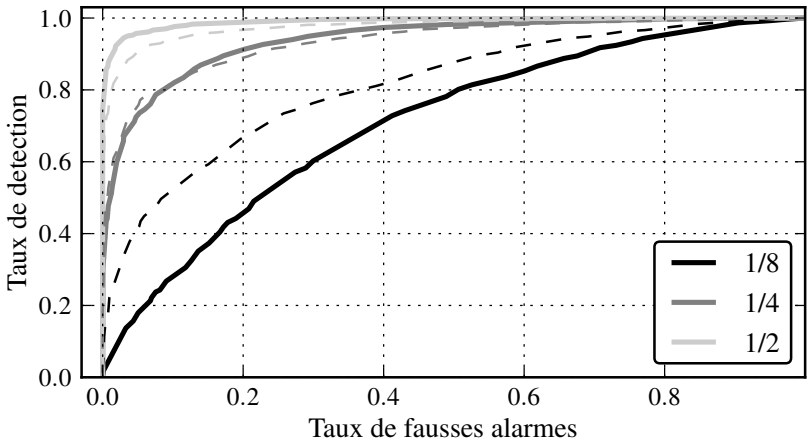


FIGURE 6.4 – Courbes ROC des statistiques W_n (traits pleins) et $\max_{n_1} T(n_1)$ (pointillés) lorsque sous H_1, la rupture est située au 1/8, 1/4 et 1/2 de la fenêtre.

nous pensons que l'on pourrait démontrer un résultat analogue à celui du théorème 5 pour $T(\hat{n}_1)$, dont le supremum n_1 convergerait en loi vers

$$\sup_{0<u<t<v<1} M_K(t)^2 \quad \text{où} \quad M_K(t) = \left(\sum_{k=1}^{K} \frac{B_k^2(t)}{t(1-t)} \right)^{1/2},$$

où u et v sont des bornes sur les valeurs admissibles de t. Csörgő et Horváth (1997, corollaire A.3.1) proposent le résultat suivant pour obtenir le comportement limite de ces formes quadratiques de ponts browniens : pour tout λ réel strictement positif et tout x réel,

$$\lim_{n\to\infty} \mathbb{P}\left\{ (2\log\log n)^{1/2} \sup_{\lambda/n\le t\le 1-\lambda/n} M_K(t) \le x + D_K(\log n) \right\} = \exp(-2e^{-x}),$$

où $D_K(x) = 2\log x + K/2 \log\log x - \log\Gamma(K/2)$ et Γ est la fonction gamma. Le terme λ impose une condition sur les valeurs admissibles de la frontière n_1 entre les deux segments, à savoir que le maximum de la statistique doit être cherché uniquement dans une plage telle que n_1/n soit bornée par valeur supérieure et inférieure. Cette condition est plutôt raisonnable : on ne peut accorder que peu de crédit au résultat obtenu lorsque, pour une petite valeur de n_1, on compare un échantillon de taille n_1 avec un autre de taille $n - n_1$.

On en conclut donc que pour choisir une statistique pour la détection de rupture, on doit choisir entre la fiabilité de la position de la rupture estimée et la possibilité de donner un niveau de significativé fiable à la décision prise.

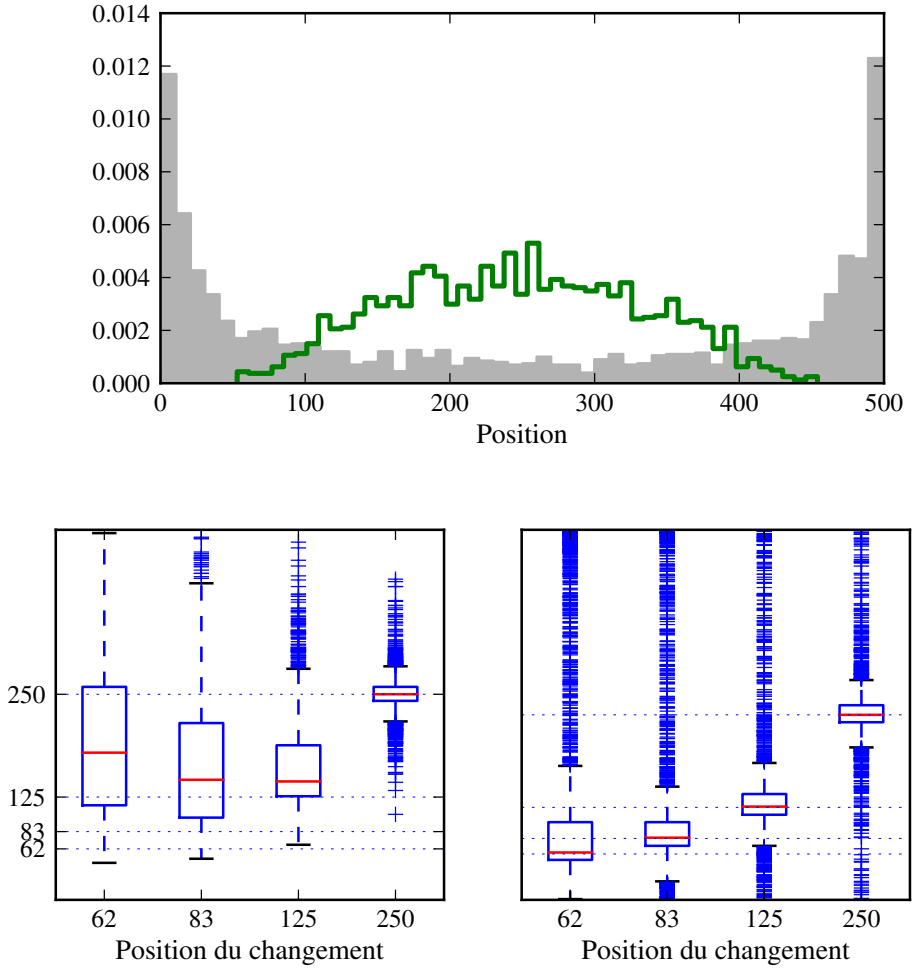


Figure 6.5 – Estimation de la position de la rupture. Haut : Histogramme sous H_0 de $\operatorname{argmax} \tilde{S}_n(n_1)$ (contour de l'histogramme) et de $\operatorname{argmax} T(n_1)$ (histogramme plein). Bas : positions estimées (boîtes à moustaches) sous H_1 des ruptures lorsque celles-ci sont situées au 1/8, 1/6, 1/4 et 1/2 (de gauche à droite) de la fenêtre de taille 500, pour $\operatorname{argmax} \tilde{S}_n(n_1)$ (gauche) et $\operatorname{argmax} T(n_1)$ (droite).

6.3 Simulations numériques

Dans cette section, nous nous intéressons aux propriétés du test de détection de ruptures utilisant la statistique W_n définie en (6.7), qui est désignée par *MultiRank* dans la suite. Les simulations numériques dont nous reportons les résultats dans cette section sont obtenus à partir d'un scénario de test commun. Sous l'hypothèse (H_0), c'est-à-dire en l'absence de rupture, on génère 500 échantillons d'une loi gaussienne de dimension 5 de moyenne nulle et de matrice de covariance identité. Sous (H_1), les observations sont similaires à l'exception du vecteur de moyenne qui devient 0,2 sur toutes les coordonnées à partir du point de changement qui est situé à $1/4$ ou $1/2$ de la fenêtre d'observation, à savoir aux indices 125 ou 250. Ce scénario correspond à une situation simple où toutes les coordonnées des données subissent un saut positif dans la moyenne. Les courbes ROC qui sont tracées dans la suite sont obtenues à partir de 2 000 réplications des observations.

6.3.1 Comparaison avec les décisions marginales

La statistique de test du MultiRank utilise une combinaison de statistiques de rang marginales, *i.e.* calculées dans chaque coordonnée. Elle incorpore néanmoins deux aspect importants du problème de la détection de rupture multivariée : d'une part, la détection de ruptures simultanées dans plusieurs coordonnées doit rendre plus probable la présence d'une rupture réelle ; d'autre part l'existence de relations de dépendances entre les coordonnées doit influencer la prise de décision. Pour illustrer ces propriétés, on compare MultiRank avec une méthode plus simple combinant les décisions marginales avec la correction de Bonferroni ; la statistique de test utilisée s'écrit ainsi

$$\max_{1\leq k\leq K} \max_{1\leq n_1\leq n-1} V_{n,k}(n_1) \ .$$

Les résultats obtenus sur les données générées comme décrit au début de la section 6.3 sont représentés sur le graphe supérieur de la figure 6.6. Nous avons aussi comparé les deux méthodes dans une configuration où la matrice de covariance du vecteur gaussien n'est plus l'identité mais une matrice tridiagonale dont les valeurs des éléments des sur- et sous-diagonale est de 0,45 (désigné dans la figure par « Corrélations positives ») ou $-0{,}45$ (« Corrélations négatives »). Les courbes ROC résultantes sont tracées au milieu et en bas de la figure 6.6.

Le haut de la figure 6.6 illustre le fait que la méthode qui combine les statistiques marginales en tenant compte de leurs corrélations, à savoir le MultiRank, obtient de meilleurs résultats qu'une méthode de type Bonferroni dans le cas indépendant. Les performances de détection de l'approche de Bonferroni ne varient pas lorsque des corrélations, négatives ou positives, apparaissent entre les coordonnées des données. En revanche, le MultiRank est affecté : quand les coordonnées sont corrélées positivement, son taux de détection diminue pour atteindre

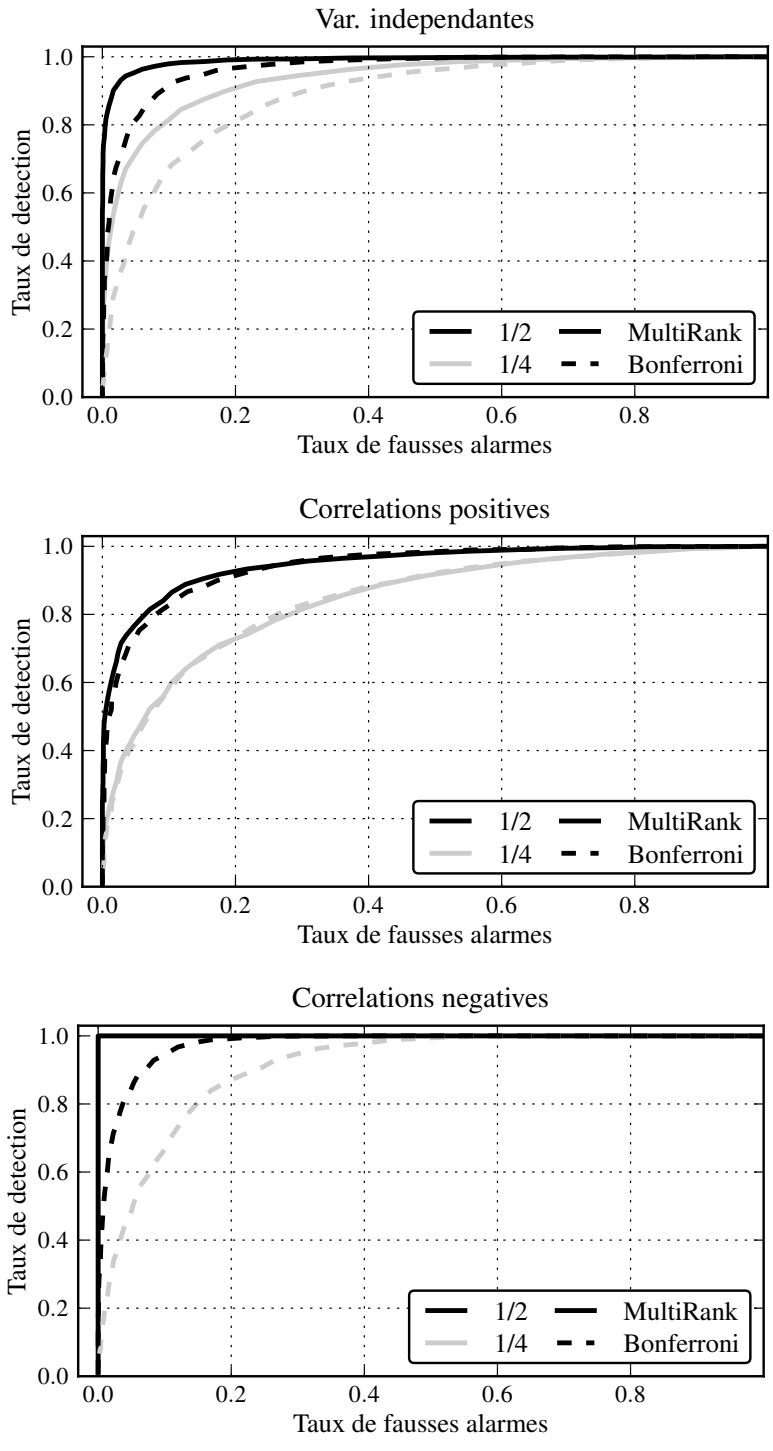


Figure 6.6 – Courbes ROC pour les approches MultiRank et de type Bonferroni lorsque les coordonnées sont indépendantes (haut), corrélées positivement (milieu) et corrélées négativement (bas). L'instant de rupture est situé au quart et à la moitié de la fenêtre d'observations de longueur 500.

celui de Bonferroni ; avec des corrélations négatives, celui-ci augmente considérablement. L'approche du MultiRank permet donc d'exploiter des données du problèmes qui ne sont pas utilisées lorsque l'on ne s'intéresse qu'aux décisions marginales : des corrélations négatives dans les données rendent plus faciles la détection de sauts simultanés positifs et des corrélations positives les rendent plus difficiles.

6.3.2 Robustesse de la statistique à la présence de valeurs aberrantes

Un avantage bien connu des méthodes de rang est leur robustesse en présence de valeurs aberrantes. Prenons en effet pour exemple le calcul de la moyenne d'un ensemble de données. Si l'un des éléments de l'échantillon est remplacé par une très grande valeur, la moyenne peut être modifiée assez substantiellement. Si au lieu de calculer la moyenne des valeurs des éléments, on calcule la moyenne de leurs rangs, les valeurs possibles des rangs ne vont pas être modifiées et cette moyenne ne sera pas affectée par cette valeur aberrante. C'est cette propriété de robustesse, dans le cas du MultiRank, que l'on illustre dans cette section. Nous considérons le scénario dans lequel les données (dont la distribution a été décrite en début de section 6.3) sont progressivement contaminées par une quantité de plus en plus grande de valeurs aberrantes. La distribution de ces valeurs aberrantes est celle d'une gaussienne multivariée dont la matrice de covariance est de $10\ \mathrm{Id}_5$, alors que la matrice de covariance des données initiales est Id_5 (matrice identité de dimension 5). La proportion de valeurs aberrantes dans les données est de 0, 5 ou 20%. L'approche du MultiRank est comparée au test paramétrique de détection de ruptures utilisant le rapport de vraisemblance qui est décrit par Srivastava et Worsley (1986) ainsi que dans la section 4.1. Cette méthode, qui s'appuie elle-même sur la statistique du T^2 de Hotelling, est optimale en l'absence de valeurs aberrantes puisque les distributions des données avant et après une éventuelle rupture sont gaussiennes.

Dans le cas où les données ne contiennent pas de valeurs aberrantes, le MultiRank obtient des performances comparables à celles de l'approche paramétrique, comme illustré sur le graphe supérieur de la figure 6.7. Les deux autres graphiques de la figure 6.7 mettent quant à elles en évidence la robustesse du MultiRank vis à vis de la contamination par valeurs aberrantes, puisque, contrairement à la méthode paramétrique, les performances de MultiRank sont à peine affectées par cette contamination.

6.3.3 Robustesse par rapport à différents profils de changement

Nous avons jusqu'à présent examiné dans nos simulations numériques le cas des changements abrupts et simultanés : le passage d'un régime à un autre se faisait instantanément et simultanément sur toutes les coordonnées concernées. Nous

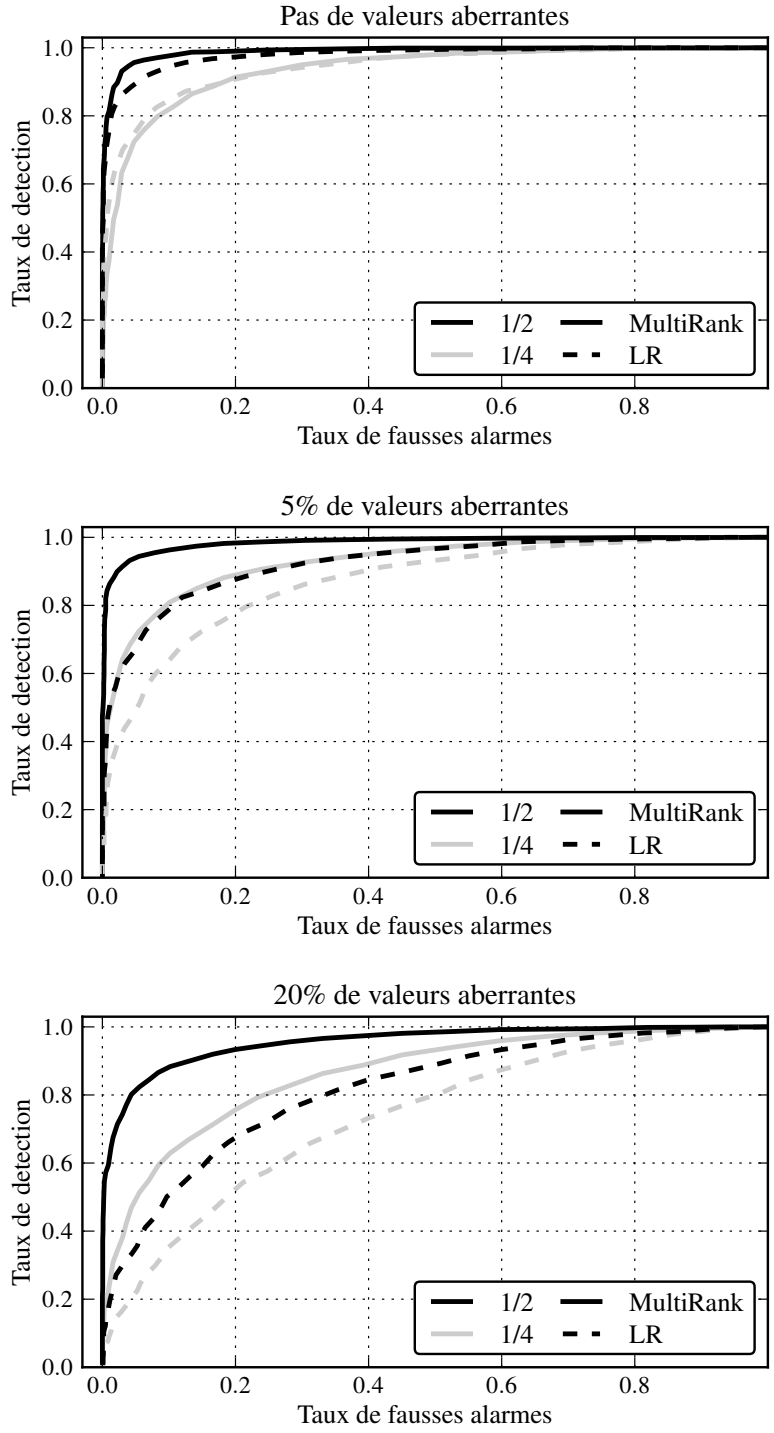


Figure 6.7 – Courbes ROC pour les approches MultiRank et utilisant le rapport de vraisemblance pour trois proportions différentes de valeurs aberrantes (de haut en bas : 0, 5 et 20%) pour un instant de changement situé au quart et à la moitié de la fenêtre d'observations de longueur 500.

nous intéressons maintenant à une configuration où il existe une zone ou période de transition. Le mécanisme de génération des données décrit en début de section 6.3 est conservé, on considère néanmoins deux nouveaux profils de changements. Dans la première situation, la position des ruptures dans chacune des coordonnées est possiblement différente et est répartie uniformément dans l'intervalle $[n_1 - \delta, n_1 + \delta]$. Dans la seconde situation, au lieu de « sauter » de manière abrupte à la position n_1, les données présentent une progression linéaire entre les deux niveaux du signal, depuis 0 à l'indice $n_1 - \Delta$ à 0,2 à l'indice $n_1 + \Delta$. Ces deux situations sont illustrées dans des cas typiques à la figure 6.8.

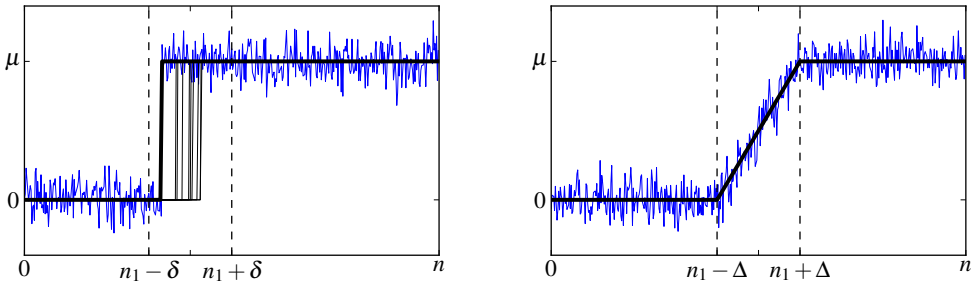


FIGURE 6.8 – Deux différents profils de changements, pour un saut de 0 à $\mu = 0{,}2$ autour de la position n_1. Gauche : position des ruptures réparties entre $n_1 - \delta$ et $n_1 + \delta$. Droite : saut linéaire entre $n_1 - \Delta$ et $n_1 + \Delta$.

La méthode du MultiRank est encore une fois comparée ici au test utilisant le rapport de vraisemblance de Srivastava et Worsley (1986). Dans ces expériences, les deux méthodes produisent des résultats similaires et il s'avère qu'elles sont ne sont quasiment pas affectées par le fait que les ruptures soient moins abruptes. En effet, pour $\delta = \Delta = 1$, les aires sous la courbe ROC (AUC) pour les deux algorithmes sont proches de 0,99 quand $n_1/n = 1/2$ et de 0,94 lorsque la rupture est plus éloignée du milieu de la fenêtre, à $n_1/n = 1/4$. Dans les deux cas et pour les deux méthodes, les AUC ne diminuent que de 0,02 quand on fixe δ ou Δ à 100. La dégradation des performances ne devient significative que lorsque les ruptures ne se produisent que près du début ou de la fin de la fenêtre d'observation.

6.3.4 Évaluation de la méthode pour changements multiples

On s'intéresse maintenant à la méthode d'estimation rétrospective de ruptures multiples présentée dans la section 6.1. Les simulations présentées dans ce paragraphe utilisent un signal fixe de dimension 5 constant par morceaux contenant un ensemble prédéfini de 4 ruptures. Chacune de ces ruptures ne se produit simultanément que dans un sous-ensemble des coordonnées ; c'est une caractéristique que l'on retrouve dans de nombreuses applications (par exemple dans la détection

d'anomalies réseau ou dans l'analyse des données de l'ADN d'individus atteints du cancer, application qui est présentée dans le chapitre 7). Ce signal est contaminé par du bruit gaussien corrélé de variance marginale σ^2 ; la figure 6.9 correspond par exemple à un rapport signal à bruit marginal (défini comme le rapport entre l'amplitude du saut et σ) de 16 dB. Les performances des algorithmes comparés sont mesurés en termes de *précision* (proportion de ruptures correctement estimées par rapport aux ruptures estimées) et de *rappel* (rapport entre le nombre de ruptures trouvées par l'algorithme sur le « vrai » nombre de ruptures) avec une tolérance de ± 1 et ± 5 échantillons pour déterminer si une rupture a été correctement estimée.

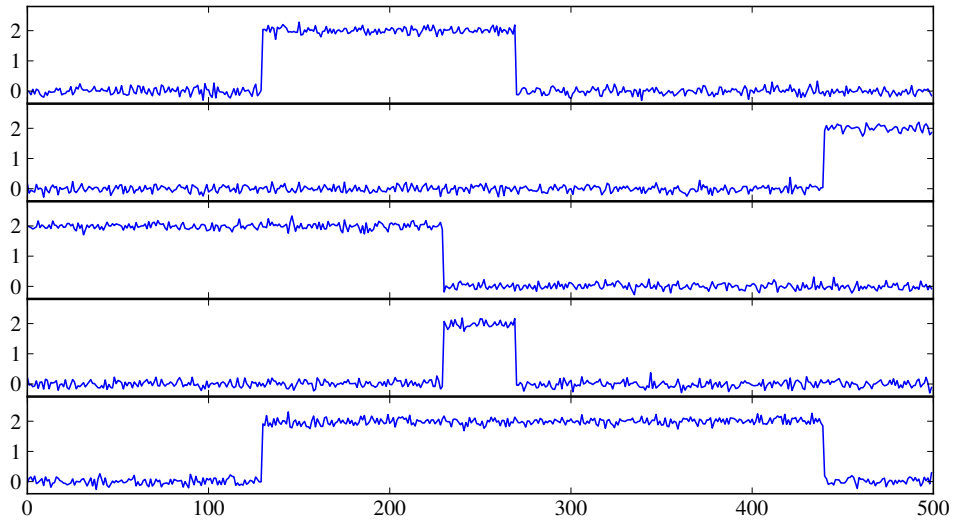


Figure 6.9 – Signal déterministe auquel on a ajouté un bruit gaussien de rapport signal sur bruit de 16 dB.

Nombre de changements connu L'algorithme d'estimation de changements multiples utilisant les rangs (que l'on appelle « DynMKW » – pour *Dynamic Multivariate Kruskal Wallis* – dans la suite de cette section) est comparé avec deux autres méthodes utilisant l'algorithme de programmation dynamique. La première (« *Linear* ») est une extension multivariée de la méthode par estimation des moindres carrés présentée au paragraphe 4.5.2. Dans la fonction de contraste Δ (6.2), le vecteur $\bar{\mathbf{R}}_{n_\ell+1:n_{\ell+1}}$ est remplacé par

$$\bar{\mathbf{S}}_{n_\ell+1:n_{\ell+1}} = \frac{1}{n_{\ell+1}-n_\ell} \sum_{i=n_\ell+1}^{n_{\ell+1}} \left(\mathbf{X}_i - \frac{1}{n} \sum_{j=1}^{n} \mathbf{X}_j \right)$$

et la matrice $\hat{\Sigma}_n$ est remplacée par un estimateur de la matrice de covariance empirique de $(\mathbf{X}_1, \ldots, \mathbf{X}_n)$. La seconde méthode (« *Kernel* ») est celle proposée par Harchaoui et Cappé (2007) pour laquelle la fonction de contraste est aussi une mesure de dispersion à l'intérieur d'un segment, mais lorsque les données sont vues comme des éléments d'un RKHS. Le noyau choisi est le noyau gaussien dont la largeur $\sigma = 1{,}06\hat{s}n^{1/5}$, $\hat{s}$ étant la variance empirique des échantillons, est suggérée par les auteurs.

Les résultats des algorithmes sont représentés dans la figure de gauche de la figure 6.10, à plusieurs niveaux de bruit. Le bruit additif étant gaussien, la méthode utilisant les rangs obtient sans surprise des résultats légèrement moins bons que ceux de la méthode paramétrique, mais meilleurs que ceux de la méthode à noyaux, particulièrement à des niveaux de bruit élevés. Dans les résultats d'expériences représentés dans le graphe de droite de la figure 6.10, on retrouve les mêmes tendances que dans les expériences de la section 6.3.2, à savoir que la méthode utilisant les rangs est plus robuste que les autres méthodes à l'ajout de valeurs aberrantes. En augmentant la tolérance sur la position à 5 échantillons, les résultats relatifs de chacun des algorithmes ne change pas ; lorsque les données ne sont pas contaminées par des valeurs aberrantes, DynMKW et l'algorithme linéaire obtiennent tous deux une précision de 0,8, contre 0,6 pour la méthode à noyaux lorsque le rapport signal à bruit marginal est de -4 dB ; la précision est supérieure à 0,95 pour toutes les méthodes à partir de 0 dB de bruit.

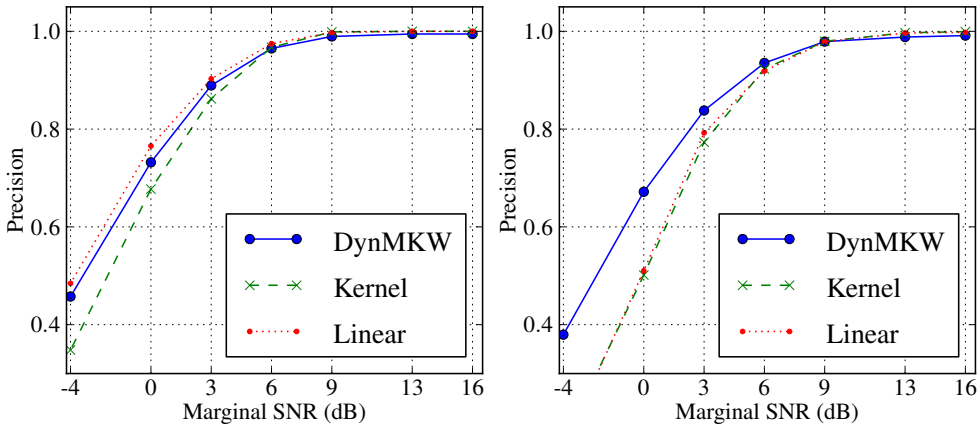


FIGURE 6.10 – Courbes de précision des algorithmes utilisant les méthodes de rang, à noyau (*kernel*) et paramétrique à différent niveaux de bruit dans les situations sans (gauche) et avec (droite) valeurs aberrantes (c'est-à-dire 5% des points ont une variance de 10 dB plus élevée que pour le bruit de fond). Les courbes de rappel, identiques à celles des courbes de précision dans le cas où le nombre de ruptures est supposé connu, ne sont pas représentées.

Nombre de changements inconnu Utilisant le même signal que précédemment, on suppose maintenant que le nombre de changements dans le signal est inconnu. Nous comparons maintenant l'algorithme d'estimation de changements multiples utilisant les rangs couplé à la méthode heuristique d'estimation du nombre de changements présenté dans la section 6.1.2 à la méthode de segmentation binaire (Vostrikova (1981), cf. section 4.5.1) couplé au test de changement d'une rupture (section 6.2), avec un seuil de décision à 1% et 5%.

La précision et le rappel de ces méthodes sont représentés dans la figure 6.11. La méthode de segmentation binaire obtient des résultats moins bons que la méthode globale de segmentation ; en augmentant la tolérance à ± 5, les résultats deviennent plus serrés, ce qui indique que DynMKW réussit à estimer la position des changements avec une grande précision.

De manière plus qualitative, la segmentation binaire a tendance à sur-segmenter le signal, alors que la méthode globale (DynMKW) manque certains changements.

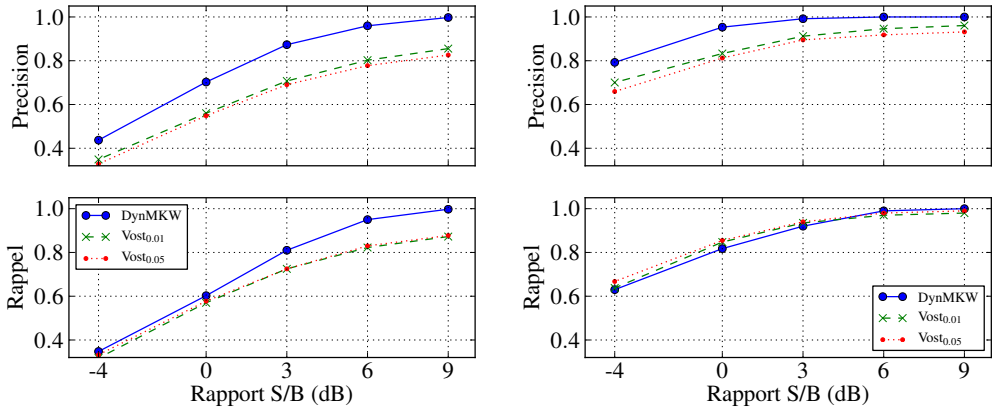


Figure 6.11 – Courbes de précision (haut) et de rappel (bas) pour la procédure *dynMKW* avec estimation heuristique du nombre de ruptures ainsi que la segmentation binaire (*Vost*) utilisée avec des seuils de détection de 0,01 et 0,05, pour une tolérance de $\pm$ 1 (gauche) et $\pm$ 5 (droite).

6.4 Démonstration du théorème 5

Dans un premier temps, on commence par prouver (6.8) lorsque l'on remplace $\hat{\Sigma}_n$ par Σ dans l'équation (6.6). Pour cela, on doit vérifier les hypothèses de Billingsley,

1968, théorème 15.6, à savoir la convergence des lois fini-dimensionnelles

$$\begin{aligned}&\Big(\mathbf{V}_n(\lfloor nt_1\rfloor)'\Sigma^{-1}\mathbf{V}_n(\lfloor nt_1\rfloor),\ldots,\mathbf{V}_n(\lfloor nt_p\rfloor)'\Sigma^{-1}\mathbf{V}_n(\lfloor nt_p\rfloor)\Big)\\&\quad\xrightarrow{d}\left(\sum_{k=1}^{K}B_k^2(t_1),\ldots,\sum_{k=1}^{K}B_k^2(t_p)\right),\quad\text{pour}\quad 0<t_1<\ldots<t_p<1\ ,\ n\to\infty\ ,\end{aligned}\tag{6.10}$$

ainsi que le critère de tension pour le processus

$$\Big\{\mathbf{V}_n(\lfloor nt\rfloor)'\Sigma_n^{-1}\mathbf{V}_n(\lfloor nt\rfloor);\ 0<t<1\Big\}\,,$$

où $\lfloor x\rfloor$ désigne la partie entière de x. Soit $n_1=\lfloor nt_1\rfloor$, avec $t_1\in(0,1)$. De la même manière que dans la démonstration de la section 5.4.1, comme $V_{n,k}(\cdot)$ ne diffère de $U_{n,k}(\cdot)$ que d'un facteur de normalisation, on peut montrer que $V_{n,k}(n_1)=\hat{V}_{n,k}(n_1)+o_p(1)$, avec $0<n_1<n$ et

$$\hat{V}_{n,k}(n_1)=\frac{n_1}{n^{3/2}}\sum_{j=n_1+1}^{n}h_{1,k}(X_{j,k})-\frac{n-n_1}{n^{3/2}}\sum_{i=1}^{n_1}h_{1,k}(X_{i,k})\ ,$$

où $h_{1,k}(x)=2F_k(x)-1$, et que

$$\mathbb{E}[\hat{V}_{n,k}(n_1)\hat{V}_{n,k'}(n_1)]\to 4t_1(1-t_1)\operatorname{Cov}\left(F_k(X_{1,k}),F_{k'}(X_{1,k'})\right),\ \text{quand}\ n\to\infty\ .\tag{6.11}$$

Soit $n_2=\lfloor nt_2\rfloor$, puisque $1<n_1<n_2<n$, $n_1/n\to t_1\in(0,1)$, et $n_2/n\to t_2\in(0,1)$ on a

$$\begin{aligned}\mathbb{E}[\hat{V}_{n,k}(n_1)\hat{V}_{n,k'}(n_2)]=\mathbb{E}\Bigg[&\left\{\frac{n_1}{n^{3/2}}\sum_{j=n_1+1}^{n}h_{1,k}(X_{j,k})-\frac{n-n_1}{n^{3/2}}\sum_{i=1}^{n_1}h_{1,k}(X_{i,k})\right\}\\&\times\left\{\frac{n_2}{n^{3/2}}\sum_{j=n_2+1}^{n}h_{1,k'}(X_{j,k'})-\frac{n-n_2}{n^{3/2}}\sum_{i=1}^{n_2}h_{1,k'}(X_{i,k'})\right\}\Bigg].\end{aligned}\tag{6.12}$$

En décomposant les intervalles $[n_1+1,n]$ en $[n_1+1,n_2]$ et $[n_2+1,n]$; et $[1,n_2]$ en $[1,n_1]$ et $[n_1+1,n_2]$ puis en développant cette expression, on obtient

$$\begin{aligned}&\mathbb{E}[\hat{V}_{n,k}(n_1)\hat{V}_{n,k'}(n_2)]=\\&\mathbb{E}\left[\frac{(n-n_1)(n-n_2)}{n^3}\sum_{i=1}^{n_1}h_{1,k}(X_{i,k})h_{1,k'}(X_{i,k'})-\frac{n_1(n-n_2)}{n^3}\sum_{j=n_1+1}^{n_2}h_{1,k}(X_{i,k})h_{1,k'}(X_{i,k'})\right.\\&\qquad\left.+\frac{n_1n_2}{n^3}\sum_{j=n_2+1}^{n}h_{1,k}(X_{i,k})h_{1,k'}(X_{i,k'})\right]\\&\qquad=\frac{n_1(n-n_2)}{n^2}\Sigma_{kk'}\longrightarrow t_1(1-t_2)\Sigma_{kk'},\ \text{quand}\ n\to\infty\ .\end{aligned}\tag{6.13}$$

Des équations (6.11) et (6.13), on obtient

$$\begin{pmatrix} \hat{\mathbf{V}}_n(n_1) \\ \hat{\mathbf{V}}_n(n_2) \end{pmatrix} \xrightarrow{d} \mathcal{N}\left(0; \left(\begin{array}{c|c} t_1(1-t_1)\Sigma & t_1(1-t_2)\Sigma \\ \hline t_1(1-t_2)\Sigma & t_2(1-t_2)\Sigma \end{array}\right)\right), \tag{6.14}$$

qui est équivalent à

$$\begin{pmatrix} \hat{\mathbf{V}}_n(n_1) \\ \hat{\mathbf{V}}_n(n_2) \end{pmatrix} \xrightarrow{d} \begin{pmatrix} \Sigma^{\frac{1}{2}} & 0 \\ 0 & \Sigma^{\frac{1}{2}} \end{pmatrix} \begin{pmatrix} \mathbf{B}(t_1) \\ \mathbf{B}(t_2) \end{pmatrix}, \tag{6.15}$$

où $\mathbf{B}(t) = (B_1(t), \ldots, B_K(t))$, $0 \leq t \leq 1$. Par souci de clarté et sans perte de généralité, on a ainsi prouvé (6.10) dans le cas où $p = 2$ après avoir appliqué la fonction continue

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \longmapsto \begin{pmatrix} x_1' x_1 \\ x_2' x_2 \end{pmatrix}, \quad \text{où } x_1, x_2 \in \mathbb{R}^K. \tag{6.16}$$

On vérifie ensuite le critère de tension, c'est-à-dire que pour $0 < t_1 < t < t_2 < 1$, on montre que

$$\begin{aligned} \mathbb{E}\Big[& |\hat{\mathbf{V}}_n(\lfloor nt \rfloor)\Sigma^{-1}\hat{\mathbf{V}}_n(\lfloor nt \rfloor) - \hat{\mathbf{V}}_n(\lfloor nt_1 \rfloor)\Sigma^{-1}\hat{\mathbf{V}}_n(\lfloor nt_1 \rfloor)|^2 \\ & \times |\hat{\mathbf{V}}_n(\lfloor nt_2 \rfloor)\Sigma^{-1}\hat{\mathbf{V}}_n(\lfloor nt_2 \rfloor) - \hat{\mathbf{V}}_n(\lfloor nt \rfloor)\Sigma^{-1}\hat{\mathbf{V}}_n(\lfloor nt \rfloor)|^2 \Big] \leq C|t_2 - t_1|^2, \end{aligned} \tag{6.17}$$

où C est une constante strictement positive. Soit

$$\mathbf{x}_n(t) = (x_{n,1}(t), \ldots, x_{n,K}(t))' = A\hat{\mathbf{V}}_n(\lfloor nt \rfloor),$$

où $A = \Sigma^{-\frac{1}{2}}$, dont on désigne le $(p,q)^{\text{e}}$ élément par $a_{p,q}$. Le premier membre de l'inégalité (6.17) peut ainsi être réécrit

$$\begin{aligned} & \mathbb{E}\left[|\mathbf{x}_n'(t)\mathbf{x}_n(t) - \mathbf{x}_n'(t_1)\mathbf{x}_n(t_1)|^2 |\mathbf{x}_n'(t_2)\mathbf{x}_n(t_2) - \mathbf{x}_n'(t)\mathbf{x}_n(t)|^2\right] \\ & \qquad = \mathbb{E}\left[\left|\sum_{k=1}^K \{x_{n,k}^2(t) - x_{n,k}^2(t_1)\}\right|^2 \left|\sum_{k'=1}^K \{x_{n,k'}^2(t_2) - x_{n,k'}^2(t)\}\right|^2\right]. \end{aligned} \tag{6.18}$$

Notons que

$$\begin{aligned} & \sum_{k=1}^K \{x_{n,k}^2(t) - x_{n,k}^2(t_1)\} = \sum_{k=1}^K (x_{n,k}(t) - x_{n,k}(t_1))(x_{n,k}(t) + x_{n,k}(t_1)) \\ & = \sum_{k=1}^K \left(\sum_{p=1}^K a_{k,p}[\hat{V}_{n,p}(\lfloor nt \rfloor) - \hat{V}_{n,p}(\lfloor nt_1 \rfloor)]\right)\left(\sum_{p'=1}^K a_{k,p'}[\hat{V}_{n,p'}(\lfloor nt \rfloor) + \hat{V}_{n,p'}(\lfloor nt_1 \rfloor)]\right) \\ & \qquad = \sum_{p,p'=1}^K b_{p,p'}\left[\hat{V}_{n,p}(\lfloor nt \rfloor) - \hat{V}_{n,p}(\lfloor nt_1 \rfloor)\right]\left[\hat{V}_{n,p'}(\lfloor nt \rfloor) + \hat{V}_{n,p'}(\lfloor nt_1 \rfloor)\right], \end{aligned} \tag{6.19}$$

où $b_{p,p'} = \sum_{k=1}^{K} a_{k,p} a_{k,p'}$ est le $(p,p')^{\text{e}}$ élément de la matrice $B = A^2 = \Sigma^{-1}$. De même

$$\sum_{k'=1}^{K} \{x_{n,k'}^2(t_2) - x_{n,k'}^2(t)\} = \sum_{q,q'=1}^{K} b_{q,q'} \left[\hat{V}_{n,q}(\lfloor nt_2 \rfloor) - \hat{V}_{n,q}(\lfloor nt \rfloor)\right] \left[\hat{V}_{n,q'}(\lfloor nt_2 \rfloor) + \hat{V}_{n,q'}(\lfloor nt \rfloor)\right]. \quad (6.20)$$

En utilisant les notations $\ell = \lfloor nt \rfloor$, $\ell_1 = \lfloor nt_1 \rfloor$ et $\ell_2 = \lfloor nt_2 \rfloor$, puis en décomposant l'intervalle $[1,n]$ en quatre sous intervalles $[1,\ell_1]$, $[\ell_1+1,\ell]$, $[\ell+1,\ell_2]$ et $[\ell_2+1,n]$, on a

$$\hat{V}_{n,p}(\ell) - \hat{V}_{n,p}(\ell_1) = \frac{\ell-\ell_1}{n^{3/2}} \left(\sum_{i=1}^{\ell_1} h_{1,p}(X_{i,p})\right) - \frac{n-(\ell-\ell_1)}{n^{3/2}} \left(\sum_{i=\ell_1+1}^{\ell} h_{1,p}(X_{i,p})\right) + \frac{\ell-\ell_1}{n^{3/2}} \left(\sum_{i=\ell+1}^{\ell_2} h_{1,p}(X_{i,p})\right) + \frac{\ell-\ell_1}{n^{3/2}} \left(\sum_{i=\ell_2+1}^{n} h_{1,p}(X_{i,p})\right) \quad (6.21)$$

et

$$\hat{V}_{n,p}(\ell) + \hat{V}_{n,p}(\ell_1) = \frac{\ell+\ell_1}{n^{3/2}} \left(\sum_{i=1}^{\ell_1} h_{1,p}(X_{i,p})\right) - \frac{n-(\ell+\ell_1)}{n^{3/2}} \left(\sum_{i=\ell_1+1}^{\ell} h_{1,p}(X_{i,p})\right) + \frac{\ell+\ell_1}{n^{3/2}} \left(\sum_{i=\ell+1}^{\ell_2} h_{1,p}(X_{i,p})\right) + \frac{\ell+\ell_1}{n^{3/2}} \left(\sum_{i=\ell_2+1}^{n} h_{1,p}(X_{i,p})\right), \quad (6.22)$$

avec des résultats similaires pour les termes de l'équation (6.20). Le terme (6.18) est l'espérance du produit des carrés de (6.19) et (6.20). En utilisant l'inégalité de Cauchy-Schwarz, (6.18) est majorée par la somme de plusieurs termes qui sont obtenus en insérant (6.21) et (6.22) dans les expressions (6.19) et (6.20), respectivement. Parmi ces termes, considérons par exemple le cas

$$C_1 \sum_{p,p'=1}^{K} \sum_{q,q'=1}^{K} b_{p,p'}^2 b_{q,q'}^2 \frac{(n-(\ell-\ell_1))^2(\ell+\ell_1)^2(n-(\ell_2-\ell))^2(\ell_2+\ell)^2}{n^{12}} \times \mathbb{E}\left[\left|\sum_{i=\ell_1+1}^{\ell} h_{1,p}(X_{i,p})\right|^2 \left|\sum_{i=1}^{\ell_1} h_{1,p}(X_{i,p})\right|^2 \left|\sum_{i=\ell+1}^{\ell_2} h_{1,p}(X_{i,p})\right|^2 \left|\sum_{i=\ell_2+1}^{n} h_{1,p}(X_{i,p})\right|^2\right]. \quad (6.23)$$

En utilisant l'indépendance des $(X_{i,k})_{1\le i\le n}$, l'espérance qui apparaît dans (6.23) peut être décomposée en un produit de quatre espérances et peut donc être bornée par

$$(\ell-\ell_1)\ell_1(\ell_2-\ell)(n-\ell_2)/3^4 \le n^2(\ell_2-\ell_1)^2/3^4. \quad (6.24)$$

On peut donc majorer (6.23) par une quantité qui est proportionnelle à $(\ell_2 - \ell_1)^2/n^2 = (\lfloor nt_2 \rfloor - \lfloor nt_1 \rfloor)^2/n^2$. Tous les termes qui apparaissent dans le développement de (6.18) peuvent être majorés de la même façon. On a ainsi montré le critère de tension en montrant (6.17), ce qui prouve que

$$\sup_{0<t<1} \mathbf{V}_n(\lfloor nt \rfloor)' \Sigma^{-1} \mathbf{V}_n(\lfloor nt \rfloor) \xrightarrow{d} \sup_{0<t<1} \sum_{k=1}^{K} B_k^2(t), \; n \to \infty \, . \tag{6.25}$$

Pour montrer (6.25) lorsque l'on remplace $\hat{\Sigma}_n$ par Σ il suffit de montrer que

$$\sup_{0<t<1} |\mathbf{V}_n(\lfloor nt \rfloor)' (\Sigma^{-1} - \hat{\Sigma}_n^{-1}) \mathbf{V}_n(\lfloor nt \rfloor)| = o_p(1).$$

Notons que

$$|\mathbf{V}_n(\lfloor nt \rfloor)' (\Sigma^{-1} - \hat{\Sigma}_n^{-1}) \mathbf{V}_n(\lfloor nt \rfloor)| \leq \|\hat{\Sigma}_n^{-1} - \Sigma^{-1}\| \sup_{0<t<1} \|\mathbf{V}_n(\lfloor nt \rfloor)\|^2 \, ,$$

où $\hat{\Sigma}_n^{-1} \xrightarrow{p} \Sigma^{-1}$ et $\sup_{0<t<1} \|\mathbf{V}_n(\lfloor nt \rfloor)\|^2 = O_p(1)$, d'après (6.25), puisque Σ est définie positive par hypothèse, ce qui conclut cette démonstration.

Chapitre 7

Applications

L'objectif de ce chapitre est double : il s'agit d'une part d'évaluer les méthodes que nous avons proposées sur des données aussi variées que possible ; d'autre part, de tirer de ces expériences des conclusions sur les propriétés des méthodes proposées ainsi que sur leur mise en œuvre pratique. Dans ce chapitre, on désigne par *MultiRank* la méthode mettant en œuvre le test de détection d'un changement qui a été proposé dans la section 6.2, et *DynMKW* la méthode d'estimation de plusieurs changements de la section 6.1.

Pour faire le lien avec la méthode proposée dans la première partie de cette thèse, le TopRank distribué, on applique MultiRank au jeu de données (simulé) utilisé dans la section 3.4 et on compare les résultats obtenus avec ceux du DTopRank. Le MultiRank est aussi appliqué dans la section 7.2 à des données économétriques dans des fenêtres glissantes, ce qui nous permet de détecter de nombreux changements sur l'ensemble des données. Il devient ainsi naturel d'évaluer la méthode d'estimation de changements multiples DynMKW sur ces mêmes données. Le DynMKW est aussi évalué dans la section 7.3 sur une application en bioinformatique où il s'agit de segmenter des données issues de micro-puces à ADN.

7.1 Détection d'anomalies réseau

Nous avons consacré la première partie de cette thèse à la détection d'attaques réseau, puis nous avons proposé dans la seconde partie une nouvelle méthode de détection de changements plus générale qui s'applique aux données multidimensionnelles. Nous évaluons le comportement de celle-ci dans l'application réseau, avec le protocole expérimental utilisé à la section 3.4, c'est-à-dire les données synthétiques. Les paramètres utilisés sont de manière générale identiques (réseau utilisé, nombre de moniteurs de $K = 15$ qui est donc la dimension maximale des séries temporelles testées) hormis pour la longueur de la fenêtre qui est désormais

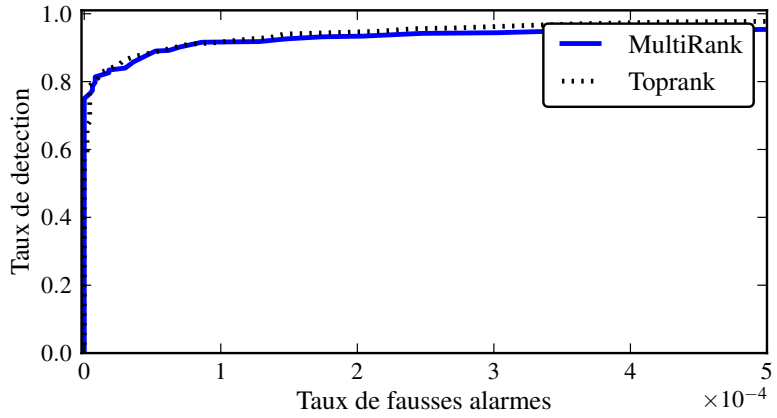
de $P = 100$, le coefficient multiplicateur η de l'intensité du processus de Poisson après changement est fixé à 1,2. L'instant de changement pour les adresses attaquées est fixé à la moitié de la fenêtre (à la position 50).

L'algorithme de détection de changement que l'on évalue est la méthode de détection d'au plus une rupture que l'on a décrit à la section 6.2. On le désigne par *MultiRank* dans cette section. On n'effectue pas d'étape de filtrage qui diminuerait la quantité de données analysées ; aussi on ne compare le MultiRank qu'avec une version de DTopRank sans censure : les trois premières étapes de la méthode présentée à la section 3.2 sont omises pour ne garder que l'étape d'agrégation par somme, suivie du test pour données unidimensionnelles utilisant la statistique (3.2). Les étapes de réduction de dimension sont cependant possibles pour DTopRank et MultiRank (en utilisant l'extension aux données censurées décrite dans le paragraphe 5.1.3), mais nous avons voulu nous concentrer ici sur une comparaison entre l'étape d'agrégation et la combinaison des statistiques marginales utilisée par MultiRank.

Il est important de noter que l'on utilise l'extension de MultiRank aux données discrètes (section 5.1.3) puisque les données sont des comptes de paquets qui sont des entiers. Par ailleurs la régularisation de la matrice de covariance des rangs et l'utilisation de la pseudo-inverse (section 5.1.2) pour le calcul de la statistique sont nécessaires. En effet dans cette application, plusieurs coordonnées des données ont de fortes chances d'être dupliquées ; il suffit par exemple que les sondes enregistrant les données se situent sur deux arêtes consécutives, ce qui est possible vu la topologie réseau utilisée (figure 3.5).

Les données générées pour cette expérience comprennent environ 500 000 adresses IP destination parmi lesquelles 500 sont attaquées et présentent une rupture. Ce déséquilibre flagrant, mais caractéristique de l'application réelle, entre exemples positifs (avec rupture) et négatifs (sans rupture) impose d'examiner les performances des algorithmes avec un très faible taux de fausses alarmes. Les méthodes MultiRank et DTopRank sont ainsi évaluées et leurs courbes ROC sont représentées dans la figure 7.1, dans la zone où l'ordre de grandeur du taux de fausses alarmes est de 10^{-4}.

Les deux méthodes utilisant les rangs que nous avons proposées obtiennent finalement des résultats plutôt similaires dans cette application malgré le fait que MultiRank utilise la structure de dépendance des données. La méthode d'agrégation par somme est en effet *conçue* pour ce type de problèmes où les données sont positives et les changements attendus se traduisent par une augmentation conjointe des différents flux de paquets destinés à une adresse cible particulière. D'une part, l'agrégation par somme et l'invariance de la méthode de rang par homothétie permettent à DTopRank de ne pas faire de détection intempestive : par exemple l'agrégation de deux séries temporelles identiques va donner la même valeur de la statistique de test que la série prise individuellement. D'autre part,



(a)

FIGURE 7.1 – Courbes ROC pour MultiRank et DTopRank.

on dispose d'une connaissance a priori importante du problème lorsque l'on sait que le type de changement que l'on cherche est le saut positif. DTopRank serait incapable de détecter la somme d'une série avec un saut positif et d'une série qui serait son opposé : la série agrégée ne présenterait alors pas de rupture. C'est un type de changement que MultiRank pourrait par contre détecter.

7.2 Données économétriques

La méthode de détection de changement introduite dans le chapitre 6 est évaluée sur une application en économétrie ; il s'agit de trouver plusieurs ruptures potentielles dans les observations. Aussi évalue-t-on la méthode d'estimation de changements multiples que l'on a proposé. On utilise le jeu de données « portfolio industriel »[1] qui a été étudié dans le contexte de la détection de ruptures par Talih et Hengartner (2005); Xuan et Murphy (2007). Ces données couvrent une période allant de 1926 à 2004. À chaque année de cette période, on assigne chacune des sociétés cotées en bourse aux États-Unis (dans le NYSE, le NASDAQ ou l'AMEX) à un portefeuille spécialisé dans un secteur d'activité. On dispose de trois jeux de données de dimensions respectives 5, 17 et 30 ; la dimension correspond au nombre de secteurs d'activité dans lesquelles sont réparties les différentes sociétés cotées. Par exemple pour les données de dimension 5, les secteurs d'activité sont l'industrie manufacturière, les produits pour consommateurs finaux, la haute-technologie, l'industrie pharmaceutique et de santé et un dernier secteur qui regroupe les sociétés qui n'entrent pas dans ces quatre catégories. Chacun des 942 points de données est alors le pourcentage de variation mensuel de ces porte-

[1] `http://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html`

feuilles d'actions. Il existe deux variantes de chacune de ces données qui diffèrent par le poids de chacune des sociétés dans chaque portefeuille. Dans la première, « AEWR » (*Average Equal Weighted Returns*), chacune des sociétés est pondérée de la même manière ; dans la seconde, « AVWR » (*Average Value Weighted Returns*), le poids assigné à une société dans le portefeuille correspond à la proportion de sa capitalisation boursière par rapport à la capitalisation totale du portefeuille. La première méthode de pondération a ainsi tendance à augmenter l'influence des petites capitalisations boursières.

Il est courant (Talih et Hengartner, 2005; Xuan et Murphy, 2007) d'appliquer une transformation ($x \mapsto \log(1 + x/100)$) à des données de ce type pour obtenir des rendements logarithmiques : cela a pour propriété de rendre symétrique les données et de les « gaussianiser ». Cette transformation, monotone, n'est pas nécessaire avec les méthodes de rang que nous utilisons.

Nous appliquons nos méthodes de détection de ruptures à ces séries temporelles de la façon suivante : on applique l'algorithme de détection d'un changement, MultiRank, à une fenêtre de $n = 200$ points, ce qui correspond à peu près à une période de 16 ans, et que l'on décale de cinquante points. Avec les données AVWR, il ne résulte de cette méthode qu'un seul changement ayant une p-valeur inférieure à 10^{-2}, quel que soit le nombre de dimensions utilisées. On obtient des résultats plus concluants avec les séries temporelles correspondant aux données AEWR – qui sont reproduites dans la figure 7.2-(a). La figure 7.2-(b) illustre les instants de rupture déterminés avec des p-valeurs significatives, à différents niveaux symbolisés par une épaisseur de trait différente. On peut constater que les ruptures détectées sont cohérentes entre les différents portefeuilles, ce qui est rassurant vu que les différents portefeuilles sont des agrégations à différents niveaux de détail des mêmes données de base. Par ailleurs, puisque l'on utilise des fenêtres glissantes avec un recouvrement, une rupture détectée dans une fenêtre est la plupart du temps aussi détectée à la même position dans une fenêtre adjacente.

On ne dispose pas d'annotations qui indiqueraient la vraie position des ruptures à déterminer. Aussi compare-t-on MultiRank au test de changement dans la moyenne pour données multivariées que l'on a décrit dans la section 4.1 que l'on applique aux données après transformation logarithmique. Les positions des ruptures ainsi détectées (figure 7.2-(c)) sont plutôt en accord avec celles détectées avec MultiRank. On obtient avec la méthode paramétrique des p-valeurs bien plus petites que celles obtenues avec MultiRank, plus particulièrement en grande dimension (portefeuilles de 17 et 30 secteurs d'activités). Cependant les ruptures détectées sont plus inconsistantes entre les trois jeux de données, en particulier au début de la fenêtre d'observation (avant l'an 1 940). On peut peut-être expliquer ce défaut par le caractère non-gaussien des données, même après la transformation logarithmique ; sur ce jeu de données, MultiRank semble plus robuste que la méthode paramétrique, même si cette dernière obtient des bonnes performances

qualitativement assez similaires.

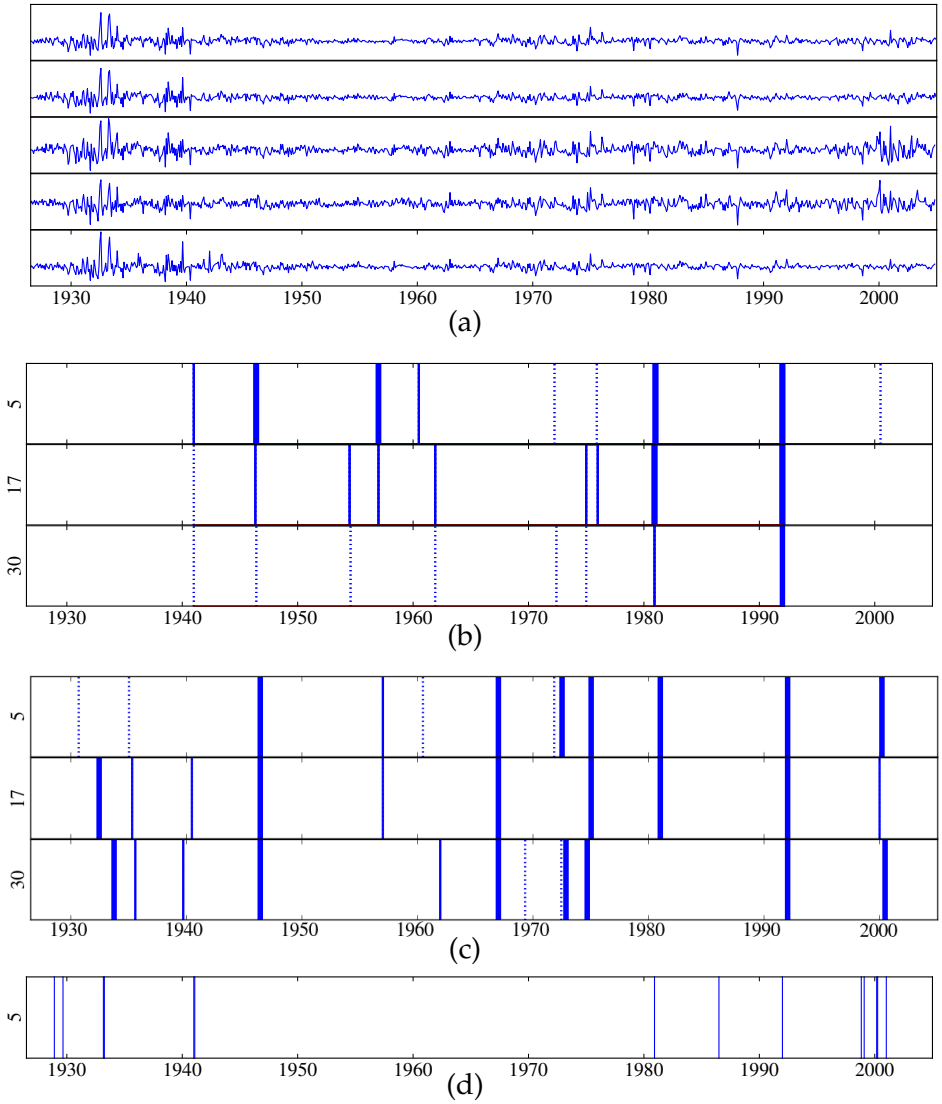


FIGURE 7.2 – (a) : Séries temporelles des rendements mensuels AEWR pour les 5 portefeuilles. (b) : Ruptures détectées par MultiRank à différents seuils de probabilités de fausses alarmes ; 0,05 (pointillés), 0,01 (traits pleins), 0,001 (gras) pour les portefeuilles de 5, 17, et 30 secteurs d'activité. (c) : Ruptures détectées avec le test du rapport de vraisemblance. (d) : Positions estimées des ruptures avec DynMKW.

Application de la méthode d'estimation de changements multiples Nous avons aussi testé la procédure d'estimation de changements multiples *DynMKW* sur ces données, en particulier le portefeuille à 5 secteurs d'activité. La première étape

consiste à tester la présence d'un changement à l'aide de la statistique MultiRank que l'on calcule sur l'intégralité des 942 points. La p-valeur obtenue est de 0,022, ce qui est assez peu significatif, pour une rupture détectée au 257^{e} point (décembre 1947). Les meilleures segmentations possibles à L segments ($1 \leq L \leq L_{\max} = 40$) sont calculées et l'heuristique de pente détermine un nombre de ruptures de 14 (figure 7.3). La figure 7.2-(d) représente la position de ces ruptures. Si quelques ruptures sont identiques à celles obtenues avec la méthode du MultiRank sur fenêtre glissante, les résultats sont plutôt qualitativement différents. On obtient d'une part quelques ruptures très rapprochées les unes des autres. Par exemple, la rupture située marquée vers l'an 1941 est en fait constituée de deux ruptures consécutives ; il en est de même pour la rupture ayant lieu juste après l'an 2000. La rupture supplémentaire est probablement un artefact, et les 14 ruptures estimées doivent en fait se réduire à un nombre de 9 ou 10. Une des manières d'éliminer de telles anomalies serait de modifier l'algorithme de programmation dynamique comme expliqué par Bai et Perron (2003) pour interdire des changements trop rapprochés. On constate aussi des ruptures éloignées d'une dizaine de points ; celles-ci ne sont pas détectées par la méthode de la fenêtre glissante. D'autre part, la méthode d'estimation globale de changements multiples n'a estimé aucun changement dans la période 1941–1980.

On peut tenter d'expliquer ces différences en examinant en détail quelques fenêtres de 200 points, dans la figure 7.4. Par exemple, la fenêtre septembre 1983 – mai 2000 est représentée à la troisième ligne de la figure 7.4 ; en 1992, un changement est détecté par l'ensemble des méthodes. On note en particulier l'existence d'un saut dans la moyenne qui est significatif par rapport à la variance des observations dans les dimensions 3 et 5.

Le saut détecté en 1946 par la méthode à fenêtre glissante n'est en revanche pas présent dans les ruptures signalées par la méthode globale. Un saut est en effet visible à l'œil nu à la deuxième ligne de la figure 7.4 en examinant la fenêtre de 200 points. Mais nous pensons que lorsque le signal est pris dans sa globalité (sur les 942 points), ce saut devient insignifiant du fait du niveau comparativement faible des variations des signaux dans la période 1940–1970 (cf. figure 7.2-(a)).

On remarque par ailleurs, sur les figures de gauche qui illustrent les cinq dimensions des séries temporelles correspondant aux portefeuilles de 5 industries, que les différentes dimensions sont très corrélées entre elles.

Ces exemples permettent ainsi de mettre en évidence des différences importantes entre un algorithme qui étudie le signal dans sa totalité et un algorithme à fenêtre glissante. Ce type d'algorithme, local, nécessite une connaissance a priori de la fréquence des changements – on a vu qu'utiliser un recouvrement assez faible entre deux fenêtre successives interdit des ruptures très rapprochées – et de l'échelle d'étude : doit-on prendre en compte un changement très local qui serait autrement noyé dans le signal global ? ce d'autant plus que les tests étudiés ici ont

précisément pour caractéristique d'être insensibles à l'échelle des signaux. Autrement dit dans ce cas, est-il aussi pertinent d'étudier des tendances sur une période de 80 années que de le faire sur 16 ans ?

De manière plus générale, il s'avère que les méthodes que l'on a utilisées ont des difficultés sur ce type de données. Les p-valeurs déterminées avec le test de détection d'un changement demeurent assez élevées, seul un faible nombre d'entre elles prennent des valeurs inférieures à 10^{-3}, c'est-à-dire sont réellement significatives. De plus la méthode d'heuristique de pente obtient des résultats assez imprécis : il est difficile de distinguer sur la figure 7.3 deux portions de droite de pentes différentes. La courbe de l'exemple de la section suivante (figure 7.10), sans atteindre un degré de netteté comme à l'exemple jouet de la figure 6.1, nous donne une estimation plus fiable du nombre de ruptures que celle obtenue avec les données économétriques. Une explication possible à ces difficultés est que certains éventuels « vrais » changements présents dans le signal semblent plutôt correspondre à un changement dans la variance (cf. expériences de la section 5.3.1). La fenêtre septembre 1931 – mai 1948 de la figure 7.4 illustre un tel changement ; en particulier, les boîtes à moustache des observations avant et après janvier 1940 révèlent un changement dans la variance.

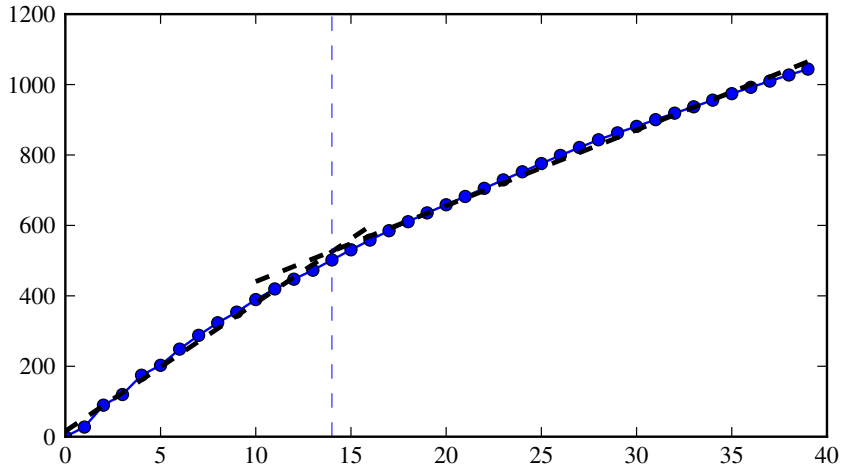


FIGURE 7.3 – Sélection du nombre de changements à l'aide de l'heuristique de pente

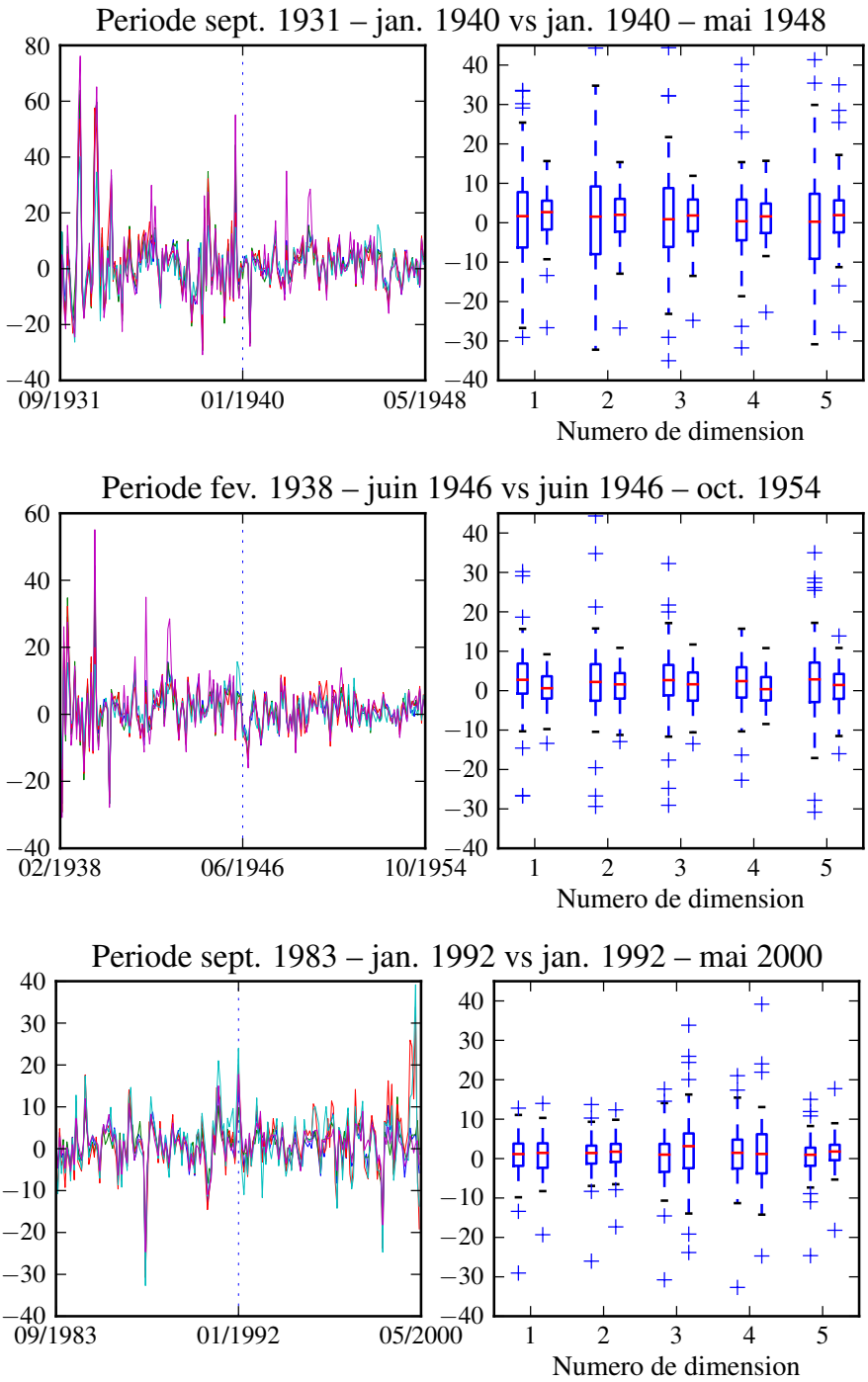


Figure 7.4 – Fenêtres d'observations de 200 points à trois périodes différentes dans les cinq dimensions du portefeuille à 5 industries, avec à droite les boîtes à moustaches associées aux parties droites et gauche de la fenêtre.

7.3 Détection de variations de nombres de copies sur micro réseaux d'ADN

Si la méthode DynMKW donnait des résultats plutôt mitigés dans l'exemple précédent, quels sont ses résultats lorsque les changements que l'on souhaite estimer s'apparentent plus nettement à des sauts dans la moyenne, comme c'est le cas dans l'application suivante ?

Un individu possède en général deux copies d'un même gène donné, une sur chaque chromosome d'une paire. Des événements peuvent cependant faire varier ce nombre : des duplications ou des délétions vont faire augmenter ou diminuer ce nombre de copies. L'hybridation génomique comparative sur micro-réseau d'ADN (*Array comparative genomic hybridisation* ou *Array CGH*) est une technique permettant d'analyser ces variations du nombre de copies de gènes dans l'ADN (*Copy number variation*). Les points de données correspondent au logarithme du rapport entre le nombre de copies constaté par une sonde (qui correspond plus ou moins à un gène connu du génome, au bruit de mesure près – certains gènes peuvent par exemple être mesurés par plusieurs sondes) sur le nombre normal (qui est égal à 2). Ainsi une valeur positive indique la duplication d'un gène et une valeur négative une délétion.

Certaines pathologies (cancer, maladies génétiques) sont responsables de variations du nombre de copies, parfois sur de larges portions d'ADN ; une pathologie donnée peut produire les mêmes effets sur plusieurs patients atteints. C'est pourquoi il est intéressant de modéliser la représentation du génome issue d'un arrayCGH par une fonction constante par morceaux et d'en effectuer la segmentation jointe entre les données de plusieurs patients atteints de la même maladie. On peut ainsi espérer fournir au biologiste un moyen de déterminer les régions de délétions ou de duplications qui seraient caractéristiques de la maladie.

La tâche effectuée consiste à segmenter les données correspondant à chacun des 22 chromosomes non-sexuels humains. Nous évaluons ainsi la méthode d'estimation de changements multiples présentée à la section 6.1 (*DynMKW*) sur deux échantillons étudiés par Bleakley et Vert (2011) dans ce même contexte. Là encore, on ne dispose pas d'annotations permettant d'évaluer les résultats obtenus, mais on compare l'algorithme que l'on propose aux algorithmes de Bleakley et Vert (2011), *group fused Lasso* (GFLasso) et son approximation rapide *GFLars*, dont on a précédemment rappelé le principe à la section 4.5.3, et dont l'implémentation a été fournie par leurs auteurs[2].

Le premier échantillon de données[3] provient de 32 patients au stade T2 du cancer de la vessie et fournit la variation du nombre de copies mesurée par 2 143 sondes. Les données correspondant à chacun des chromosomes contiennent entre

[2] `http://cbio.ensmp.fr/~jvert/svn/GFLseg/html/`

[3] disponible à `http://cbio.ensmp.fr/~frapaport/CGHfusedSVM/index.html`

$n = 50$ et 200 points et sont de dimension $K = 32$ (chacune des dimensions correspond à un patient).

La figure 7.5 illustre les données de variations du nombre de copies pour deux individus en particulier (le premier présentant de nombreuses régions de délétions ou de duplications, le second avec un profil plus régulier) avec la segmentation obtenue à l'aide de la méthode de rang et l'ensemble des approximations constantes par morceaux pour les 32 individus, à la ligne 3 de la figure 7.5 : les segmentations sont représentées par un signal constant (et égal à la moyenne des données) au sein de chaque segment estimé. Sur cette représentation des segmentations des 32 individus, on constate que de nombreux segments s'écartent simultanément chez plusieurs individus de la valeur 0. Cela indique que le modèle de segmentation jointe est plutôt pertinent.

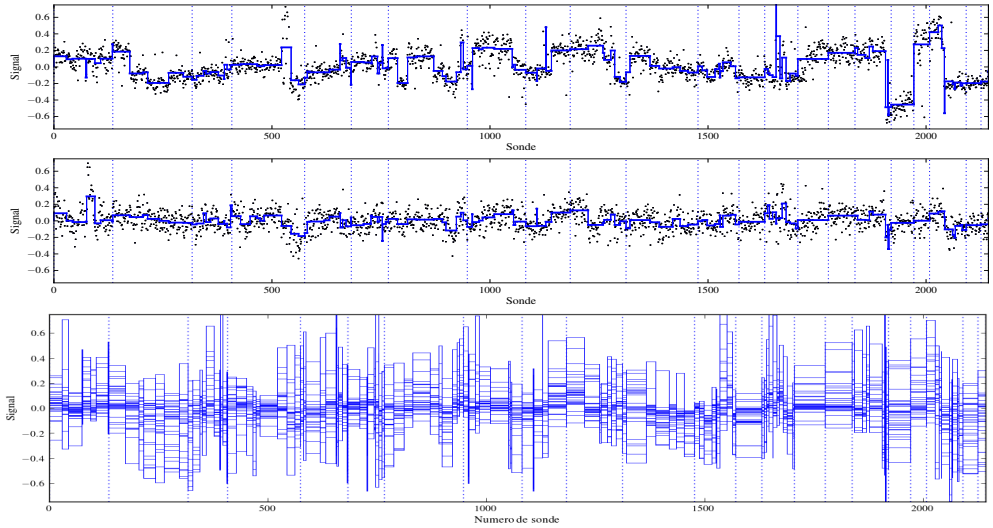


Figure 7.5 – Ligne 1 et 2 : variations du nombre de copies pour deux individus différents sur l'ensemble de leurs chromosomes ; la segmentation obtenue avec DynMKW est superposée aux données. Ligne 3 : superposition des signaux constants par morceaux résultant de la segmentation jointe pour les 32 individus. Les lignes verticales en pointillés représentent les frontières entre les différents chromosomes.

La figure 7.6 représente le résultat de la segmentation sur le chromosome 7 : l'heuristique de pente a sélectionné 10 ruptures (figure 7.7, avec $L_{\max} = 30$) et les segmentations obtenues avec GFLars et GFLasso sont données à la figure 7.8 : les résultats sont très proches. Sur l'ensemble des chromosomes les nombres de ruptures estimées sont du même ordre : 118 pour GFLars et GFLasso contre 140 pour DynMKW.

Le deuxième échantillon[4] provient de 18 patients souffrant d'un cancer des

[4] `http://cbio.ensmp.fr/~jvert/svn/GFLseg/html/`

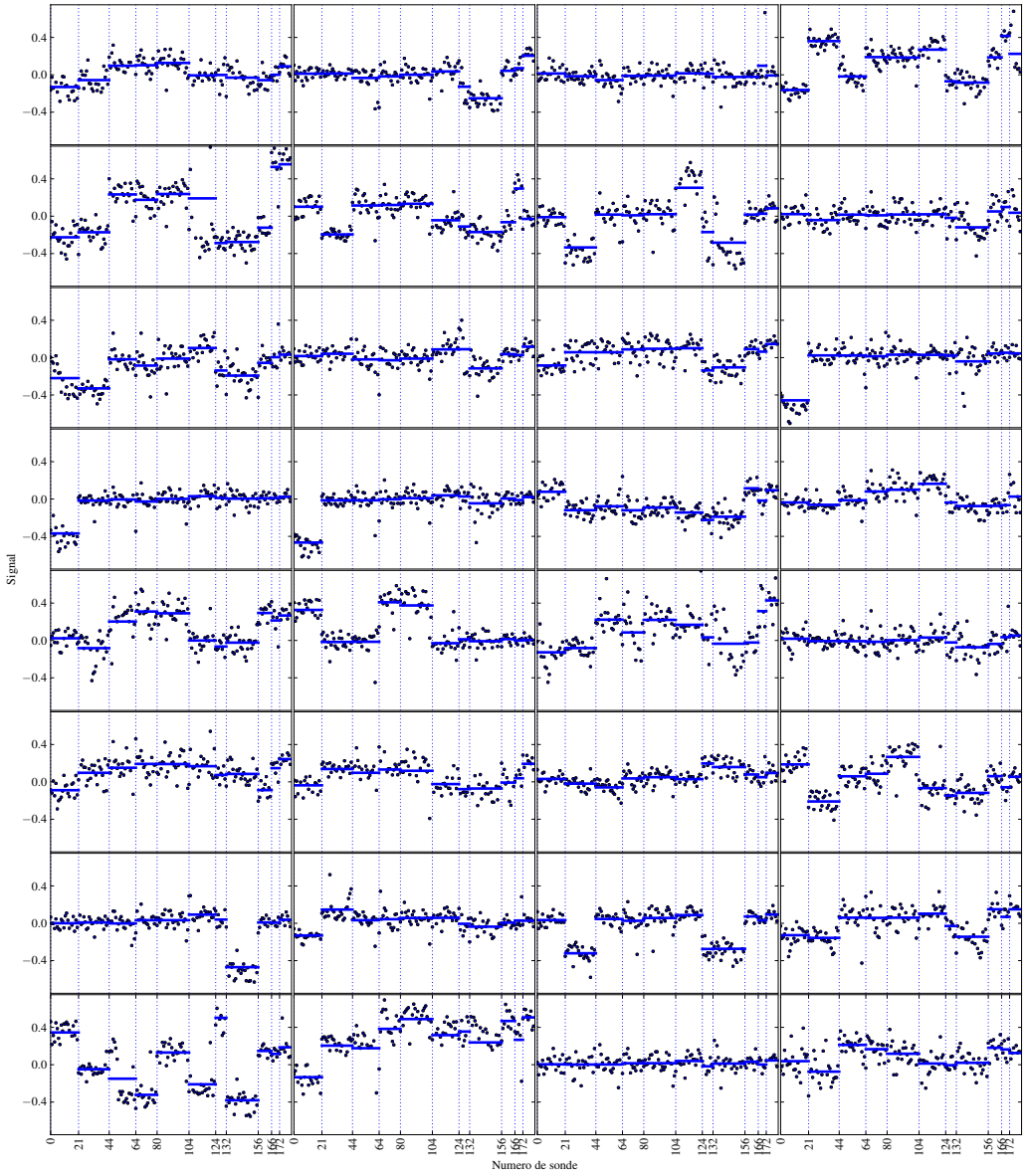


FIGURE 7.6 – Données correspondant à 32 individus au stade T2 du cancer de la vessie, sur le chromosome 7 ainsi que la segmentation estimée. 11 segments ont été estimés et les lignes verticales en pointillés correspondent aux frontières entre ceux-ci.

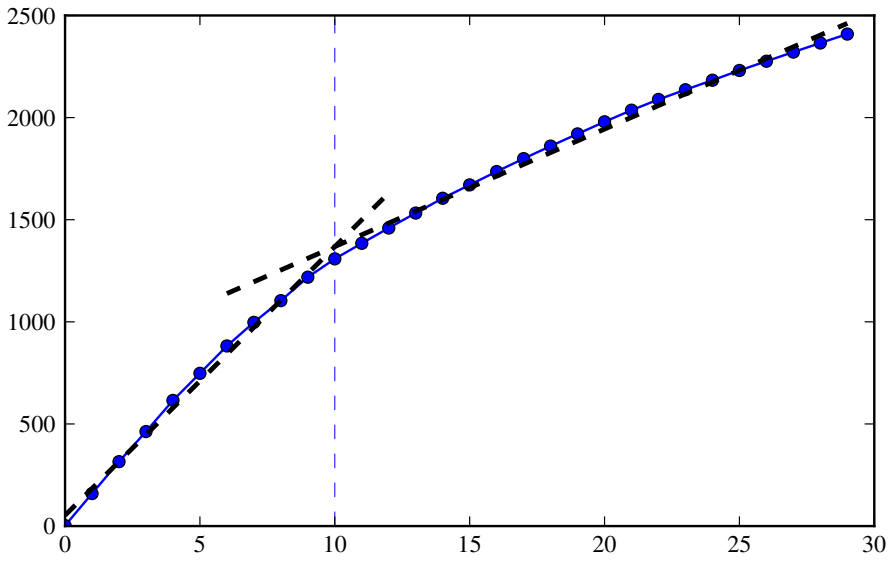


Figure 7.7 – Sélection du nombre de changements à l'aide de l'heuristique de pente pour le chromosome 7 des données « vessie ». Les points représentent la valeur de la statistique en fonction du nombre de ruptures sélectionnées ; les portions de droite en pointillés sont les estimations linéaires des parties droite et gauche de la courbe pour la valeur de L déterminée par l'algorithme.

poumons de type NSCLC (*non small cell lung cancer*, ou « non à petites cellules ») publié par Coe *et al.* (2006). Le nombre de sondes par chromosome est dans cet échantillon plus élevé que celui des données sur le cancer de la vessie : les données étudiées contiennent entre 350 et 2 500 points, pour un total de 31 708 sondes. Ces nombres plus élevés révèlent les limites des algorithmes de segmentation utilisant la programmation dynamique : les complexités en mémoire et temporelle quadratiques allongent considérablement le temps de calcul qui est beaucoup plus élevé sur les données « poumon » que sur les données « vessie ».

La figure 7.10 illustre l'heuristique de pente, et la figure 7.9 montre les données et la segmentation obtenue avec DynMKW et GFLars sur le chromosome 8. Ces segmentations sont aussi représentées avec celles de GFLasso à la figure 7.11. Les segmentations sont encore proches ; on note cependant que l'algorithme GFLasso estime plusieurs changements assez proches les uns des autres (ce qui est aussi constaté sur les données « vessie »), par exemple aux positions 723 et 743 ou 1 250 et 1 282, phénomène qui est absent avec DynMKW. Ces segmentations proches semblent refléter le saut étroit que l'on peut voir sur le premier individu de la quatrième ligne de la figure 7.9. Sur l'ensemble des chromosomes, DynMKW a estimé 173 segments contre 235 pour GFLars, différence qui peut être expliquée, pour l'essentiel, par ces estimations de segments étroits.

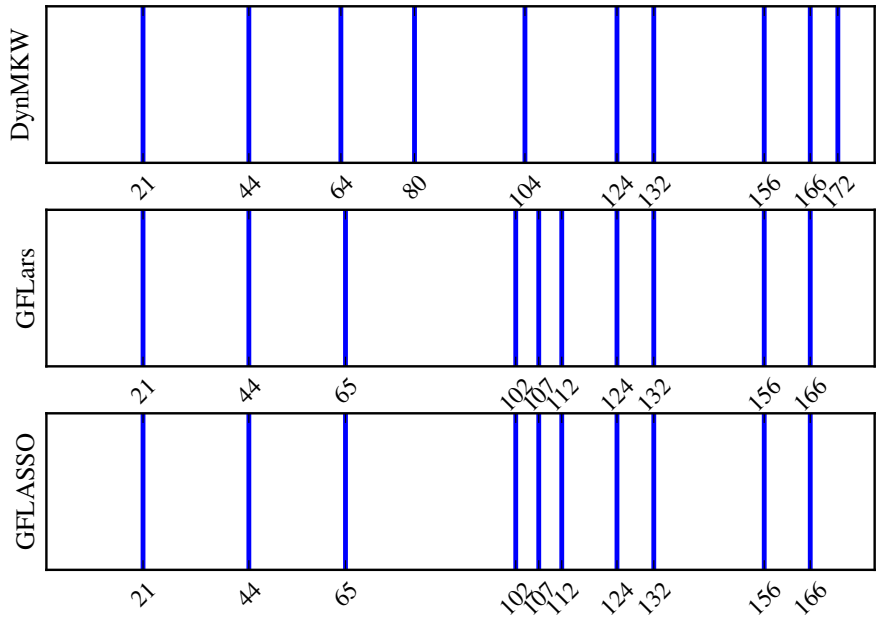


FIGURE 7.8 – Position des ruptures pour les algorithmes *DynMKW*, *GFLars* et *GFLasso* pour le chromosome 7 de l'échantillon « vessie » représenté à la figure 7.6.

L'application de DynMKW sur ces données issues d'une application en bioinformatique montre que cette méthode parvient à s'adapter à une grande variété de tailles (n allant de quelques dizaines à plusieurs milliers) et de dimensions des données. De plus, cette méthode utilisant des statistiques de rang a montré qu'elle est applicable dans un modèle où la variance des données peut varier suivant les différents segments. Cette situation est illustrée par exemple dans la figure 7.9, où la variance des observations dans le premier segment estimé est clairement différente de celle des autres segments. Dans ce type de situation, on peut penser que l'approche MultiRank, de par son caractère non paramétrique, est susceptible de donner des résultats plus fiable que les méthodes s'appuyant sur la minimisation d'un critère des moindres carrés.

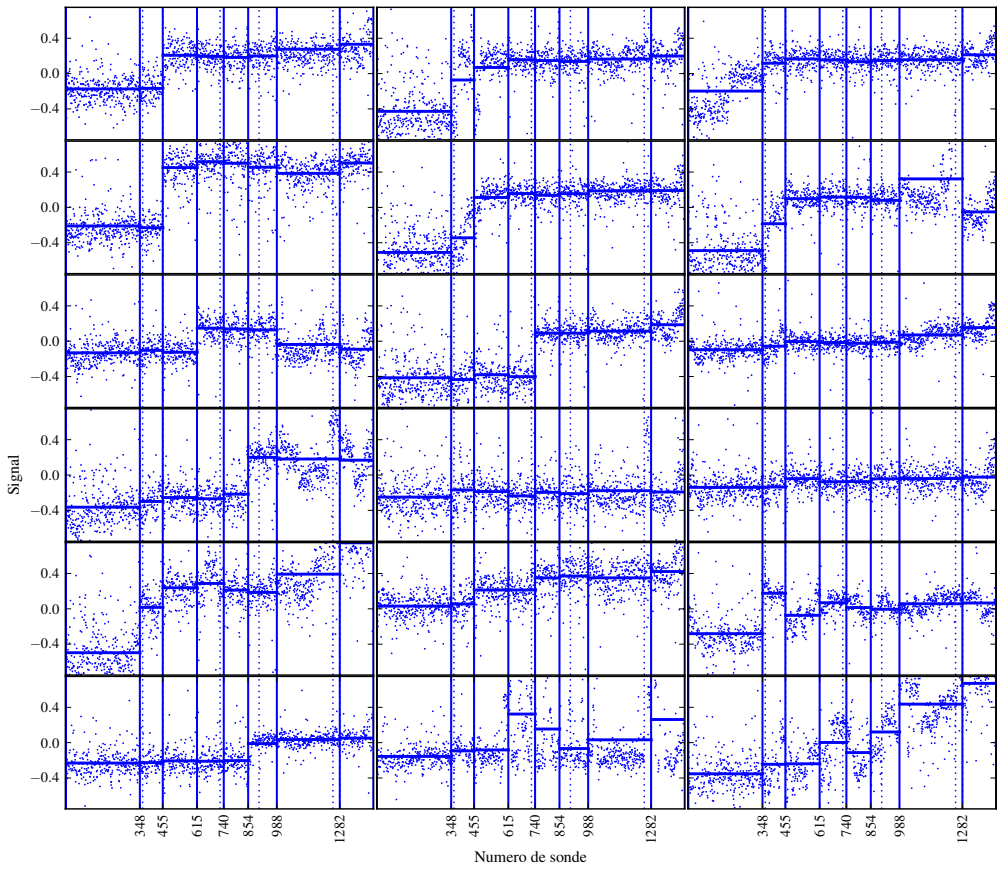


Figure 7.9 – Variation du nombre de copies sur le chromosome 8 pour 18 individus ayant le cancer du poumon NSCLC et segmentations obtenues par DynMKW (lignes verticales pleines) et l'algorithme GFLars (pointillés)

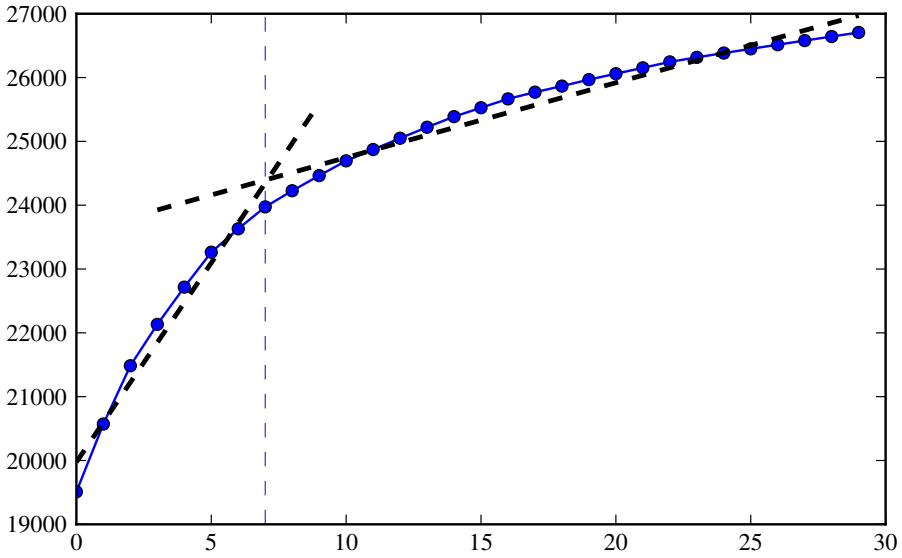


FIGURE 7.10 – Sélection du nombre de changements à l'aide de l'heuristique de pente pour le chromosome 8 des données « poumon ».

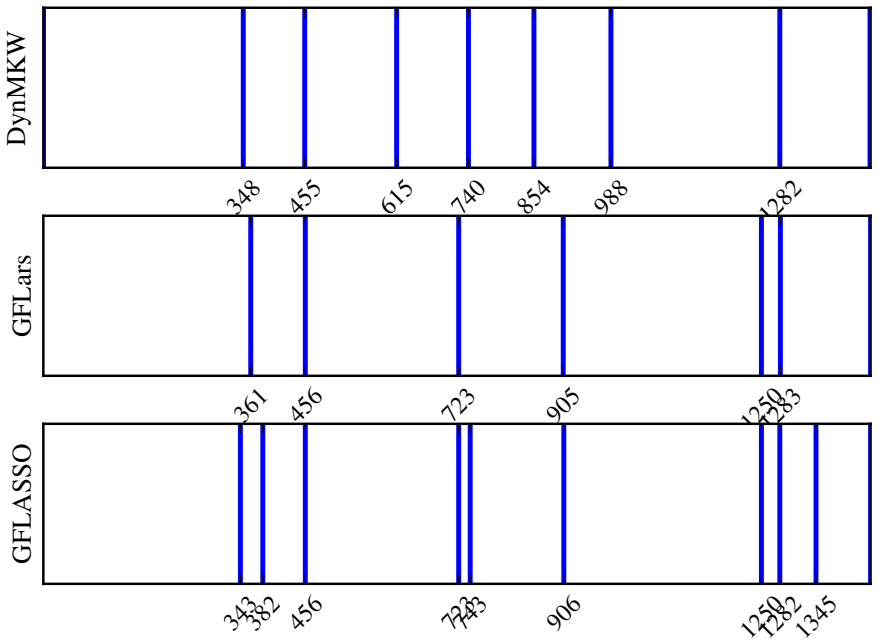


FIGURE 7.11 – Position des ruptures pour les algorithmes *DynMKW*, *GFLars* et *GFLasso* pour le chromosome 8 de l'échantillon « poumon » représenté à la figure 7.9

Conclusion

Dans cette thèse, nous avons proposé, dans un premier temps, une méthode semi-décentralisée de détection de ruptures conçue pour détecter des attaques réseau à grande échelle. Dans un second temps, nous avons proposé des tests d'homogénéité et de détection d'une rupture non-paramétriques et rétrospectifs pour données multivariées ainsi qu'une méthode d'estimation robuste de plusieurs changements au sein d'une fenêtre. Ces méthodes peuvent être vues comme des généralisations du test de rang de Wilcoxon et du test de Kruskal-Wallis, respectivement. Toutes ces méthodes reposent sur le calcul et l'utilisation des rangs relatifs des observations dans toutes les dimensions. Les lois asymptotiques des différentes statistiques de test sous l'hypothèse nulle ont été établies, permettant ainsi le calcul de p-valeurs. Nous avons montré grâce à des expériences de simulation que les procédures proposées obtiennent des résultats très satisfaisants lorsqu'elles sont utilisées pour détecter des changements dans la moyenne, et ce même lorsque les données contiennent des valeurs aberrantes, ou pour des changements multiplicatifs dans des données positives, comme l'a montré l'exemple dans les simulations réseau de la section 3.4. Les changements dans la moyenne peuvent tout à fait ne concerner qu'un nombre réduit de coordonnées, les méthodes proposées sont quand même en mesure de les détecter. D'ailleurs, le fait que les tests proposés reposent sur le calcul de statistiques dans chacune des dimensions permet l'interprétabilité des résultats : on est en mesure d'exhiber quelles sont les coordonnées dans lesquelles un changement a lieu. Il suffit pour cela de calculer le niveau de significativité de toutes les coordonnées.

Plusieurs éléments d'amélioration des méthodes proposées sont possibles. Dans la méthode du DTopRank, la décision finale est prise par un collecteur central qui recueille les données résumées envoyées par les sondes. Reposer sur une entité unique impose l'envoi de toutes les données vers cette dernière, et crée un point individuel de défaillance potentiel. Une possibilité serait d'organiser les sondes et centres de décision de manière hiérarchique. Par exemple, les sondes enverraient les séries temporelles intéressantes à un collecteur « délégué », qui à son tour enverrait ses séries agrégées à un niveau supérieur après exécution d'un test de changement. Alternativement, on pourrait envisager un modèle complètement décentralisé dans lequel les sondes communiqueraient uniquement entre

elles pour prendre une décision commune.

Concernant le test de détection d'un changement, s'il est performant au milieu de la fenêtre d'observation, il perd de la puissance lorsque les ruptures sont situées à proximité des bords de la fenêtre. Afin de pallier ce problème, nous avons suggéré dans la section 6.2.1 une renormalisation de la statistique de test qui est moins sensible à ce phénomène, au prix, éventuellement d'un léger accroissement du taux de fausse alarme. Il reste cependant un travail théorique pour confirmer une intuition sur la loi asymptotique de la statistique sous l'hypothèse nulle d'absence de changement.

De façon plus générale, nous n'avons, dans le cadre de cette thèse, abordé que la question du contrôle du taux de fausses alarmes des tests proposés. La question complémentaire de la puissance de ces tests face à différents types d'alternatives est bien évidemment d'un grand intérêt.

Nous pouvons aussi mentionner le fait que les méthodes proposées ne permettent pas de détecter n'importe quelle alternative. Par exemple, nous avons vu que MultiRank était insensible aux changements dans la variance. Il a cependant été montré dans la littérature que dans le cas unidimensionnel, en changeant le noyau h dans la U-statistique (voir la démonstration dans la section 5.4.1) on pouvait détecter les changements dans les moments d'ordre supérieur à 1 (Ferger, 1994). Il s'agirait alors de changer la U-statistique dans l'écriture du MultiRank pour obtenir d'autres statistiques plus appropriées, par exemple au cas des signaux économétriques comme ceux considérés dans la section 7.2.

Enfin, deux problématiques se dégagent concernant la méthode d'estimation de plusieurs ruptures que nous avons proposée. Premièrement, la question du contrôle du taux de fausse alarme : à nombre de ruptures connu, il faudrait déterminer la loi de la statistique sous l'hypothèse nulle afin de pouvoir évaluer le degré de pertinence des segmentations, par exemple sous la forme d'une p-valeur. Deuxièmement le choix du nombre de segments : nous avons abordé cette problématique de manière très pragmatique, mais un travail d'analyse plus approfondi reste à faire.

Annexe **A**

Segmentation de signaux issus d'un accéléromètre

Cette section est adaptée de l'article de Oudre et al. *(2011), en collaboration avec Laurent Oudre et Pascal Bianchi.*

La surveillance de la dépense énergétique est indispensable pour le suivi et le traitement des personnes obèses. Néanmoins, la plupart des méthodes fiables pour évaluer le niveau d'activité physique (telle que l'eau doublement marquée) sont onéreuses et contraignantes pour le patient. Ainsi, de plus en plus de travaux s'intéressent à des méthodes non-intrusives d'évaluation de la dépense énergétique, se reposant sur l'utilisation de capteurs tels que des accéléromètres. Une des activités qui consomme le plus d'énergie est aussi l'une des plus courantes : il s'agit de la marche. Néanmoins, la dépense énergétique en période de marche dépend énormément de la vitesse et de la pente du terrain (Terrier *et al.*, 2001). Afin d'évaluer de façon plus précise le niveau d'activité physique, il faut donc être capable de diviser une période de marche continue en différents segments de vitesse et pente constante.

Bien que de nombreuses méthodes existent pour la classification des signaux de marche selon leur pente (Aminian *et al.*, 1995; Wang *et al.*, 2009; Sekine *et al.*, 2000), elles supposent dans la plupart des cas que la segmentation a déjà été faite (le plus souvent à la main). Nous comparons ici plusieurs méthodes de segmentation existantes non spécifiques aux signaux de marche, et introduisons une nouvelle approche pour la détection de ruptures dans la vitesse et la pente basée sur un modèle de structure fréquentielle adapté à l'activité de marche.

A.1 Protocole expérimental

On a demandé à 24 sujets sains et volontaires de marcher sur un tapis roulant pendant 20 ou 25 minutes. Durant ces expériences, les sujets portaient à la ceinture et

au tibia un accéléromètre (MotionPod), développé par MOVEA, permettant d'enregistrer les accélérations selon 3 axes (vertical, médio-latéral et antéro-postérieur) avec une fréquence d'échantillonnage de 100 Hz. Un opérateur changeait soit la vitesse soit la pente du tapis environ toutes les 5 minutes (ce qui donne 3 ou 4 changements par sujet). Un exemple de séquence d'activités est présenté sur le Tableau A.1. Il faut préciser ici que les vitesses considérées ici sont appropriées pour des sujets sains, mais seraient probablement plus lentes pour des sujets âgés ou obèses.

Table A.1 – Exemple de séquence d'activités

Activité	Début	Fin
Marche à plat à 3.3 km/h	10 h 56 min 00	11 h 01 min 00
Marche à plat à 4.4 km/h	11 h 01 min 00	11 h 06 min 00
Marche à plat à 5.5 km/h	11 h 08 min 00	11 h 13 min 00
Marche en pente à 4.4 km/h (pente de 5%)	11 h 14 min 30	11 h 19 min 30
Marche en pente à 4.4 km/h (pente de 10%)	11 h 19 min 30	11 h 24 min 30

A.2 Représentations du signal

Nous avons testé deux représentations du signal, qui permettent de mettre en évidence certaines propriétés des signaux de marche.

- **A. Transformée de Fourier à court terme (TFCT).** Nous avons calculé une TFCT à partir du signal d'accélération antéro-postérieur, en ne gardant que les bins fréquentiels correspondant aux fréquences pertinentes pour des signaux de marche (0.5 Hz à 5 Hz) (Henriksen *et al.*, 2004). En pratique, les calculs ont été faits avec des fenêtres de 1 024 échantillons (soit 10 s) avec un recouvrement de 75%.

- **B. Vecteur de caractéristiques.** Il peut être intéressant de travailler avec des caractéristiques reflétant des propriétés physiologiques de l'activité de marche. Nous avons donc travaillé sur un ensemble de 12 caractéristiques calculées dans le domaine temporel à partir des trois composantes des signaux d'accélérométrie et décrites par Wang *et al.* (2009). Les signaux sont préalablement divisées en trames de 3,6 s, avec un recouvrement de 5/6. En notant respectivement a_{ML}, a_V et a_{AP} les accélérations médio-latérale, verticale et antéro-postérieure, on calcule pour chaque trame :

  - moyennes de $a_{ML} + a_V$, a_{AP} et a_V ;
  - écarts-type de $a_{AP} + a_V$ et a_{ML} ;

- médiane de a_V ;
- 95-quantile de a_{ML} ;
- nombre de changements de signe dans a_{ML} et a_V ;
- corrélations croisées entre a_{ML}, a_{AP} et a_V.

L'interprétation de ces caractéristiques dans un contexte de marche est présentée par Henriksen *et al.* (2004).

A.3 Méthodes de détection de ruptures

Méthodes 1 et 2 : détection de ruptures dans le paramètre de non-centrage d'une distribution multivariée et non-centrée du chi-2. Application à la TFCT.

Focalisons nous dans un premier temps sur la détection d'une rupture unique. La méthode que l'on utilise ici est inspirée de celle proposée par Basseville et Nikiforov (1993), reposant sur le rapport de vraisemblance généralisé. Considérons une séquence de N vecteurs aléatoires indépendants de taille F, que l'on notera $\{\mathbf{y}_n\}_{1\leq n\leq N}$. Les $\mathbf{y}_n$ correspondent ici aux modules au carré de TFCT. Supposons que les F composantes sont aussi indépendantes et que $\mathbf{y}_n$ suit une loi du χ^2 non centrée à $l = 2$ degrés de liberté et de paramètre de non-centrage $\boldsymbol{\theta}_n$. Cette densité sera notée $p_{\boldsymbol{\theta}_n}(\mathbf{y}_n)$.

Considérons tout d'abord le cas d'une rupture unique : on suppose qu'il existe au plus une rupture dans $[1 : N]$. Les différentes hypothèses peuvent être écrites de la façon suivante :

$$\mathbf{H}_0 \quad \boldsymbol{\theta}_n = \boldsymbol{\theta} \quad 1 \leq n \leq N \tag{A.1}$$

$$\mathbf{H}_k \quad \boldsymbol{\theta}_n = \boldsymbol{\theta}^0 \quad 1 \leq n \leq k \tag{A.2}$$

$$\boldsymbol{\theta}_n = \boldsymbol{\theta}^1 \quad k+1 \leq n \leq N \tag{A.3}$$

où les paramètres $\boldsymbol{\theta}$, $\boldsymbol{\theta}^0$ and $\boldsymbol{\theta}^1$ sont supposés inconnus, ainsi que l'éventuel point de rupture k. En utilisant l'hypothèse selon laquelle les données sont distribuées selon une loi du chi-2 non centrée, on peut estimer les paramètres avec :

$$\hat{\boldsymbol{\theta}}^0 = \frac{1}{k}\sum_{n=1}^{k} \mathbf{y}_n \; - l \qquad \hat{\boldsymbol{\theta}}^1 = \frac{1}{N-k}\sum_{n=k+1}^{N} \mathbf{y}_n \; - l \tag{A.4}$$

Si l'on suppose maintenant qu'il existe une et une seule rupture sur l'intervalle $[1 : N]$, l'indice de cette rupture est donné par :

$$\hat{k} = \underset{k}{\operatorname{argmax}} \sum_{n=k+1}^{N} \log \frac{p_{\hat{\boldsymbol{\theta}}^1}(\mathbf{y}_n)}{p_{\hat{\boldsymbol{\theta}}^0}(\mathbf{y}_n)} \tag{A.5}$$

Cette rupture est conservée si le rapport entre la vraisemblance généralisée qu'un changement ait lieu à $\hat{k}$ ($\mathbf{H}_k$) et la vraisemblance généralisée qu'il n'y ait pas de changement ($\mathbf{H}_0$) est supérieur à un seuil empirique. En pratique, on calcule le logarithme de ce rapport :

$$M(\hat{k}) = \sum_{n=1}^{\hat{k}} \log p_{\hat{\boldsymbol{\theta}}^0}(\mathbf{y}_n) + \sum_{n=\hat{k}+1}^{N} \log p_{\hat{\boldsymbol{\theta}}^1}(\mathbf{y}_n) - \sum_{n=1}^{N} \log p_{\hat{\boldsymbol{\theta}}}(\mathbf{y}_n). \tag{A.6}$$

On conclut que si $M(\hat{k}) > M*$ (où $M*$ est un seuil empirique choisi comme précisé dans la section A.4), alors il existe un point de rupture, et que l'indice de ce point est $\hat{k}$.

Cette approche est généralisée à la détection de plusieurs ruptures en estimant les points de rupture de façon itérative, comme décrit par Inclan et Tiao (1994) : nous appliquons notre méthode de détection de rupture unique pour estimer la première et la dernière rupture, puis nous appliquons l'algorithme de façon itérative sur la séquence comprise entre ces deux points, jusqu'à ce que plus aucune rupture ne soit détectée.

Méthode 1 : Application à une TFCT (« classique ») Appelons $x_{f,n} \in \mathbb{C}$ la valeur de la TFCT pour la trame $n \in [1 : N_f]$ et le bin fréquentiel $f \in [1 : F]$. Si l'on suppose que

$$x_{f,n} \sim \mathcal{CN}(\mu_{f,n}, \sigma^2) \tag{A.7}$$

où $\mathcal{CN}(\mu, \sigma^2)$ désigne la distribution Gaussienne circulaire complexe de moyenne μ et de variance σ^2.

Alors, si l'on note $v_{f,n} = \frac{2|x_{f,n}|^2}{\sigma^2}$ et $\theta_{f,n} = \frac{2|\mu_{f,n}|^2}{\sigma^2}$, on a :

$$v_{f,n} \sim \chi^2\left(\theta_{f,n}, 2\right). \tag{A.8}$$

où $\chi^2(\theta, l)$ désigne une loi du chi-2 non centrée à l degrés de liberté et de paramètre de non-centrage θ.

La méthode 1 consiste en l'application directe de la technique de détection de rupture précédemment décrite à la séquence d'observations $\mathbf{v}_n = [v_{1,n}, \cdots, v_{F,n}]'$.

Méthode 2 : Modélisation de la structure fréquentielle des signaux de marche (« structurée ») En observant les TFCT calculées sur des périodes de marche, on s'aperçoit que le spectre présente une série de pics fréquentiels, localisés à des multiples d'une fréquence fondamentale de bin f_0. En supposant que le spectre de la fenêtre d'analyse présente un lobe principal suffisamment étroit pour éviter le recouvrement, nous pouvons définir le modèle de structure harmonique suivant :

$$\mathbb{E}\left[v_{f,n}\right] = \sum_{h=1}^{H} \rho_{h,n} \left|W(f - hf_0)\right|^2 \tag{A.9}$$

où $W(f)$ désigne la transformée de Fourier de la fenêtre d'analyse w, H désigne le nombre total d'harmoniques et $\rho_{h,n}$ l'amplitude de l'harmonique h sur la trame n.

La méthode 2 est une variante de celle de la méthode 1, où l'équation (A.4) est remplacée par :

$$\hat{f}^0 = \underset{f}{\operatorname{argmax}} \frac{1}{k} \sum_{n=1}^{k} v_{f,n} \qquad \hat{\rho_h}^0 = \frac{1}{k} \sum_{n=1}^{k} v_{h\hat{f}^0,n} \tag{A.10}$$

$$\hat{\boldsymbol{\theta}}^0 = \sum_{h=1}^{H} \hat{\rho_h}^0 \left| W(f - h\hat{f}^0) \right|^2 - l \tag{A.11}$$

Méthode 3 : détection de ruptures non paramétrique utilisant les rangs, présentée à la section 6.1. Celle-ci peut être utilisée sur les deux représentations du signal présentées au paragraphe A.2.

A.4 Résultats

Évaluation des méthodes 1 et 2

Un exemple de détection est présenté sur la Figure A.1. On s'aperçoit que, bien que toutes les méthodes semblent parvenir à détecter les changements de vitesse (qui correspondent aux 3 premières ruptures), seule la méthode 2 (structurée) détecte le changement de pente (uniquement sur le capteur tibia). Cela peut s'expliquer par le fait qu'un changement de vitesse semble se manifester par un changement de la fréquence fondamentale, alors qu'une rupture dans la pente semble agir sur les amplitudes relatives des différentes harmoniques. Puisque la méthode 2 (structurée) estime de façon explicite ces amplitudes, il est logique qu'elle soit plus à même de repérer les changements de pente. En ce qui concerne le choix du capteur, de meilleurs résultats semblent être obtenus avec le capteur tibia.

Résultats généraux

Afin d'évaluer les différentes approches, il faut effectuer une comparaison entre la séquence de ruptures détectées et celles présentes dans les annotations. La difficulté de la tâche provient du fait que ces ruptures sont annotées soit sous la forme d'un instant précis, soit sous celle d'une plage temporelle dans laquelle la rupture est censée avoir eu lieu (cf. Table A.1). De plus ces annotations n'étant pas parfaites, nous avons introduit une tolérance pour l'évaluation. Les conventions utilisées sont les suivantes (les temps sont exprimés en secondes) :

- Si la rupture est annotée comme un temps unique t, la rupture détectée $\hat{t}$ est correcte si $\hat{t} \in [t - 30 : t + 30]$.

Table A.2 – Précisions et rappels obtenus sur le corpus de 24 sujets

		Ceinture		Tibia	
		p	r	p	r
(A) TFCT	(1) classique	0.50	**0.72**	0.49	0.77
	(2) structurée	0.46	0.65	0.49	**0.79**
	(3) dynMKW	0.50	0.67	0.51	0.71
(B) Caractéristiques	(3) dynMKW	0.51	0.69	0.57	**0.79**

- Si au contraire elle est annotée comme une plage temporelle $[t_1 : t_2]$, la rupture détectée est correcte si $\hat{t} \in [t_1 - 10 : t_2 + 10]$

On a calculé les précisions (pourcentage de ruptures correctes parmi les ruptures détectées) et les rappels (pourcentage des ruptures annotées ayant été détectées) pour les différents capteurs, représentations du signal, et méthodes de détection de rupture : ces scores sont présentés dans le tableau A.2. Le seuil empirique utilisé pour les méthodes 1 et 2 a été choisi pour donner une précision d'approximativement 50% (qui est la précision moyenne obtenue avec la méthode 3).

Ces résultats montrent qu'en travaillant sur la TFCT et le capteur tibia, les meilleurs résultats sont en effet obtenus grâce à la méthode 2 (structurée) (79% des ruptures sont détectées). Néanmoins, lorsque l'on travaille sur le capteur ceinture, il faut alors utiliser des méthodes plus générales (méthode 1 (72%)) car le modèle de structure fréquentielle est moins adapté.

L'exemple de la figure A.1 est finalement assez révélateur des performances générales. Avec les bonnes caractéristiques, on peut obtenir de bonnes performances :[1] l'algorithme utilisé parvient à détecter les quatre changement sans fausse alarme. Lorsque l'on utilise la représentation par vecteur de caractéristiques, les meilleures performances sont ainsi obtenues avec la méthode non-paramétrique 3 (dynMKW) donnant respectivement 69% et 79% de bonne détection pour les capteurs ceinture et tibia.

[1] Les séries temporelles correspondant aux différentes caractéristiques calculées ont été normalisées pour la visualisation de la figure présentée ici (la méthode utilisant les rangs n'a pas besoin d'un tel pré-traitement puisqu'elle est invariante par transformation monotone des coordonnées).

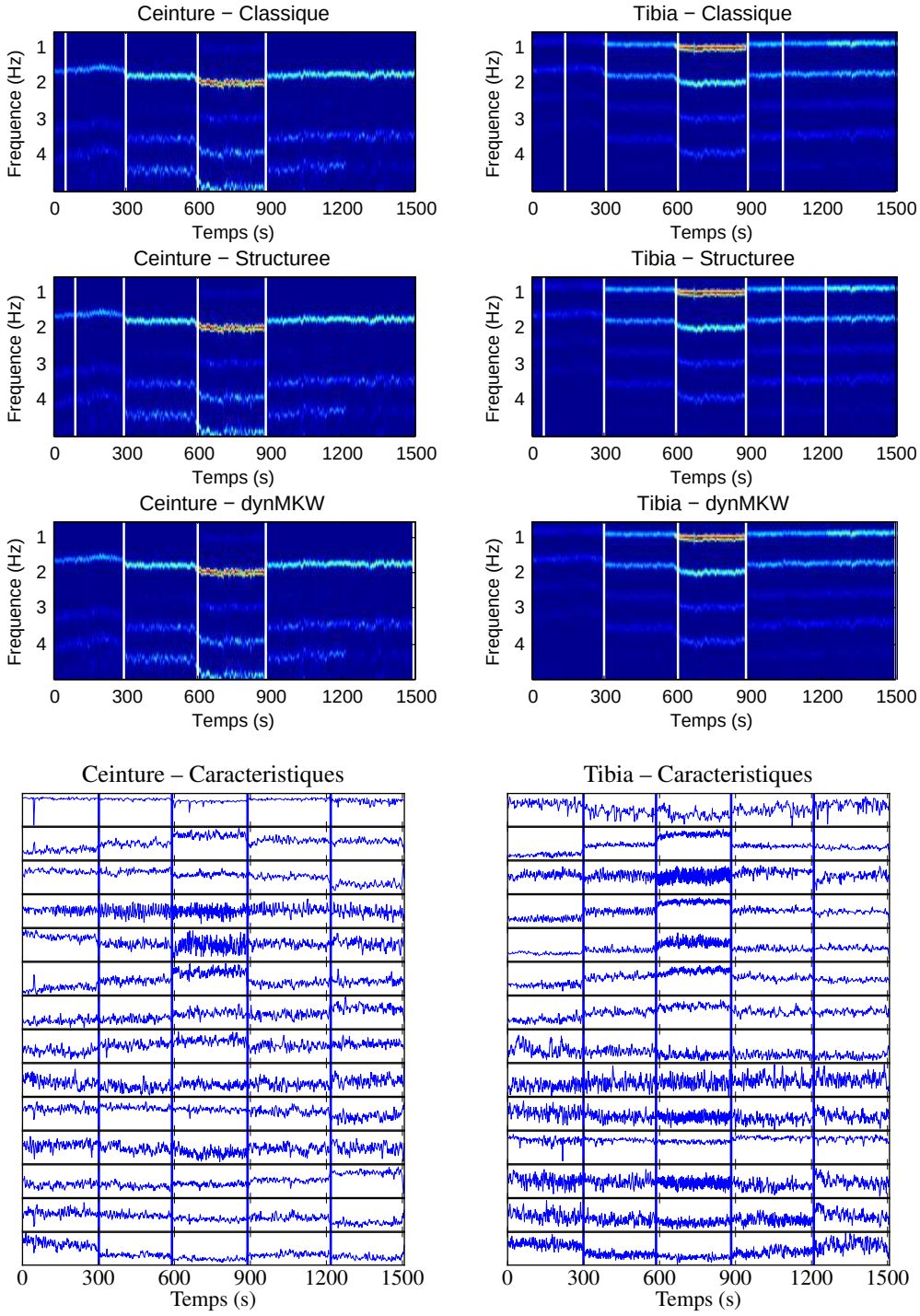


FIGURE A.1 – Ligne 1 : méthode classique sur la TFCT ; ligne 2 : méthode structurée ; ligne 3 : *dynMKW* sur la TFCT ; ligne 4 : *dynMKW* sur le vecteur de caractéristiques (qu'on aura normalisées à variance unitaire pour les représenter ici). Les ruptures présentes dans les fichiers d'annotation sont localisées à 300, 600, 900 et 1 200 secondes. Les 5 périodes ainsi définies sont les mêmes que celles présentées sur le Tableau A.1.

Bibliographie

[1] Aizerman, A., Braverman, E. et Rozoner, L. (1964). Theoretical foundations of the potential function method in pattern recognition learning. *Automation and remote control*, 25:821–837.

[2] Aminian, K., Robert, P., Jequier, E. et Schutz, Y. (1995). Estimation of speed and incline of walking using neural network. *Instrumentation and Measurement, IEEE Transactions on*, 44(3):743–746.

[3] Bai, J. et Perron, P. (1998). Estimating and testing linear models with multiple structural changes. *Econometrica*, 66(1):47–78.

[4] Bai, J. et Perron, P. (2003). Computation and analysis of multiple structural change models. *Journal of Applied Econometrics*, 18(1):1–22.

[5] Basseville, M. et Nikiforov, I. V. (1993). *Detection of Abrupt Changes: Theory and Applications*. Prentice-Hall.

[6] Bellman, R. (1961). On the approximation of curves by line segments using dynamic programming. *Communications of the ACM*, 4(6):284.

[7] Bertrand, P. R., Fhima, M. et Guillin, A. (2011). Off-line detection of multiple change points by the filtered derivative with p-value method. *Sequential Analysis*, 30(2):172–207.

[8] Billingsley, P. (1968). *Convergence of probability measures*. Wiley, New York.

[9] Bleakley, K. et Vert, J.-P. (2011). The group fused lasso for multiple change-point detection. Rapport technique. arXiv:1106.4199.

[10] Boser, B. E., Guyon, I. M. et Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. *In Proceedings of the fifth annual workshop on Computational learning theory*, COLT '92, pages 144–152, New York, NY, USA. ACM.

[11] Brodsky, B. E. et Darkhovsky, B. S. (1993). *Nonparametric Methods in Change-Point Problems*. Kluwer Academic Publisher.

[12] Chazelle, B. (2000). A minimum spanning tree algorithm with inverse-Ackermann type complexity. *Journal of the ACM*, 47:1028–1047.

[13] Chen, J. et Gupta, A. K. (2000). *Parametric statistical change point analysis*. Birkhäuser Boston Inc., Boston, MA.

[14] Coe, B., Lockwood, W., Girard, L., Chari, R., Macaulay, C., Lam, S., Gazdar, A., Minna, J. et Lam, W. (2006). Differential disruption of cell cycle pathways in small cell and non-small cell lung cancer. *British journal of cancer*, 94(12):1927–1935.

[15] Csörgő, M. et Horváth, L. (1997). *Limit theorems in change-point analysis*. Wiley, New-York.

[16] Dempster, A. P., Laird, N. M. et Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):pp. 1–38.

[17] Désobry, F., Davy, M. et Doncarli, C. (2005). An online kernel change detection algorithm. *IEEE Transactions on Signal Processing*, 53(8):2961–2974.

[18] Dijkstra, E. (1959). A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271.

[19] Dudoit, S., Shaffer, J. P. et Boldrick, J. C. (2003). Multiple hypothesis testing in microarray experiments. *Statistical Science*, 18(1):pp. 71–103.

[20] Eddy, W. (2007). TCP SYN Flooding Attacks and Common Mitigations. RFC 4987 (Informational).

[21] Efron, B., Hastie, T., Johnstone, I. et Tibshirani, R. (2004). Least angle regression. *Annals of Statistics*, 32(2):407–499. With discussion, and a rejoinder by the authors.

[22] Erdős, P. et Rényi, A. (1959). On random graphs. I. *Publ. Math. Debrecen*, 6:290–297.

[23] Fearnhead, P. (2006). Exact and efficient bayesian inference for multiple change-point problems. *Statistics and Computing*, 16:203–213.

[24] Ferger, D. (1994). Change-point estimators in case of small disorders. *Journal of statistical planning and inference*, 40(1):33–49.

[25] Friedman, J. H., Bentley, J. L. et Finkel, R. A. (1977). An algorithm for finding best matches in logarithmic expected time. *ACM Transactions on Mathematical Software*, 3:209–226.

[26] Friedman, J. H. et Rafsky, L. C. (1979). Multivariate generalizations of the wald-wolfowitz and smirnov two-sample tests. *The Annals of Statistics*, 7(4):pp. 697–717.

[27] Gehan, E. (1965). A generalized Wilcoxon test for comparing arbitrarily single censored samples. *Biometrika*, 52:203–223.

[28] Gombay, E. et Liu, S. (2000). A nonparametric test for change in randomly censored data. *The Canadian Journal of Statistics*, 28(1):113–121.

[29] Gretton, A., Borgwardt, K. M., Rasch, M., Schölkopf, B. et Smola, A. J. (2007). A kernel method for the two-sample-problem. *In* Schölkopf, B., Platt, J. et Hoffman, T., éditeurs : *Advances in Neural Information Processing Systems 19*, pages 513–520. MIT Press, Cambridge, MA.

[30] Gretton, A., Fukumizu, K., Harchaoui, Z. et Sriperumbudur, B. (2009). A fast, consistent kernel two-sample test. *In* Bengio, Y., Schuurmans, D., Lafferty, J., Williams, C. K. I. et Culotta, A., éditeurs : *Advances in Neural Information Processing Systems 22*, pages 673–681. MIT Press.

[31] Harchaoui, Z., Bach, F. et Eric, M. (2008). Testing for homogeneity with kernel Fisher discriminant analysis. *In* Platt, J., Koller, D., Singer, Y. et Roweis, S., éditeurs : *Advances in Neural Information Processing Systems 20*, pages 609–616. MIT Press, Cambridge, MA.

[32] Harchaoui, Z., Bach, F. et Moulines, E. (2009a). Kernel change-point analysis. *In* Koller, D., Schuurmans, D., Bengio, Y. et Bottou, L., éditeurs : *Advances in Neural Information Processing Systems 21*, pages 609–616. MIT Press.

[33] Harchaoui, Z. et Cappé, O. (2007). Retrospective multiple change-point estimation with kernels. *In IEEE Workshop on Statistical Signal Processing*.

[34] Harchaoui, Z. et Lévy-Leduc, C. (2010). Multiple change-point estimation with a total variation penalty. *Journal of the American Statistical Association*, 105(492): 1480–1493.

[35] Harchaoui, Z., Vallet, F., Lung-Yut-Fong, A. et Cappé, O. (2009b). A regularized kernel-based approach to unsupervised audio segmentation. *In IEEE Int. Conf. Acoust., Speech, Signal Processing*, pages 1665–1668, Taiwan.

[36] Henriksen, M., Lund, H., Moe-Nilssen, R., Bliddal, H. et Danneskiod-Samsøe, B. (2004). Test–retest reliability of trunk accelerometric gait analysis. *Gait and Posture*, 19(3):288 – 297.

[37] Henze, N. (1988). A multivariate two-sample test based on the number of nearest neighbor type coincidences. *The Annals of Statistics*, 16(2):pp. 772–783.

[38] Hero III, A. (2007). Geometric entropy minimization (GEM) for anomaly detection and localization. *In* Schölkopf, B., Platt, J. et Hoffman, T., éditeurs : *Advances in Neural Information Processing Systems 19*, pages 585–592. MIT Press, Cambridge, MA.

[39] Hotelling, H. (1931). The generalization of Student's ratio. *The Annals of Mathematical Statistics*, 2(3):360–378.

[40] Huang, L., Nguyen, X., Garofalakis, M., Jordan, M. I., Joseph, A. et Taft, N. (2007). In-Network PCA and anomaly detection. *In* Schölkopf, B., Platt, J. et Hoffman, T., éditeurs : *Advances in Neural Information Processing Systems 19*, pages 617–624. MIT Press, Cambridge, MA.

[41] Inclan, C. et Tiao, G. C. (1994). Use of cumulative sums of squares for retrospective detection of changes of variance. *Journal of the American Statistical Association*, 89(427):pp. 913–923.

[42] Jackson, J. E. et Mudholkar, G. S. (1979). Control procedures for residuals associated with principal component analysis. *Technometrics*, 21(3):pp. 341–349.

[43] Kay, S. (1993). *Fundamentals of statistical signal processing: detection theory*. Prentice-Hall, Inc.

[44] Kiefer, J. (1959). *K*-sample analogues of the Kolmogorov-Smirnov and Cramér-V. Mises tests. *Annals of Mathematical Statistics*, 30:420–447.

[45] Krishnamurthy, B., Sen, S., Zhang, Y. et Chen, Y. (2003). Sketch-based change detection: methods, evaluation, and applications. *In IMC '03: Proceedings of the 3rd ACM SIGCOMM conference on Internet measurement*, pages 234–247, New York, NY, USA. ACM.

[46] Kruskal, Joseph B., J. (1956). On the shortest spanning subtree of a graph and the traveling salesman problem. *Proceedings of the American Mathematical Society*, 7(1):pp. 48–50.

[47] Kruskal, W. et Wallis, W. (1952). Use of ranks in one-criterion variance analysis. *Journal of the American statistical Association*, 47(260):583–621.

[48] Lakhina, A., Crovella, M. et Diot, C. (2004). Diagnosing network-wide traffic anomalies. *In SIGCOMM '04: Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 219–230, New York, NY, USA. ACM.

[49] Lakhina, A., Crovella, M. et Diot, C. (2005). Mining anomalies using traffic feature distributions. *SIGCOMM Comput. Commun. Rev.*, 35:217–228.

[50] Lavielle, M. (2005). Using penalized contrasts for the change-points problems. *Signal Processing*, 85(8):1501–1510.

[51] Lavielle, M. et Moulines, E. (2000). Least-squares estimation of an unknown number of shifts in a time series. *Journal of Time Series Analysis*, 21(1):33–59.

[52] Lebarbier, E. (2005). Detecting multiple change-points in the mean of a Gaussian process by model selection. *Signal Processing*, 85:717–736.

[53] Lehmann, E. (1975). *Nonparametrics: statistical methods based on ranks*. Holden-Day Inc.

[54] Lévy-Leduc, C. et Roueff, F. (2009). Detection and localization of change-points in high-dimensional network traffic data. *Annals of Applied Statistics*, 3(2):637–662.

[55] Li, X., Bian, F., Crovella, M., Diot, C., Govindan, R., Iannaccone, G. et Lakhina, A. (2006). Detection and identification of network anomalies using sketch subspaces. *In Proceedings of the 6th ACM SIGCOMM conference on Internet measurement*, IMC '06, pages 147–152, New York, NY, USA. ACM.

[56] Mann, H. B. et Whitney, D. R. (1947). On a test of whether one of two random variables is stochastically larger than the other. *The Annals of Mathematical Statistics*, 18(1):pp. 50–60.

[57] Mantel, N. (1967). Ranking procedures for arbitrarily restricted observations. *Biometrics*, 23:65–78.

[58] Nucci, A., Sridharan, A. et Taft, N. (2005). The problem of synthetically generating IP traffic matrices: initial recommendations. *ACM SIGCOMM Computer Communication Review*, 35(3):19–32.

[59] Oudre, L., Lung-Yut-Fong, A. et Bianchi, P. (2011). Segmentation automatique de signaux issus d'un accéléromètre triaxial en période de marche. *In Proceedings of the Groupe de Recherche et d'Etudes en Traitement du Signal et des Images (GRETSI)*, Bordeaux, France.

[60] Page, E. S. (1954). Continuous inspection schemes. *Biometrika*, 41:100–115.

[61] Park, C., Hernandez-Campos, F., Marron, J. et Smith, F. D. (2005). Long-range dependence in a changing internet traffic mix. *Computer Networks*, 48(3):401 – 422. Long Range Dependent Traffic.

[62] Paxson, V. (1999). Bro: A system for detecting network intruders in real-time. *Computer Network*, 31(23–24):2435–2463.

[63] Prim, R. C. (1957). Shortest connection networks and some generalizations. *Bell Systems Technical Journal*, pages 1389–1401.

[64] Roesch, M. *et al.* (1999). Snort-lightweight intrusion detection for networks. *In Proceedings of the 13th USENIX conference on System administration*, pages 229–238. Seattle, Washington.

[65] Ruanaidh, J. et Fitzgerald, W. (1996). *Numerical Bayesian Methods Applied to Signal Processing*. Springer.

[66] Schilling, M. F. (1986). Multivariate two-sample tests based on nearest neighbors. *Journal of the American Statistical Association*, 81(395):pp. 799–806.

[67] Scholkopf, B., Platt, J., Shawe-Taylor, J., Smola, A. et Williamson, R. (2001). Estimating the Support of a High-Dimensional Distribution. *Neural Computation*, 13(7):1443–1471.

[68] Schölkopf, B. et Smola, A. (2002). *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT Press.

[69] Sekine, M., Tamura, T., Togawa, T. et Fukui, Y. (2000). Classification of waist-acceleration signals in a continuous walking record. *Medical engineering & physics*, 22(4):285–291.

[70] Shawe-Taylor, J. et Cristianini, N. (2004). *Kernel Methods for Pattern Analysis*. Cambridge University Press.

[71] Shyu, M.-L., Chen, S.-c., Sarinnapakorn, K. et Chang, L. (2003). A novel anomaly detection scheme based on principal component classifier. *In in Proceedings of the IEEE Foundations and New Directions of Data Mining Workshop, in conjunction with the Third IEEE International Conference on Data Mining (ICDM 2003*, pages 172–179.

[72] Siris, V. A. et Papagalou, F. (2006). Application of anomaly detection algorithms for detecting SYN flooding attacks. *Computer Communications*, 29(9):1433 – 1442. ICON 2004 - 12th IEEE International Conference on Network 2004.

[73] Soule, A., Salamatian, K. et Taft, N. (2005). Combining filtering and statistical methods for anomaly detection. *In Proceedings of the 5th ACM SIGCOMM conference on Internet Measurement*, IMC '05, pages 331–344, Berkeley, CA, USA. USENIX Association.

[74] Srivastava, M. S. et Worsley, K. J. (1986). Likelihood ratio tests for a change in the multivariate normal mean. *Journal of the American Statistical Association*, 81(393):199–204.

[75] Susitaival, R., Juva, I., Peuhkuri, M. et Aalto, S. (2006). Characteristics of origin-destination pair traffic in Funet. *Telecommunication Systems*, 33:67–88. 10.1007/s11235-006-9007-z.

[76] Talih, M. et Hengartner, N. (2005). Structural learning with time-varying components: tracking the cross-section of the financial time series. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(3):321–341.

[77] Tartakovsky, A., Rozovskii, B., Blazek, R. et Kim, H. (2006). A novel approach to detection of intrusions in computer networks via adaptive sequential and batch-sequential change-point detection methods. *IEEE Transactions on Signal Processing*, 54(9).

[78] Terrier, P., Aminian, K. et Schutz, Y. (2001). Can accelerometry accurately predict the energy cost of uphill/downhill walking ? *Ergonomics*, 44(1):48–62.

[79] Thorup, M. et Zhang, Y. (2004). Tabulation based 4-universal hashing with applications to second moment estimation. *In Proceedings of the fifteenth annual ACM-SIAM symposium on Discrete algorithms*, SODA '04, pages 615–624, Philadelphia, PA, USA. Society for Industrial and Applied Mathematics.

[80] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):pp. 267–288.

[81] van der Vaart, A. W. (1998). *Asymptotic statistics*, volume 3 de *Cambridge Series in Statistical and Probabilistic Mathematics*. Cambridge University Press, Cambridge.

[82] Vert, J. et Bleakley, K. (2010). Fast detection of multiple change-points shared by many signals using group LARS. *In* Lafferty, J., Williams, C. K. I., Shawe-Taylor, J., Zemel, R. et Culotta, A., éditeurs : *Advances in Neural Information Processing Systems 23*, pages 2343–2351. MIT Press.

[83] Vostrikova, L. Y. (1981). Detecting disorder in multidimensional random processes. *Soviet Math. Dokl.*, 24:55–59.

[84] Wald, A. et Wolfowitz, J. (1940). On a test whether two samples are from the same population. *The Annals of Mathematical Statistics*, 11(2):147–162.

[85] Wang, H., Zhang, D. et Shin, K. G. (2002). Detecting SYN flooding attacks. *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, 3:1530–1539.

[86] Wang, N., Ambikairajah, E., Redmond, S., Celler, B. et Lovell, N. (2009). Classification of walking patterns on inclined surfaces from accelerometry data. *In Digital Signal Processing, 2009 16th International Conference on*, pages 1 –4.

[87] Wei, L. J. et Lachin, J. M. (1984). Two-sample asymptotically distribution-free tests for incomplete multivariate observations. *Journal of the American Statistical Association*, 79(387):653–661.

[88] Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6):pp. 80–83.

[89] Xuan, X. et Murphy, K. (2007). Modeling changing dependency structure in multivariate time series. *In ICML '07: Proceedings of the 24th international conference on Machine learning*, pages 1055–1062, New York, NY, USA. ACM.

[90] Yao, Y.-C. (1988). Estimating the number of change-points via Schwarz' criterion. *Statistics & Probability Letters*, 6(3):181 – 189.

[91] Zhao, M. et Saligrama, V. (2009). Anomaly detection with score functions based on nearest neighbor graphs. *In* Bengio, Y., Schuurmans, D., Lafferty, J., Williams, C. K. I. et Culotta, A., éditeurs : *Advances in Neural Information Processing Systems 22*, pages 2250–2258. MIT Press, Cambridge, MA.

Publications

Les travaux menés durant cette thèse ont fait l'objet des publications suivantes :

[92] Harchaoui, Z., Vallet, F., Lung-Yut-Fong, A. et Cappé, O. (2009). A regularized kernel-based approach to unsupervised audio segmentation. *In ICASSP 2009*, pages 1665–1668, Taiwan.

[93] Lung-Yut-Fong, A., Cappé, O., Lévy-Leduc, C. et Roueff, F. (2009a). Détection et localisation décentralisées d'anomalies dans le trafic internet. *In GRETSI*.

[94] Lung-Yut-Fong, A., Lévy-Leduc, C. et Cappé, O. (2009b). Distributed detection/localization of network anomalies using rank tests. *In IEEE Workshop on Statistical Signal Processing, 2009*, pages 749–752, Cardiff, UK.

[95] Lung-Yut-Fong, A., Lévy-Leduc, C. et Cappé, O. (2011c). Robust changepoint detection based on multivariate rank statistics. *In IEEE Int. Conf. Acoust., Speech, Signal Processing (ICASSP)*, pages 3608–3611, Prague, Czech Republic.

[96] Lung-Yut-Fong, A., Lévy-Leduc, C. et Cappé, O. (2011d). Robust retrospective multiple change-point estimation for multivariate data. *In IEEE Workshop on Statistical Signal Processing*, pages 405–408, Nice, France.

[97] Lung-Yut-Fong, A., Lévy-Leduc, C. et Cappé, O. (2011a). Estimation robuste de ruptures multiples dans un signal multivarié. *In GRETSI*, Bordeaux, France.

[98] Oudre, L., Lung-Yut-Fong, A. et Bianchi, P. (2011a). Segmentation automatique de signaux issus d'un accéléromètre triaxial en période de marche. *In GRETSI*, Bordeaux, France.

[99] Oudre, L., Lung-Yut-Fong, A. et Bianchi, P. (2011b). Segmentation of accelerometer signals recorded during continuous treadmill walking. *In European Signal Processing Conference (EUSIPCO)*, Barcelone, Espagne.

[100] Lung-Yut-Fong, A., Lévy-Leduc, C. et Cappé, O. (2012). Distributed detection/localization of change-points in high-dimensional network traffic data. *Statistics and Computing*, 22(12):485-496, March 2012.

[101] Lung-Yut-Fong, A., Lévy-Leduc, C. et Cappé, O. (2011e). Homogeneity and change-point detection tests for multivariate data using rank statistics. *Soumis*.