UNIVERSITE KASDI MERBAH OUARGLA

Faculté Des Nouvelles Technologies De L'Information et De Télécommunications

Département D'Electronique et De Télécommunications

Mémoire

MASTER PROFESSIONNEL

Domaine : Electronique

Filière : Automatique

Spécialité : Instrumentation et systèmes

Présenté par : MOUANE Mohammed lamine et BENSEDDIK Riad

L'apprentissage statistique pour le diagnostic de défauts dans un système automatique

Soutenu publiquement

Le :

Devant le jury :

Mr. CHAKOUR Chouaib	Directeur de mémoire
Mr. HAMMOUCHI Fateh	Président de jury
Mr. BENHELAL Belkhir	Examinateur

Année Universitaire : 2022 /2023

Remerciements

Nous commençons par exprimer notre profonde gratitude à Dieu pour nous avoir donné la force, le courage et la patience de faire ce travail. Ce travail a été rendu possible grâce à l'aide et au soutien de nombreuses personnes, qui nous tenons à leur exprimer notre gratitude.

Tout d'abord, nous voudrions adresser nos sincères remerciements à monsieur l’encadreur Dr. CHAKOUR Chouaib pour sa direction scientifique et technique de ce travail. Sa présence et ses précieux conseils ont été d'une grande aide pour la finalisation de notre travail. Nous lui sommes reconnaissants pour sa patience, sa disponibilité et ses conseils avisés qui ont grandement enrichi nos réflexions.

Nous tenons également à exprimer notre profonde gratitude à tous les membres de jury pour avoir accepté l'examen de ce travail. Leur évaluation et leurs expériences sont inestimables et rajoute une grande valeur à notre mémoire.

Nous sommes profondément reconnaissants à tous ceux qui ont contribué dans notre travail. Leurs efforts et leur soutien ont été indéfectibles et inestimables et nous les remercions du fond du cœur pour leur contribution à la réalisation de cette thèse.

Dédicace

Je dédie ce travail, à mes chers parents pour tous leurs sacrifices, leur amour, leur tendresse, leur soutien et leurs prières tout au long de mon parcours universitaire, qui m'ont permis d'atteindre ce niveau où je suis aujourd'hui.

A mes chers frères pour leur soutien et leurs encouragements, et à tous ceux qui ont contribué à mon arrivée là où je suis, de près ou de loin.

الملخص

في الإطار العام لمراقبة النظام ، تم تطوير العديد من طرق تشخيص الأخطاء في التعلم الآلي. يركز هذا العمل على كشف وتوطين أعطال أجهزة الاستشعار والمشغل في العمليات الصناعية. تُستخدم طرق التعلم الإحصائي لتشخيص الأخطاء ، بما في ذلك طريقة KNN (Kأقرب الجار) و k-means

يتم تشخيص الأعطال على أساس أساليب التقليدية بمشكلة التصنيف (إلى فئتين: طبيعي وغير طبيعي) ، والتي تقتصر على أنواع معينة من الأخطاء. لاحظ أنه ينبغي تقسيم استراتيجية FDI (اكتشاف الأخطاء وعزلها) الكاملة إلى اكتشاف الأخطاء وتشخيص الأخطاء. يشير اكتشاف الخطأ إلى مراقبة الحالات الشاذة أثناء تشغيل النظام ، بينما يحدد تشخيص الخطأ و نوع الخطأ وشدته بعد مرحلة الكشف. في الواقع ، من الخطأ اعتبار استراتيجية الاستثمار الأجنبي المباشر من وجهة نظر التصنيف فقط.

الهدف من هذا العمل هو الجمع بين تقنية التعلم الآلي الخاضعة للإشراف من KNN والطريقة الإحصائية لتحليل المكونات الرئيسية ACP لأسباب التشخيص التنبئي مما يسمح باكتشاف أفضل وتوطين فوري للفشل المحتمل.

الكلمات مفتاحية: التشخيص ، وأقرب جيران k ، و تحليل المكونات الرئيسية ، k-center ، و PC-KNN ،اكتشاف الأعطال والموقع

Résumé :

Dans le cadre général de la surveillance des systèmes, plusieurs méthodes de diagnostic de défauts ont été développées en apprentissage automatique. Ce travail porte sur la détection et la localisation des défauts capteurs et actionneurs dans les processus industriels. Les méthodes d'apprentissage statistique sont utilisées pour le diagnostic de défauts, notamment la méthode KNN (k-plus proches voisins) et les k-moyennes.

Le diagnostic de défaut à base de ces approches traditionnelles est assimilé à un problème de classification (en deux classes : normale et anormale), ce qui est limité à des types de défauts spécifiques. Il est à noter qu'une stratégie FDI (Fault Detection and Isolation) complète devrait être divisée en détection de défauts et diagnostic de défauts. La détection de défauts fait référence à la surveillance des anomalies pendant le fonctionnement du système, tandis que le diagnostic de défauts identifie le type et la gravité du défaut après l'étape de la détection. En fait, il est erroné de considérer la stratégie de FDI uniquement du point de vue de la classification.

L'objectif de ce travail consiste à faire une combinaison d'une technique d'apprentissage automatique supervisé KNN avec la méthode statistique de l'analyse

en composantes principales pour des raison de diagnostic prédictif permettant de meilleure détection et localisation instantanée des éventuelles défaillances.

Mots-clés : Diagnostic, k plus proches voisins, ACP, k-centre, PC-KNN, Détection et Localisation de défauts .

Abstract:

In the general framework of system monitoring, several fault diagnostic methods have been developed using machine learning. This work focuses on the detection and isolation of sensor and actuator faults in industrial processes. Statistical learning methods are used for fault diagnosis, including the KNN (k-nearest neighbors) method and k-means.

The fault diagnosis based on these traditional approaches is considered as a classification problem (with two classes: normal and abnormal), which is limited to specific types of faults. It should be noted that a complete Fault Detection and Isolation (FDI) strategy should be divided into fault detection and fault diagnosis. Fault detection refers to monitoring anomalies during system operation, while fault diagnosis identifies the type and severity of the fault after the detection step. In fact, it is incorrect to consider the FDI strategy only from the perspective of classification.

The objective of this work is to combine a supervised machine learning technique, KNN, with the statistical method of principal component analysis for predictive diagnostic purposes, enabling the instant detection and isolation of probable failures.

Keywords: Diagnosis, k nearest neighbors, PCA, k-means, PC-KNN, , Fault detection and localization.

Table des matières

Table des figures

Introduction générale

Notre société actuelle est fortement axée sur la technologie et repose de plus en plus sur la disponibilité, la sécurité et la fiabilité de systèmes technologiques de plus en plus complexes. Néanmoins, la maîtrise totale de ces systèmes demeure un enjeu crucial et ouvert. Afin d'assurer un fonctionnement optimal, il devient essentiel d'avoir recours à une stratégie de diagnostic, qui vise à identifier les causes d'une défaillance à partir des symptômes ou des signaux observés. L'objectif est d'empêcher toute propagation des défauts. Le diagnostic automatique des pannes dans un système, comprenant les étapes suivantes : surveillance du système, génération du modèle de référence, détection des anomalies, localisation des pannes, analyse et résolution. Ce processus nécessite souvent une expertise spécialisée et peut être amélioré grâce à l'utilisation de techniques avancées d'apprentissage automatique de l'intelligence artificielle.

Les méthodes basées sur les données comprennent l'analyse statistique multivariée (MVSA), l'extraction de données (data mining) et l'apprentissage automatique (machine learning), où elles sont utilisées pour extraire des caractéristiques à partir d'une énorme quantité de données historiques provenant de capteurs, puis utilisent ces caractéristiques sélectionnées pour la détection et le diagnostic de pannes. L'analyse en composantes principales (PCA), les moindres carrés partiels (PLS), l'analyse en composantes indépendantes (ICA) et l'analyse de corrélation canonique (CCA) sont des exemples d'algorithmes traditionnels de MVSA. Une autre technique prometteuse basée sur les données pour la détection de pannes est le k-plus proches voisins (FD-kNN) qui a été proposé par He et Wang. Sans hypothèse sur la distribution des données, un échantillon défectueux est détecté en mesurant sa distance, qui est simplement la somme des distances euclidiennes au carré entre cet échantillon et ses k-plus proches voisins. Ensuite, cette distance est comparée à la région de fonctionnement normal. Il est important de noter que l'avantage principal des techniques de détection de pannes basées sur les k plus proches voisins réside dans leur capacité à traiter efficacement les caractéristiques liées à la non-linéarité, la multimodalité et les données non gaussiennes.

Nous avons devisé notre mémoire en quatre chapitres :

Le premier chapitre présente l'idée générale de l'approche diagnostic avec ces principales fonctions. Les différentes méthodes de diagnostic qui sont en relation avec l'apprentissage automatique sont présentées.

Dans le deuxième chapitre, le principe de mise en application de la méthode des k-plus proches (k-NN) et la méthode des k-centres pour le diagnostic défauts est présenté. La méthode k-NN utilise la similarité entre les points de données pour détecter les défauts et mesure la distance entre chaque point et ses k voisins les plus proches pour identifier les points anormaux. La méthode k-center regroupe les points de données en K clusters en minimisant la distance entre chaque point et son centre de cluster.

La première partie du troisième chapitre aborde les principes fondamentaux de l'analyse en composantes principales (ACP) dans le contexte de la modélisation et de la réduction de la dimensionnalité des données utilisées pour le diagnostic. L'objectif est de préserver les informations essentielles contenues dans les données d'origine. Dans la seconde partie de ce chapitre, nous examinons une approche combinant la méthode des K plus proches voisins (KNN) avec l'ACP, connue sous le nom de PC-KNN. Cette combinaison améliore le processus de prise de décision tout en réduisant la complexité de calcul associée à la version classique de la méthode KNN. Par ailleurs, pour réduire davantage la complexité de calcul de l'algorithme des KNN, nous explorons également l'utilisation de la méthode des K-moyennes en combinaison avec les deux techniques précédemment mentionnées.

Le quatrième chapitre concerne l'application et la validation des différentes techniques étudies sur des données de simulation Monte Carlo d'une turbine à gaz.

1 Chapitre 1 : Diagnostic de défauts à base des techniques d'apprentissage statistiques.

1.1 Introduction

À l'heure actuelle, anticiper les dysfonctionnements et les problèmes des systèmes automatisés est devenu nécessaire pour assurer le processus de production, car toute erreur dans le système peut entraîner de grandes pertes et des dommages aux éléments humains, aux équipements utilisés et à l'environnement. Pour éviter ces problèmes, il faut diagnostiquer ces dysfonctionnements afin de les traiter.

1.2 Surveillance et diagnostic dans les systèmes automatisés

Dans le domaine de l'automatique, le système de surveillance et de diagnostic vise à surveiller en temps réel l'état du système et à détecter tout dysfonctionnement. Cela permet de minimiser les temps d'arrêt non planifiés, d'optimiser les temps de maintenance et de garantir un fonctionnement efficace du système.

Le système de surveillance et diagnostic peut être basé sur une combinaison de capteurs et de logiciels. Les capteurs sont utilisés pour mesurer différents paramètres du système, tels que la température, la pression, la tension, le courant, etc. Les données collectées par les capteurs sont ensuite traitées par les logiciels pour identifier les anomalies et les éventuels défauts.

En cas de détection d'un dysfonctionnement ou d'un défaut, le système de surveillance et diagnostic peut générer des alertes pour informer les opérateurs ou les techniciens de maintenance. La première étape dans le diagnostic de défauts est la collecte de données. Les données sont généralement collectées à partir de capteurs qui mesurent les variables pertinentes pour le processus. Les données sont ensuite prétraitées pour éliminer les bruits et les erreurs, puis sont utilisées pour entraîner un modèle de prédiction.

Un modèle de prédiction est généralement construit en utilisant des algorithmes d'apprentissage statistiques. Ces algorithmes utilisent les données d'entraînement pour identifier les relations entre les variables mesurées. Une fois le modèle de prédiction construit, il est utilisé pour prédire les défauts dans les données de test.

Figure 1.1 : Explosion dans une usine

1.3 Comment ça fonctionne le système de surveillance ?

Les erreurs dans les systèmes automatiques sont considérées comme une déviation ou une dispersion des attentes opérationnelles, et cela est causé par des dispositifs, tels que des capteurs ou des actionneurs. Les systèmes de surveillance sont basés sur trois étapes de base : détection, isolation et identification des défauts. [1]

Les étapes de traitement des erreurs

- Détection de défaut. Pour Alerte de défauts.
- Isolation de défaut. Pour déterminer la cause de l'erreur.
- Identification de défaut. Pour quantifier les erreurs.

1.4 Terminologie

Après la large diffusion du domaine du diagnostic, une terminologie unifiée a été proposée pour permettre la compréhension et la communication sur le terrain.

Défaut : Un défaut est un changement dans le comportement normal du système, et c'est une anomalie qui se produit dans les systèmes et peut parfois conduire à une panne ou à un effondrement.

Défaillance : C'est un changement qui empêche le système d'exécuter une certaine fonction, et ce problème peut être permanent ou temporaire.

Panne : Il s'agit de la panne complète du système, qui est causée par l'arrêt du système pour exécuter ses fonctions requises avec toutes les conditions de fonctionnement spécifiées.

Surveillance : La surveillance des systèmes, en vue d'assurer un bon fonctionnement, consiste à suivre et à piloter un processus automatisé. Pour assurer un bon fonctionnement.

Détection de défaut : La détection de défaut est une tâche qui consiste à identifier les anomalies ou les défaillances dans un système, un processus ou un produit. Elle peut être effectuée à l'aide de différentes techniques, telles que l'inspection visuelle, les tests non destructifs, les mesures de performance, les analyses de données, etc.

L'isolation de défaut : est le processus de localisation de la source d'un défaut dans un système ou un produit. Elle est utilisée lorsque la détection de défaut a identifié une anomalie, mais que la cause exacte de celle-ci n'est pas connue.

Identification de défaut : implique généralement une analyse approfondie de la source du défaut, ainsi que des tests et des mesures supplémentaires pour déterminer la cause exacte du problème. Cela peut inclure l'analyse de données, la simulation informatique, les tests physiques, etc .[1]

1.5 Type de défauts :

1.5.1 Défauts capteurs

Un capteur est un instrument qui convertit des grandeurs physiques en grandeurs pouvant être traitées par un ordinateur. Les capteurs sont essentiellement l'interface de sortie entre le système et l'environnement externe. Ils sont utilisés pour transmettre des informations sur l'état et le comportement interne d'un processus. Les pannes de capteur donnent donc une mauvaise image de la grandeur physique à mesurer. Pour les systèmes en boucle fermée, les mesures de ces capteurs sont utilisées pour générer des signaux de commande. Par conséquent, la présence de défauts de capteur peut donner des signaux de commande imprécis et inefficaces.

1.5.2 Défauts actionneur

Un actionneur est un élément d'une partie opérative capable de produire un phénomène physique (déplacement, dégagement de chaleur, émission de lumière, etc.) en fonction de l'énergie qu'il reçoit. Dans la plupart des cas, les actionneurs convertissent un type d'énergie en un autre.

Ainsi, les défauts des actionneurs agissent au niveau de la partie opérative. Ils s'ajoutent aux signaux de commande du système et génèrent des problèmes liés aux organes agissant sur l'état du système. .[1]

1.5.3 Défauts système

Défauts composants sont des défauts qui affectent les composants du système lui-même. Ce sont les défauts qui ne peuvent pas être classifiés ni parmi les défauts actionneurs ni parmi les défauts capteurs. Ce type de défauts correspond à une dégradation des composants du système par un changement des paramètres internes.

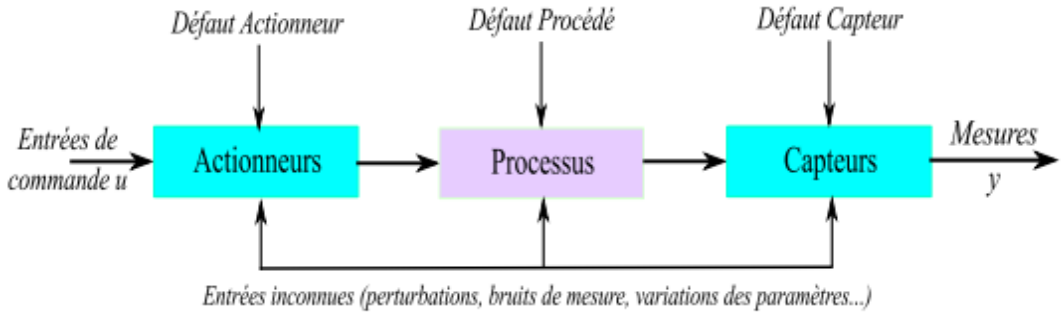

Figure 1.2 : Différents types de défauts d'un système.

1.6 Méthode de diagnostic

Voici quelques-unes des méthodes de diagnostic couramment utilisées dans le domaine du diagnostic de défauts, comme indiqué dans la figure 1.3 :

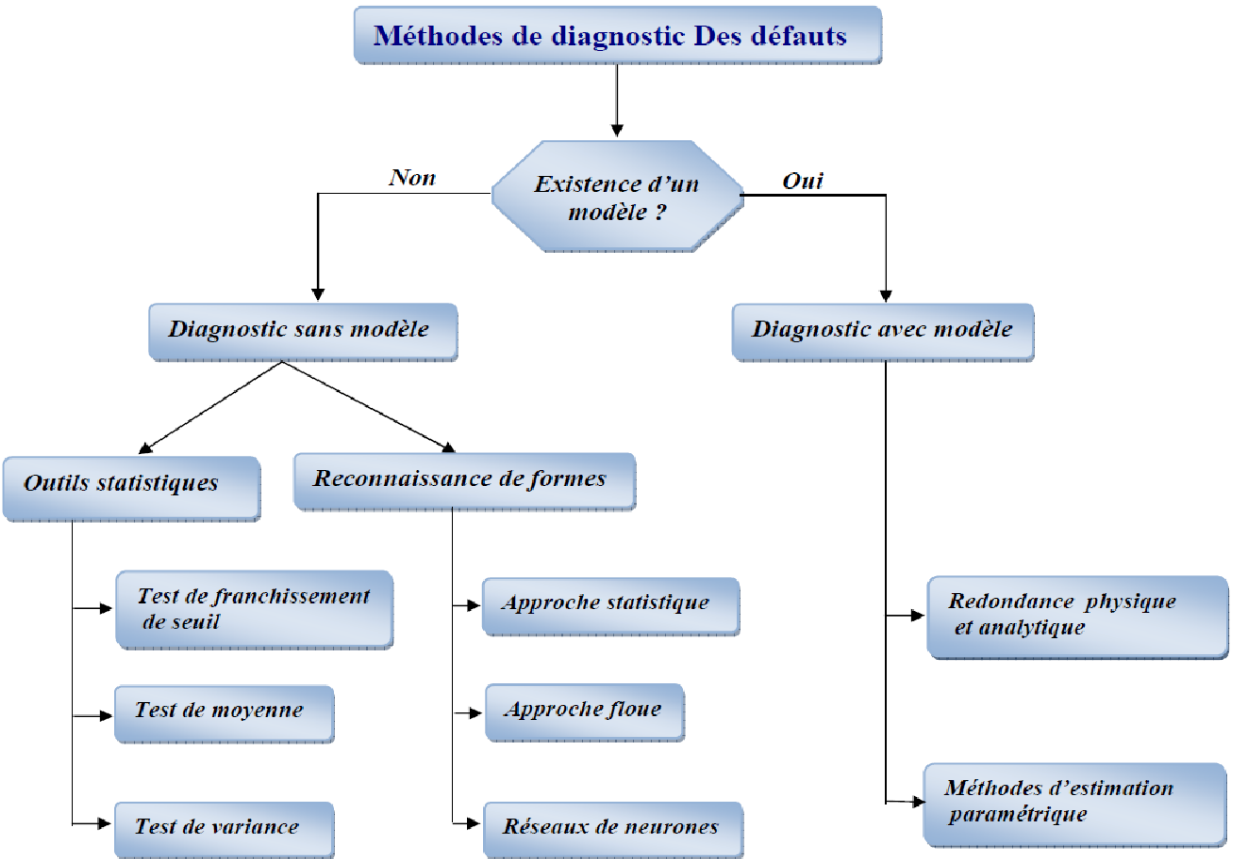

Figure 1.3 : Les méthodes de diagnostic

1.7 Redondance Matérielle

Dans un système de diagnostic basé sur des capteurs, la redondance matérielle peut être utilisée pour garantir que les capteurs sont fiables et que les données recueillies sont précises. Par exemple, en utilisant des capteurs en double ou en triple, il est possible de s'assurer que les données sont cohérentes et fiables même en cas de défaillance d'un des capteurs.

Le principe de ce processus repose principalement sur l'utilisation de plusieurs capteurs utilisés dans des mesures unifiées pour obtenir des données, donc si une erreur se produit, elle apparaît sur un seul appareil. Ce procédé se fait par vote et nécessite la présence de trois capteurs fonctionnant en même temps. Si une erreur apparaît dans un, cela dépend de deux capteurs, et de cette façon le processus de diagnostic a lieu. Comme illustré sur la figure. [4]

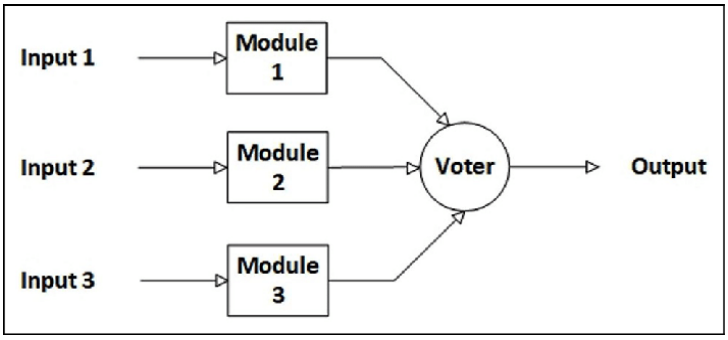

Figure 1.4 : Redondance Matérielle

Dans la figure (figure04) suivante, nous avons trois capteurs de pression qui fonctionnent sur le même processus et effectuons un processus de **Redondance Matérielle**

Figure 1.5 : exemple sur le terrain

1.8 Le diagnostic a basé des modèles

Est une méthode de diagnostic des défauts utilisant des modèles mathématiques pour identifier les problèmes dans un système automatisé. Cette méthode est souvent utilisée dans les systèmes complexes où des modèles précis et détaillés sont disponibles.

Dans cette méthode, un modèle est créé pour représenter le fonctionnement normal du système. Les mesures de paramètres sont ensuite comparées aux prédictions du modèle pour détecter les écarts, qui peuvent indiquer la présence d'un défaut. Les écarts sont souvent détectés à l'aide de techniques statistiques telles que les résidus ou les écarts moyens absolus. Lorsqu'un défaut est détecté, le modèle est utilisé pour isoler la cause sous-jacente du défaut. Les avantages de la méthode de diagnostic basée sur des modèles comprennent :

- Haute précision : Les modèles peuvent représenter précisément le comportement du système, ce qui permet une détection précise des défauts.
- Identification rapide : Les défauts peuvent être identifiés rapidement, car les écarts par rapport au modèle peuvent être détectés en temps réel.
- Localisation précise : Les modèles peuvent être utilisés pour localiser précisément la cause sous-jacente du défaut, ce qui facilite la réparation du système.

Cependant, cette méthode peut présenter certaines limitations, notamment :

- Besoin de modèles précis : Cette méthode nécessite des modèles précis et détaillés, ce qui peut être coûteux et difficile à obtenir.
- Sensibilité aux incertitudes : Les modèles peuvent être sensibles aux incertitudes et aux variations des conditions de fonctionnement réelles, ce qui peut affecter la précision du diagnostic.
- Dépendance aux mesures : Cette méthode dépend des mesures de paramètres précises pour détecter les écarts par rapport au modèle, ce qui peut être affecté par des erreurs de mesure ou des défaillances des capteurs.

1.9 Les méthodes qualitatives

Les méthodes qualitatives sont une méthode de diagnostic basée sur des modèles qui utilise des techniques de raisonnement qualitatif pour identifier les problèmes dans un système automatisé. Ces méthodes sont utilisées pour diagnostiquer des systèmes dont le comportement est difficile à modéliser mathématiquement.

Les méthodes qualitatives reposent sur la représentation des connaissances du système à l'aide de graphes de causalité ou de réseaux de connaissances. Les connaissances sont représentées sous forme de règles causales qui indiquent comment les variables du système interagissent les unes avec les autres.

Le diagnostic est effectué en analysant les relations causales entre les variables du système et les symptômes de défauts observés. Les symptômes sont identifiés à partir de mesures de paramètres ou de signaux de capteurs et peuvent être exprimés en termes de pannes, d'erreurs ou de comportements anormaux.

1.10 Les méthodes quantitatives

Ce modèle repose principalement sur des lois naturelles (physiques, chimiques et mathématiques) et ces relations décrivent les entrées et les sorties des systèmes, et ce depuis la période de développement des modèles mathématiques. Et les méthodes les plus courantes sont les méthodes suivantes : Estimations d'états, espace de parité, estimation paramétrique

1.11 Le diagnostic sans modèles

Parfois, nous rencontrons des problèmes de modélisation basée sur les lois physique du système à diagnostiquer, et cela est dû au manque de connaissances et de modèles exact qui aident au diagnostic, et dans ce cas, nous utilisons les données d'entrées/sorties reçues des outils de mesure et des capteurs.

1.11.1 L'apprentissages statistiques (l'objet du mémoire)

Les techniques d'apprentissage statistique présentent plusieurs avantages par rapport aux méthodes traditionnelles de diagnostic de défauts. Tout d'abord, ces techniques permettent une détection précoce des défauts, ce qui peut aider à éviter des défaillances coûteuses et potentiellement dangereuses. De plus, ces techniques peuvent être utilisées pour détecter des anomalies subtiles ou complexes qui pourraient être difficiles à identifier avec des méthodes traditionnelles.

En utilisant des techniques d'apprentissage statistique, il est également possible de construire des modèles prédictifs qui peuvent aider à anticiper les défauts avant qu'ils ne se produisent réellement. Un autre avantage des techniques d'apprentissage statistique est qu'elles peuvent être utilisées pour analyser des données en temps réel.

Enfin, les techniques d'apprentissage statistique peuvent être utilisées pour gérer des volumes de données très importants. Cela permet aux ingénieurs de diagnostiquer des défauts à partir de grandes quantités de données, ce qui peut être difficile à réaliser avec des méthodes traditionnelles de diagnostic de défauts.

1.11.2 L'analyse de corrélation

L'analyse de corrélation consiste à analyser les corrélations entre les variables pour identifier les relations causales entre les différentes variables et les défauts qui se produisent.

1.11.3 L'analyse des composantes principales (ACP)

L'analyse des composantes principales (ACP) : cette méthode permet d'identifier les variables qui ont le plus d'influence sur le système et d'identifier les anomalies qui peuvent être causées par des variables inattendues.

1.12 Conclusion

Le diagnostic de défauts est un processus crucial dans différents domaines d'application pour identifier et localiser les problèmes dans un système ou un appareil. Les avancées technologiques telles que l'intelligence artificielle et l'analyse des données ont amélioré l'efficacité du diagnostic. Cela permet de minimiser les temps d'arrêt, de réduire les coûts de réparation, d'améliorer la fiabilité et la sécurité des systèmes. Cependant, le diagnostic reste complexe en raison de la sophistication croissante des systèmes et des défis tels que les problèmes intermittents ou masqués. Il est donc nécessaire de continuer à développer de nouvelles approches et technologies pour améliorer le processus de diagnostic de défaut.

2 Chapitre 2 : La méthode des KNN pour la prédiction de défauts

2.1 Introduction

La méthode des K-plus proches voisins (KNN) est une technique d'apprentissage automatique couramment utilisée pour la classification des données. Plus récemment, cet algorithme a été réutilisé pour des raison de diagnostic, i.e., détection et l'isolation d'anomalies dans les données. L'algorithme a été initialement proposé en 1951 par Evelyn Fix et Joseph Hodges, qui l'ont appliqué à la classification des plantes en fonction de leurs caractéristiques mesurées.

La méthode KNN fonctionne en calculant la distance entre un point de données d'entrée et tous les autres points de données dans l'ensemble de données d'apprentissage (Training Data). Les k points de données les plus proches sont ensuite identifiés, et la classe ou la valeur de ces k points de données est utilisée pour prédire la classe ou la valeur du point de données d'entrée. Cela peut être très utile pour la détection et l'isolation des défauts dans les données, car les points de données aberrants qui diffèrent des autres points de données dans l'ensemble de données d'apprentissage peuvent être identifiés comme des données défectueuses.

2.2 Algorithme de KNN pour classification

L'algorithme des k plus proches voisins, également connu sous le nom de KNN ou k-NN, est un classificateur d'apprentissage supervisé non paramétrique, qui utilise la proximité pour effectuer des classifications ou des prédictions sur le regroupement d'un point de données individuel. Il est généralement utilisé comme algorithme de classification en partant de l'hypothèse que des points similaires peuvent être trouvés les uns à côtés des autres. Cependant, avant qu'une classification puisse être faite, la distance doit être définie. La distance euclidienne est la plus couramment utilisée, que nous aborderons plus en détail ci-dessous.

Il convient également de noter que l'algorithme KNN fait également partie d'une famille de modèles "d'apprentissage paresseux", ce qui signifie qu'il ne stocke qu'un ensemble de données d'entraînement au lieu de subir une étape d'entraînement. Cela signifie également que tous les calculs ont lieu lorsqu'une classification ou une prédiction est effectuée. Puisqu'il s'appuie fortement sur la mémoire pour stocker toutes ses données d'entraînement, il est également appelé méthode d'apprentissage basée sur les instances ou basée sur la mémoire. [6]

2.3 Comment fonctionne un algorithme KNN ?

L'algorithme KNN (K-Nearest Neighbors) est un algorithme d'apprentissage automatique qui peut être utilisé pour la classification, la régression et le regroupement. L'algorithme fonctionne en trouvant les k-voisins les plus proches d'un point de données donné, puis en classant ou en prédisant l'étiquette de ce point de données en fonction des étiquettes de ses voisins. La valeur de k est un hyper paramètre qui peut être ajusté pour optimiser les performances de l'algorithme. L'algorithme peut être combiné avec d'autres techniques telles que la méthode ACP. L'algorithme KNN est conçu pour effectuer la tâche efficacement en sélectionnant la valeur optimale du **k** à partir des résultats obtenus.

2.4 Étapes de l'algorithme de kNN

1. Tout d'abord, l'algorithme prend en entrée un nouvel échantillon de données de test pour lequel la classe doit être prédite.
2. Ensuite, l'algorithme calcul la distance entre cet exemple et tous les exemples de la base de données d'entraînement, généralement en utilisant une mesure de distance telle que la distance euclidienne.
3. L'algorithme sélectionne les K exemples les plus proches du nouvel exemple en termes de distance. Ces exemples sont appelés les K plus proches voisins.
4. Pour la classification, l'algorithme attribue la classe majoritaire parmi les K plus proches voisins au nouvel exemple. Pour la régression, l'algorithme prédit la moyenne des valeurs des K plus proches voisins.
5. Enfin, l'algorithme renvoie la classe prédite ou la valeur prédite pour le nouvel exemple.

La valeur de K est un paramètre important de l'algorithme qui doit être spécifiée avant l'exécution de l'algorithme. Si K est trop petit, la décision sera sensible aux bruits dans les données, tandis que si K est trop grand, la décision sera trop générale et ne pourra pas détecter les subtilités des données.

L'algorithme KNN est utilisé dans de nombreuses applications telles que la classification d'images, la reconnaissance de caractères manuscrits, la prédiction des prix de l'immobilier, etc.

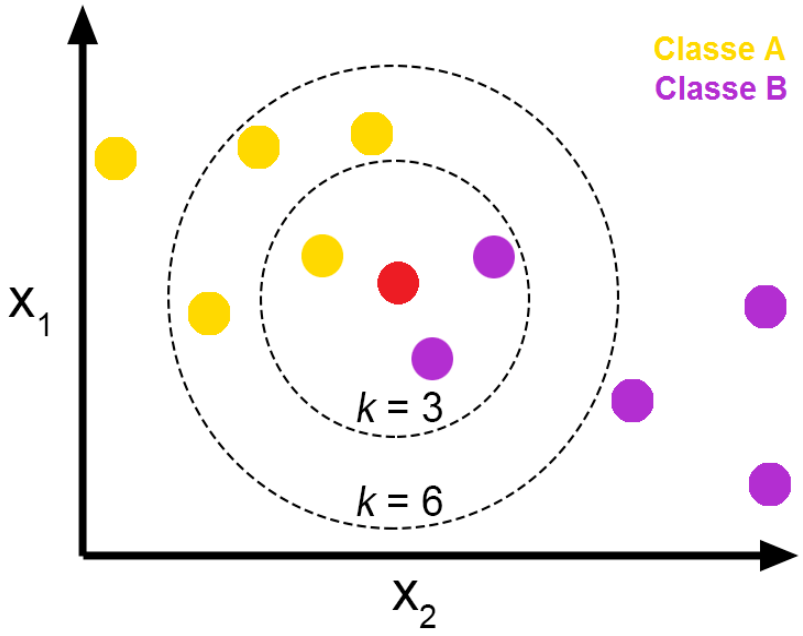

Figure 2.1: fonctionnement de l'algorithme k-NN

Afin de déterminer quels points de données sont les plus proches d'un point de requête donné, la distance entre le point de requête et les autres points de données devra être calculée. Ces métriques de distance aident à former des limites de décision, qui partitionnent les points de requête en différentes régions.

2.5 Métriques de distance dans un algorithme KNN

Distance euclidienne (p=2) : il s'agit de la mesure de distance la plus couramment utilisée, et elle est limitée aux vecteurs à valeurs réelles. En utilisant la formule ci-dessous, il mesure une ligne droite entre le point de requête et l'autre point mesuré.

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \quad \ldots\ldots\ldots (1.1)$$

Distance de Manhattan (p=1) : il s'agit également d'une autre mesure de distance populaire, qui mesure la valeur absolue entre deux points. Elle est également appelée distance en taxi ou distance d'un pâté de maisons, car elle est généralement visualisée avec une grille, illustrant comment on peut naviguer d'une adresse à une autre via les rues de la ville.

$$\boldsymbol{d_e(x,y) = \sum_{i=1}^{m} |x_i - y_i| \dots\dots\dots (1.2)}$$

Hemming distance : Cette technique est généralement utilisée avec des vecteurs booléens ou de chaîne, identifiant les points où les vecteurs ne correspondent pas. En conséquence, il a également été appelé la métrique de chevauchement. Ceci peut être représenté par la formule suivante :

$$\boldsymbol{d_H = \sum_{i=1}^{k} |x_i - y_i| \dots\dots\dots (1.3)}$$

$$x = y \quad d_H = 0$$

$$x \neq y \quad d_H \neq 1$$

2.6 Exemple de Euclidien distance

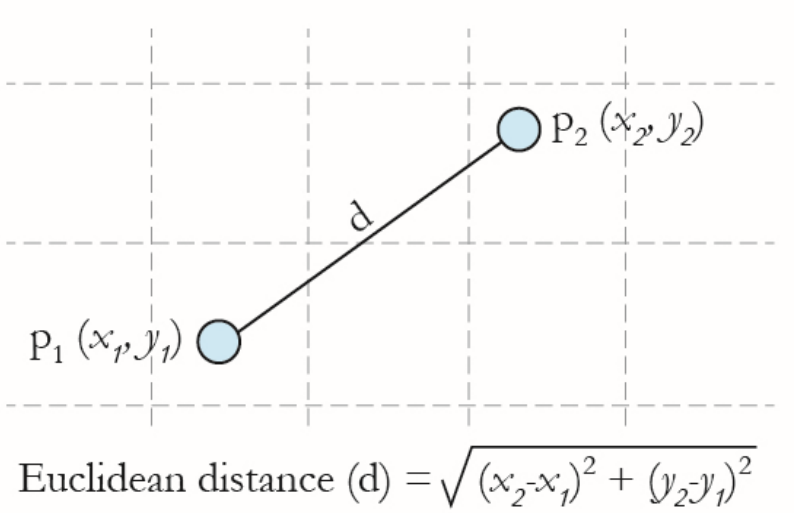

Figure 2.2 : Distance euclidienne (p=2)

2.7 Avantages

- L'algorithme est simple et facile à mettre en œuvre.
- Il n'est pas nécessaire de construire un modèle, d'ajuster plusieurs paramètres ou de faire des hypothèses supplémentaires.
- L'algorithme est polyvalent. Il peut être utilisé pour la classification, la régression et la recherche d'informations (comme nous le verrons dans la section suivante). [8]

2.8 Désavantages

- Intensif en calcul : KNN peut être intensif en calcul, en particulier lorsque vous travaillez avec de grands ensembles de données d'apprentissage. L'algorithme doit calculer la distance entre le nouveau point de données et tous les points de données d'apprentissage, ce qui peut prendre du temps.
- Sensible au choix de K : La performance de KNN est sensible au choix du nombre de plus proches voisins (K). Si K est trop petit, le modèle peut être sensible aux données bruitées, tandis que si K est trop grand, le modèle peut être trop général et ne pas capturer la structure sous-jacente des données.
- Ne convient pas aux données de grande dimension : KNN ne convient pas aux données de grande dimension car la distance entre les points de données devient moins significative dans les espaces de grande dimension. C'est ce qu'on appelle la "malédiction de la dimensionnalité".
- Nécessite un grand espace mémoire : KNN doit stocker l'intégralité de l'ensemble d'apprentissage en mémoire, ce qui peut poser problème lorsque vous travaillez avec de grands ensembles de données.

2.9 Un exemple expliquant le fonctionnement du KNN

Cet exemple montre comment trouver les indices des voisins les plus proches dans un jeu de données à l'aide d'une métrique de distance personnalisée. Plus précisément, il utilise la métrique de distance du chi carré, qui est couramment utilisée dans l'analyse des correspondances dans les applications écologiques.

L'exemple génère d'abord deux matrices, X et Y, avec des données distribuées normalement. Les lignes de X et Y correspondent aux observations et les colonnes représentent les prédicteurs. La métrique de distance chi carré entre deux points x et z de dimension j est définie comme la distance euclidienne pondérée, où les poids w_j sont associés à la dimension j. La fonction '***knnsearch***' est utilisée pour trouver les indices des k plus proches voisins dans X pour chaque observation dans Y, en utilisant la métrique de distance chi carré personnalisée. La sortie est une matrice d'indices et une matrice de distances pour chaque observation dans Y. Les indices correspondent aux indices de ligne dans X des voisins les plus proches, tandis que les distances sont les distances entre les observations correspondantes.

Enfin, l'exemple trace les points de données en X, ainsi que les voisins les plus proches pour chaque observation en X. La métrique de distance chi carré est comparée à la métrique de distance euclidienne, qui est implémentée à l'aide de l'option "euclidienne" dans kNN search. La sortie des deux implémentations s'avère pratiquement équivalente.

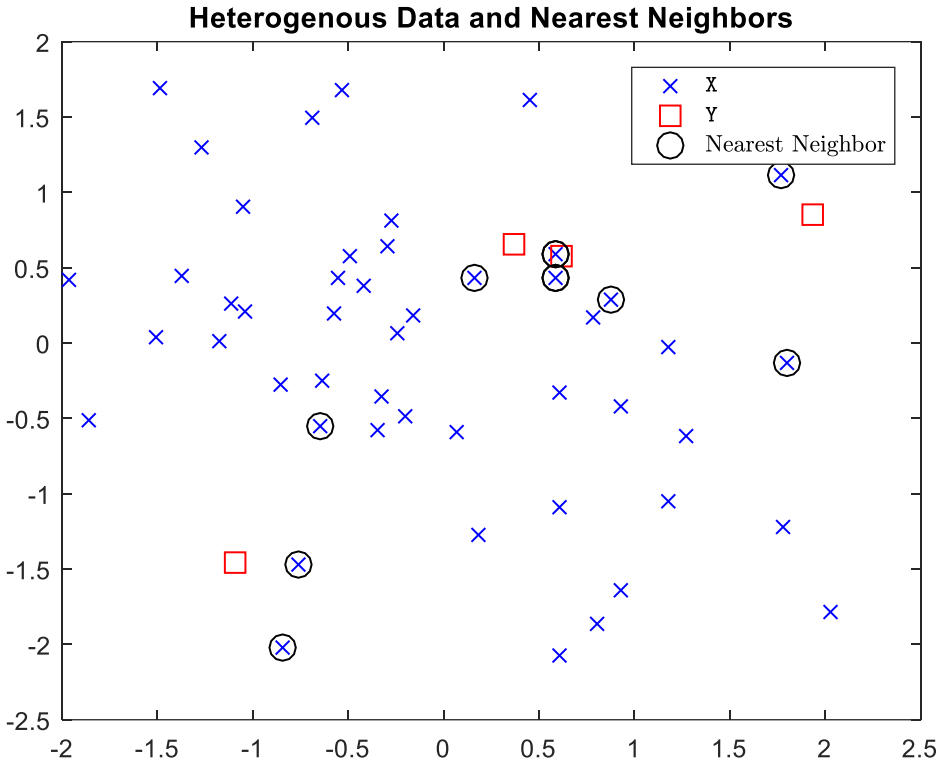

Figure 2.3 : Resulats de classification de l'algorithme KNN

2.10 KNN pour le diagnostic de défauts

La détection de défauts basée sur les méthodes kNN se compose de deux parties : la modélisation, qui utilise des données de formation collectées dans des conditions de fonctionnement normales (NOC), et la détection en ligne. Les principales démarches de cette technique sont présentées par l'organigramme de la figure 2.4.

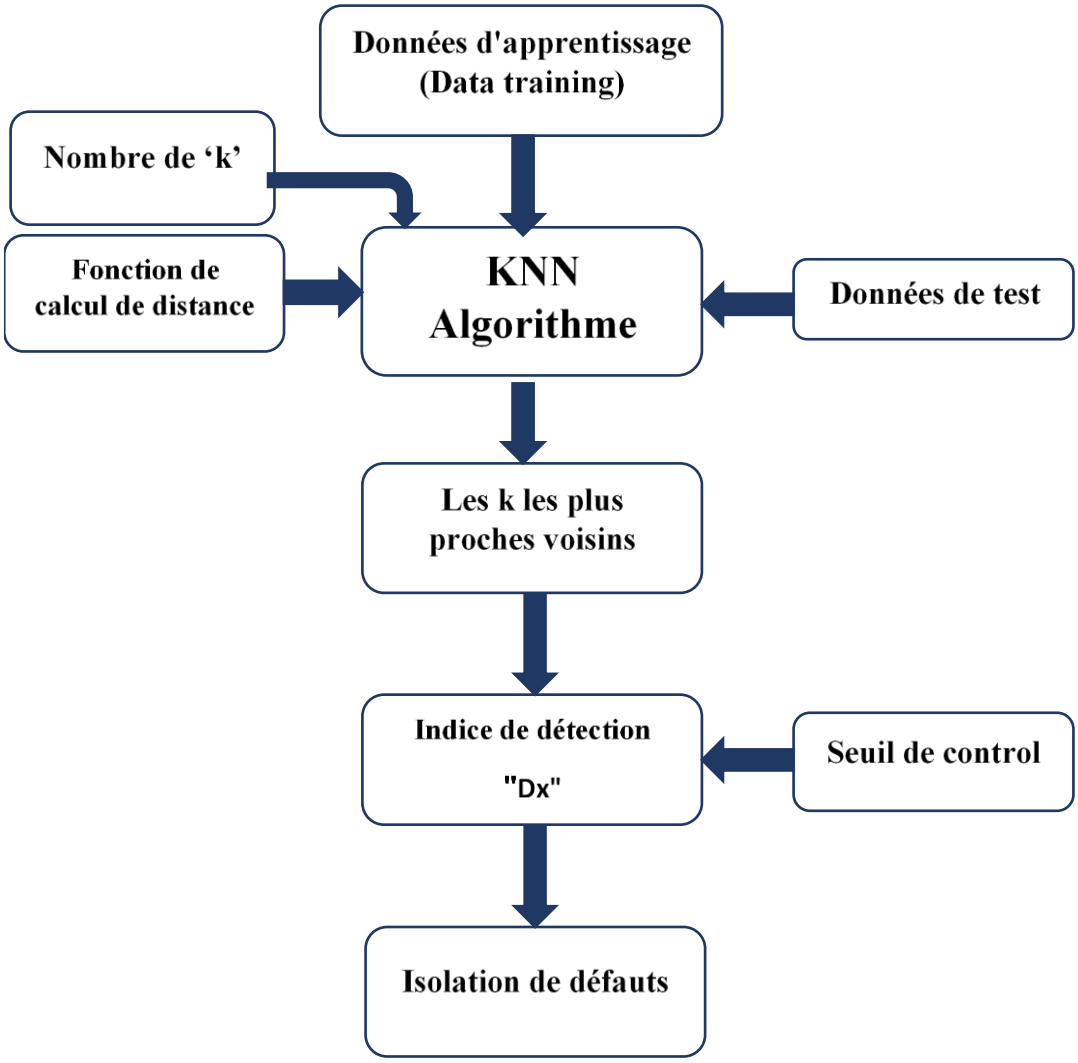

Figure 2.4: KNN pour diagnostic de defaut

-La première étape de la modélisation consiste à calculer la distance quadratique moyenne entre chaque échantillon xi ∈ m et ses (k < N) plus proches voisins dans la matrice de données apprentissage normale (saine) X comme suit :

$$D_{knn}^2(i) = \frac{1}{k}\sum d_{knn}^2(i,j) \dots\dots\dots (1.4)$$

$$d_{knn}^2(i,j) = \| x_i - x_j \|_2$$

-Ici d_{kNN} (i, j) est la distance euclidienne au carré entre le *i ème* échantillon et son *j ème* voisin.

-La deuxième étape est le calcul des limites de contrôle, qui servent à la définition des limites de détection des anomalies. Selon la littérature, il existe de nombreuses façons de calculer la limite de contrôle $D2_{\alpha kNN}$ pour le seuil de signification α. L'une de ces méthodes consiste à utiliser la fonction Chili mit Matlab de la boîte à outils PLS, puisque D^2-$_{kNN(i)}$ peut être approximé par une distribution χ^2 non centrale. D'autres possibilités sont de calibrer les données sous la CNP ou de déterminer les quartiles empiriques (1−α) comme suit :

$$D^2_{\alpha KNN} + D^2_{[N(1-\alpha)]KNN} \cdots\cdots\cdots (1.5)$$

où $D^2_{[i]knn}$, i = 1, . . ., N est le réarrangement de D2 k-NN (i) dans un ordre décroissant et [N(1 − α)] est l'entier de N(1 − α) .Après, the détection de défauts en ligne peut être simplement appliquée en calculant la distance euclidienne au carré D^2_{xkNN} de l'échantillon x $\in R^m$ aux k plus proches voisins comme dans l'Eq. (4), La détection de défauts vise à identifier les anomalies, les écarts ou les déviations par rapport à un comportement normal ou attendu dans un ensemble de données. Cela peut être appliqué, telles que :

$$\begin{cases} D^2_{xKNN} \leq D^2_{\alpha KNN}\ hypothèse\ nulle\ (cas\ sans\ faute) \\ D^2_{xKNN} > D^2_{\alpha KNN} hypothèse\ alternative\ (cas\ de\ faute). \end{cases}$$

-Pour isoler les défauts détectes en utilisant la règle k-NN, le calcul des contributions des variables à la statistique de détection (SPE et T2) comme pour le cas de la méthode statistique ACP peut être utilisée. [9]

2.10.1 Comment un algorithme KNN isole-t-il les defaults ?

L'algorithme k-NN peut être utilisé pour identifier les observations qui sont des valeurs aberrantes ou qui sont très différentes de la majorité des observations de la classe donnée. Cela peut être accompli en ajustant le nombre k de voisins les plus proches utilisés pour l'affectation. Par exemple, si k est petit, l'algorithme sera plus sensible aux valeurs aberrantes car il ne considère qu'un petit nombre d'observations proches. De même, si k est grand, l'algorithme sera moins sensible aux valeurs aberrantes car il considère un plus grand nombre d'observations pour l'affectation.

Il n'y a pas d'équation mathématique spécifique pour isoler les défauts avec k-NN. L'algorithme utilise simplement une mesure de distance pour déterminer la proximité

entre les exemples d'entraînement et les exemples de test, puis utilise les étiquettes des exemples d'entraînement les plus proches pour prédire les étiquettes des exemples de test. Les mesures de distance couramment utilisées incluent la distance euclidienne, la distance de Manhattan et la distance de Minkowski.

2.10.2 Calcul des Contributions des variables à l'indice de détection du k-NN

Le calcul des contributions est une approche de localisation des défauts qui est basée sur la quantification de la contribution de chaque variable à la statistique de détection. D'après le travail de Zhou, inspiré par l'idée des méthodes d'analyse en composantes principales, nous décomposons la distance k-NN (indice de détection FD-k-NN) d'un échantillon x en une somme de m composantes.

$$D_X^2 = \sum_{i=1}^{m} \sum_{j=1}^{k} [\xi_i^T (x - x_j)]^2 \ldots\ldots\ldots (1.7)$$

Ainsi, nous définissons la contribution de la ième variable de x à l'indice de détection D2 x comme :

$$Cont_{FCM-kNN} = \sum_{j=1}^{k} [\xi_i^T (x - x_j)]^2 \qquad i = 1 \ldots\ldots m \qquad (1.8)$$

2.11 Simulation (KNN pour diagnostic de défauts d'une turbine à gaz)

Des données de simulation sont collectées lors du fonctionnement d'une turbine à gaz, sur une heure de temps, située dans la région nord-ouest de la Turquie dans le but d'étudier les émissions de gaz de combustion, à savoir le dioxyde de carbone et les oxydes d'azote (NO + NO2). Ces données sont libres et mises en ligne sur une plateforme internet []. L'ensemble de données contient 36 733 échantillons de mesure pour 11 capteurs. Voir le tableau 2.1 pour plus informations sur la base de données collectée. Cette partie de simulation consiste à appliquer la méthode KNN pour la détection et l'isolation des défauts capteurs de la turbine. Premièrement, la base de données est composée en deux partie, des données d'apprentissage de l'algorithme KNN et des données de testes. Environ 20000 premiers échantillons sont choisis pour l'apprentissage et le reste de données sont utilisées pour le test.

Les mesures du capteur	Unités de mesure
Température ambiante	C
Pression ambiante	mbar
Humidité ambiante	%
Différence de pression du filtre à air	mbar
Pression d'échappement de la turbine à gaz	mbar
Température d'entrée turbine	C
Turbine après température	C
Pression de refoulement du compresseur	mbar
Rendement énergétique de la turbine	MWH
Monoxyde de carbone	mg/m3
oxydes d'azote	mg/m3

Tableau 2.1 Table de mesure

En absence de défauts, l'application de l'algorithme KNN sur les données de test en les comparants échantillon par échantillon avec les données d'apprentissage en utilisant la distance euclidienne, permet l'extraction des k données plus proches voisins pour chaque échantillon de test. Ces k données voisines sont ensuite utilisées pour le calcul de l'indice de détection. La figure 2.4 montre l'évolution de l'indice de détection en absence de défauts.

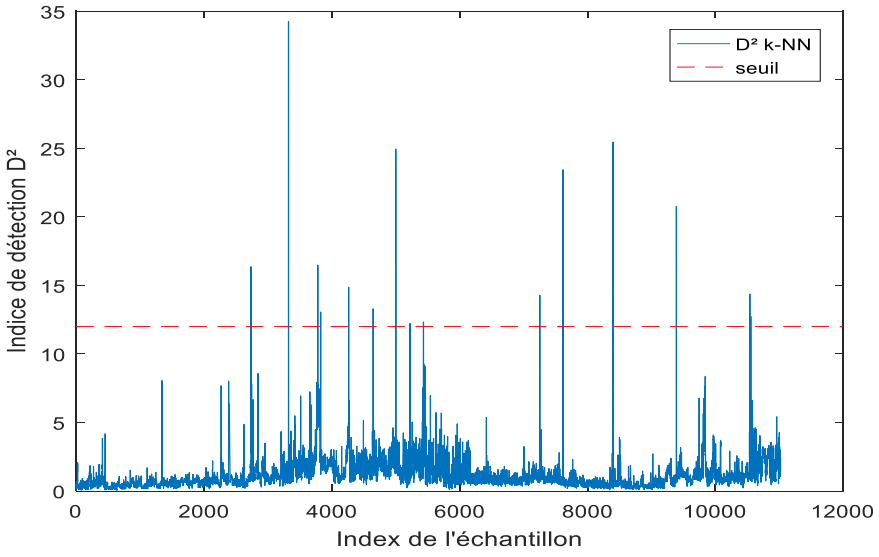

Figure 2.5 : kNN sans defaults

Pour illustrer le cas d'un défaut, un biais affectant la mesure x7 a été simulé à partir de l'instants 2000 avec une amplitude qui s'élève à environ 50% de la plage de variation de cette mesure.

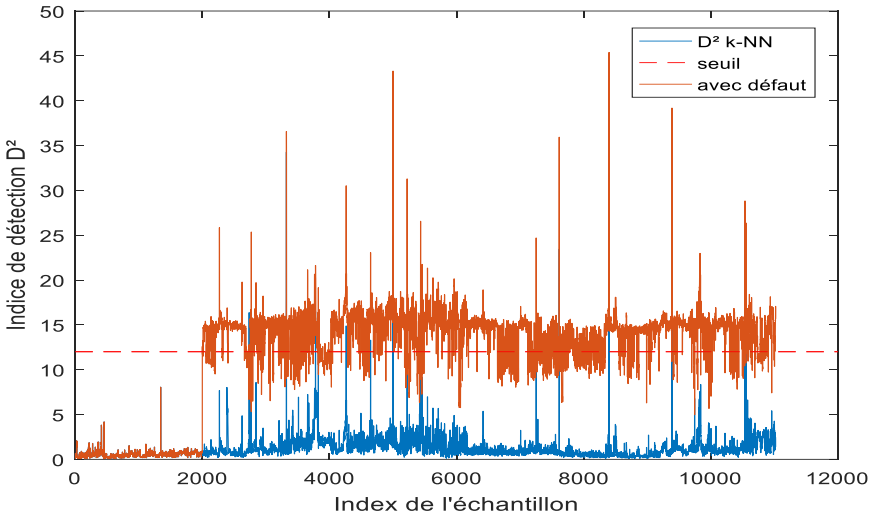

Figure 2.6 : La détection de défauts à base de kNN

La figure 2.5 présente les résultats de détection. Il est clair que le défaut simulé est détecté et dépasse le seuil de control défini. Une étape de localisation est donc nécessaire. En fait, la méthode de localisation de défauts par calcul des contributions, dont la variable qui contribue le plus est la variable en défauts. La 2.6 montre le calcul des contributions à l'indice de détection pour chaque variable, où l'on remarque que la variable qui contribue le plus est la variable x7 alors que c'est la x7 qui est en défaut.

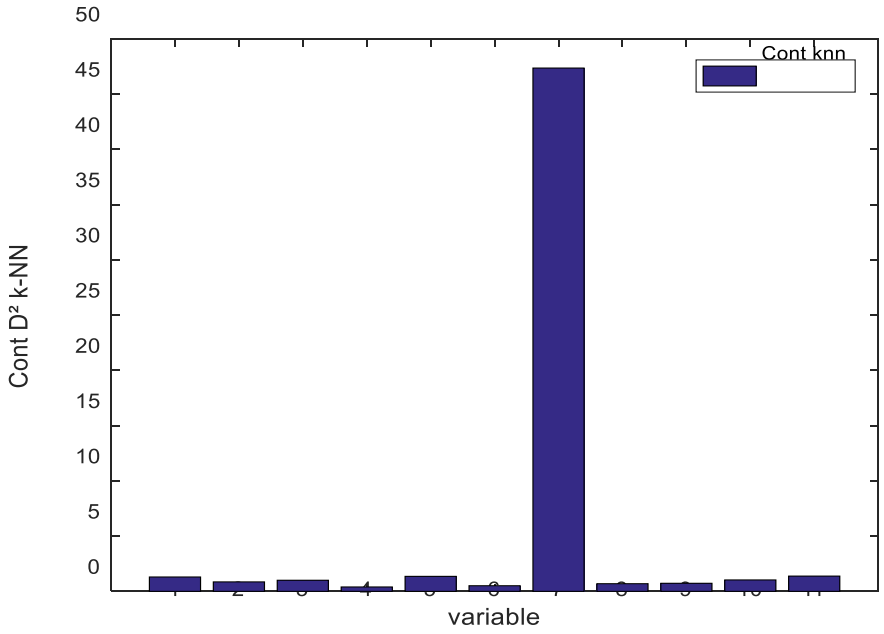

Figure 2.7 : Contrebutions D²kNN

2.12 Conclusion

En conclusion, l'algorithme k-NN est un algorithme d'apprentissage automatique simple mais puissant pour les tâches de classification, de régression et de diagnostic de défauts. Il est basé sur le principe de trouver les voisins les plus proches d'un point de données donné et d'utiliser leur classe ou leur valeur pour faire des prédictions. K-NN présente plusieurs avantages, tels qu'être facile à comprendre et à mettre en œuvre, ne pas avoir d'hypothèses sur la distribution des données. Cependant, il présente également certaines limites, notamment le fait d'être coûteux en calcul pour les grands ensembles de données, sensible aux caractéristiques non pertinentes et nécessitant une sélection minutieuse de la métrique de distance et de la valeur de k. Dans l'ensemble, KNN est un algorithme utile et largement utilisé qui peut être un bon choix pour de nombreuses tâches d'apprentissage automatique.

3 Chapitre 03 : Méthodes de réduction de données et diagnostic de défauts

3.1 Introduction

L'analyse en composantes principales (ACP) est une technique descriptive permettant d'étudier les relations qui existent entre les variables, sans tenir compte a priori d'une quelconque structure. L'utilisation de l'ACP comme un outil de modélisation des processus permet d'estimer les variables ou les paramètres du processus à surveiller. D'un point de vue géométrique, l'ACP est une technique de projection orthogonale linéaire qui projette les observations multidimensionnelles représentées dans un espace de dimension m dans un sous-espace de dimension inférieure (e<m) en maximisant la variance des projections. Elle peut être considérée comme une technique de minimisation de l'erreur quadratique d'estimation ou une technique de maximisation de la variance des projections.

Dans ce travail, l'ACP est utilisée comme un outil de modélisation des relations linéaires entre les différentes grandeurs représentant le comportement d'un processus quelconque. L'identification du modèle ACP est effectué par le calcul des valeurs et vecteurs propres de la matrice de corrélation.

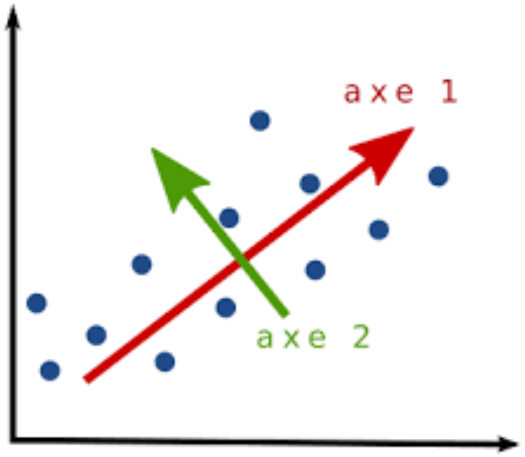

Figure 3.1 : l'analyse en composantes principales

3.1.1 Principes de l'analyse en composantes principales

L'analyse en composantes principales (ACP) est une technique qui permet de réduire la dimensionnalité d'un ensemble de données. Elle consiste à trouver une projection linéaire des données sur un sous-espace de dimension inférieure, tout en minimisant l'erreur de reconstruction des données projetées. Cette projection linéaire est obtenue en déterminant une matrice de transformation orthogonale P qui permet de projeter les données initiales sur le sous-espace de dimension inférieure.

Plus précisément, pour chaque vecteur de données **x**, l'ACP associe un vecteur caractéristique ***t*** qui optimise la représentation de **x** dans le sous-espace de dimension

inférieure. Les vecteurs ***t*** et **x** sont liés par une transformation linéaire $\mathbf{t} = \mathbf{P}^{\mathbf{T}}\,\mathbf{x}$, où la matrice de transformation P est orthogonale. Les colonnes de P sont les vecteurs de la base orthonormés du sous-espace de dimension inférieure.

Les composantes du vecteur caractéristique ***t*** représentent les composantes projetées du vecteur de données **x** dans le sous-espace de dimension inférieure. L'objectif de l'ACP est de trouver la matrice de transformation P qui minimise l'erreur quadratique d'estimation des vecteurs de données x. Cela revient à trouver la projection optimale des données sur le sous-espace de dimension inférieure.

$$X = \hat{X} + \tilde{X} = \hat{T}\hat{P}^T + \tilde{T}\tilde{P}^T$$

3.1.2 Identification du module ACP

En pratique, pour la modélisation d'un processus en utilisant l'analyse en composantes principales, les mesures des variables du processus représentant tous les modes en fonctionnement normal de ce dernier sont collectées dans une matrice X^b. Soit *m* le nombre de variables et *N* le nombre d'observations de chaque variable. $X^b \in \Re$ donnée par:

$$X^b = \begin{pmatrix} x_1(1) & \cdots & x_1(n) \\ \vdots & \ddots & \vdots \\ x_m(1) & \cdots & x_n(1) \end{pmatrix} \ldots\ldots\ldots (3.2)$$

- La première étape avant d'utiliser la matrice de données consiste à normaliser les variables en utilisant la formule suivante :

$$X_i = \frac{X_i^b - M_i}{\delta_i} \ldots\ldots\ldots (3.3)$$

- X_i^b est la i ieme composante du vecteur de mesure X^b, M_i est la moyenne et δ_i est la variance.

$$M_i = \frac{1}{n}\sum_{k=1}^{n} X_i(k) \ldots\ldots\ldots\ldots (3.4)$$

$$\delta_i^2 = \sum_{k=1}^{n} [X_i(k) - M_i]^2 \ldots\ldots\ldots\ldots (3.5)$$

- La nouvelle matrice des données normalisées est notée : $X = [X_1 \ldots X_m]$.

- Après la normalisation de la matrice, nous calculons la matrice de covariance afin que nous puissions découvrir les relations entre les variables. La matrice de covariance est donnée par la formule suivante :

$$\sum = \frac{1}{n-1}(XX^T) \dots\dots\dots\dots (3.6)$$

Après avoir trouvé la matrice de corrélation de cette manière, nous pouvons maintenant procéder à l'identification du modèle dans l'ACP.

- L'étape suivante est la diagonalisation de la matrice de covariance. Dans cette étape, nous décomposons la matrice de covariance en valeurs et vecteurs propres. Les vecteurs propres de la matrice de covariance sont les axes ou directions qui définissent le nouvel espace à rechercher.

$$\sum = P\Lambda P^T = \sum_{i=1}^{m} \lambda_i P_i P^T \dots\dots\dots\dots\dots (3.7)$$

- Les colonnes de la matrice P sont les vecteurs de base orthonormés du sous-espace de représentation réduite des données. La matrice Λ est une matrice diagonale des valeurs propres de la matrice de covariance ou d'autocorrélation.

$$C = PP^T = P^T P = I_m \dots\dots\dots\dots (3.8)$$

P = [P1, P2 . . ., Pm] ∈ $R^{m\times m}$ est la matrice des vecteurs propres et Λ = [Λ1, Λ2 . . . , Λm] ∈ R Matrice de valeurs propres diagonale m par m, où les entrées diagonales sont dans l'ordre décroissant d'amplitude : λ1 > λ2 > ... > λm.

- L'équation suivante exprime la transformation linéaire entre les données X et les composantes principales T. Les composantes principales sont obtenues en projetant les données X sur les vecteurs propres de la matrice de covariance ou d'autocorrélation des données, représentés par les colonnes de la matrice de projection P. Les composantes principales sont obtenues en multipliant la donnée X par la matrice transposée P.

$$T = P^T X \dots\dots\dots\dots (3.9)$$

- Pour obtenir l'estimation d'un vecteur de données X à partir de son vecteur de composantes principales associé T, on a juste à formuler l'écriture suivante

$$X = PT = \sum_{i=1}^{m} t_i \, P_i \, \ldots\ldots\ldots\ldots (3.10)$$

Où

$$X = \hat{X} + \hat{X}(k) \, \ldots\ldots\ldots\ldots (3.11)$$

$$X = \hat{P}\hat{T} + \tilde{P}T \, \ldots\ldots\ldots\ldots (3.12)$$

L'ACP permet de décomposer les données en un sous-espace principal qui capture l'essentiel de la variance des données, et un sous-espace résiduel qui contient les informations résiduelles ou non expliquées par le sous-espace principal. [13]

3.1.3 Pourcentage cumulé de la variance totale (PCV)

L'identification du nombre de composantes principales à retenir dans le modèle ACP est calculé par la méthode PCV. Tandis que chaque composante principale est représentative d'une portion de la variance des mesures du processus étudié. Les valeurs propres de la matrice de corrélation sont les mesures de cette variance et peuvent donc être utilisées dans la sélection du nombre de composantes principales. Le nombre *l* de composantes est alors le plus petit nombre pris de telle sorte que ce pourcentage soit atteint ou dépassé 80 % de la variation globale des données. Le pourcentage de variance expliquée par les premières composantes est donné par [20] :

$$PCV(l) = 100[\sum_{i=1}^{l} \frac{\lambda_i}{\sum_{i=1}^{m} \lambda_i}] \ldots\ldots\ldots (3.13)$$

:

3.1.4 Avantages

- Réduction de la dimensionnalité : l'ACP permet de réduire le nombre de variables tout en conservant les informations les plus importantes des données.
- Détection de relations cachées : l'ACP peut détecter des relations entre les variables qui n'auraient pas été découvertes autrement, en identifiant des combinaisons linéaires de variables qui expliquent une grande partie de la variation dans les données.
- Simplification de l'analyse : l'ACP permet de simplifier l'analyse en fournissant une représentation graphique des données qui peut être facilement interprétée.

3.1.5 Désavantages

- Perte d'informations : l'ACP peut entraîner une perte d'informations importantes, car elle ne prend en compte que la variation qui peut être expliquée par les combinaisons linéaires des variables incluses dans l'analyse.
- Interprétation subjective : l'interprétation des résultats de l'ACP est souvent subjective, car elle dépend des choix faits par l'analyste dans le processus d'analyse.

3.1.6 Exemple de simulation méthode (ACP)

Reprenant l'exemple de simulation étudié dans le chapitre précédent, qui porte sur les données collectées lors du fonctionnement d'une turbine à gaz, l'objectif de cette étude est de démontrer l'efficacité de la méthode de l'Analyse en Composantes Principales (ACP) pour modéliser les données, ainsi que pour représenter les données de manière réduite. La figure 3.2 montre l'évolution des données.

Variables	**Mesures**
X1	Température
X2	Pression
X3	Humidité
X4	Différence de pression du filtre à air

Tableau des variables cibles

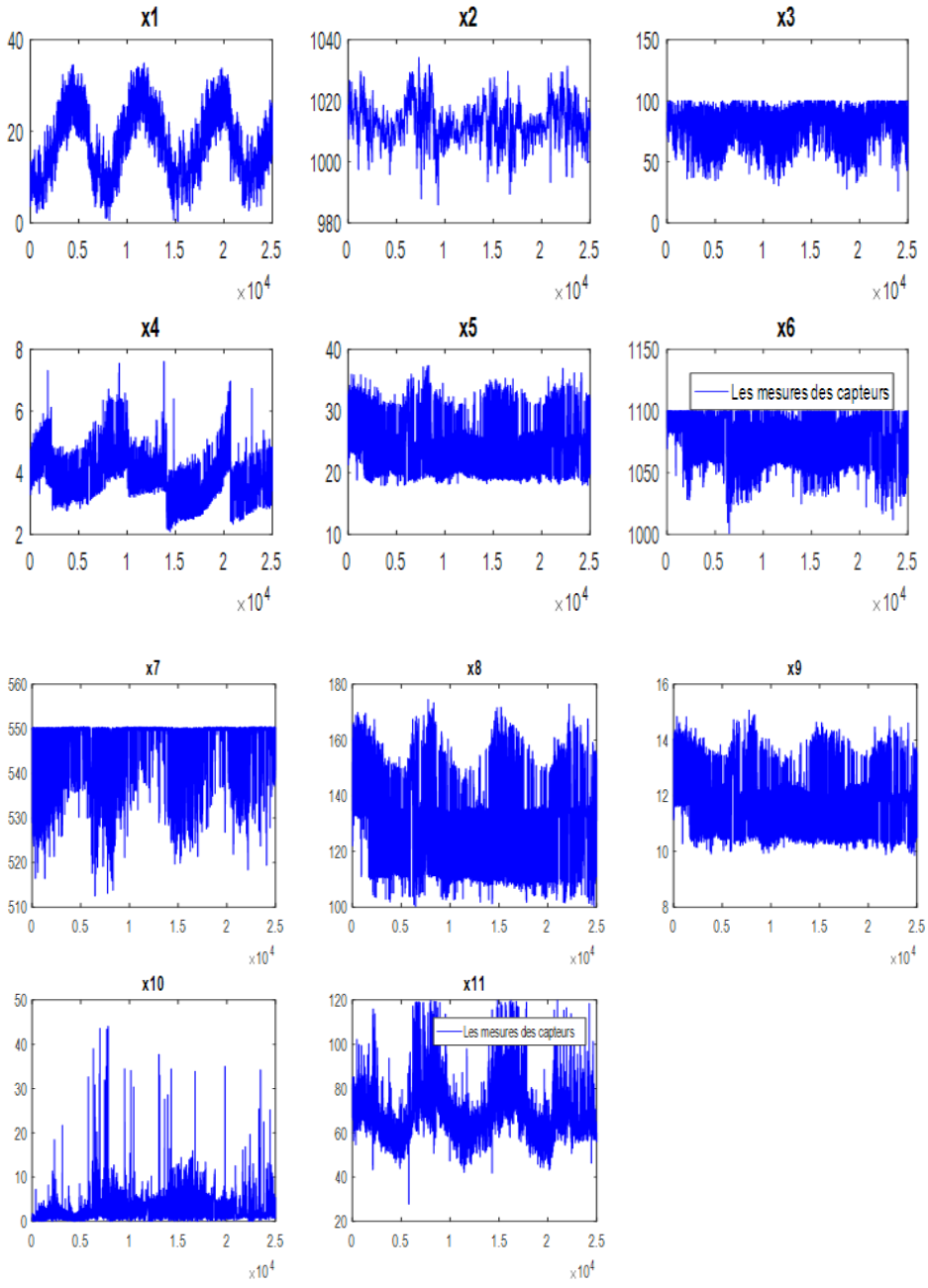

Figure 3.2 : mesures de capteur

En appliquant la méthode ACP sur ces données, i.e., la projection des données dans le nouvel espace orthonormé est représentée par la figure 3.3.

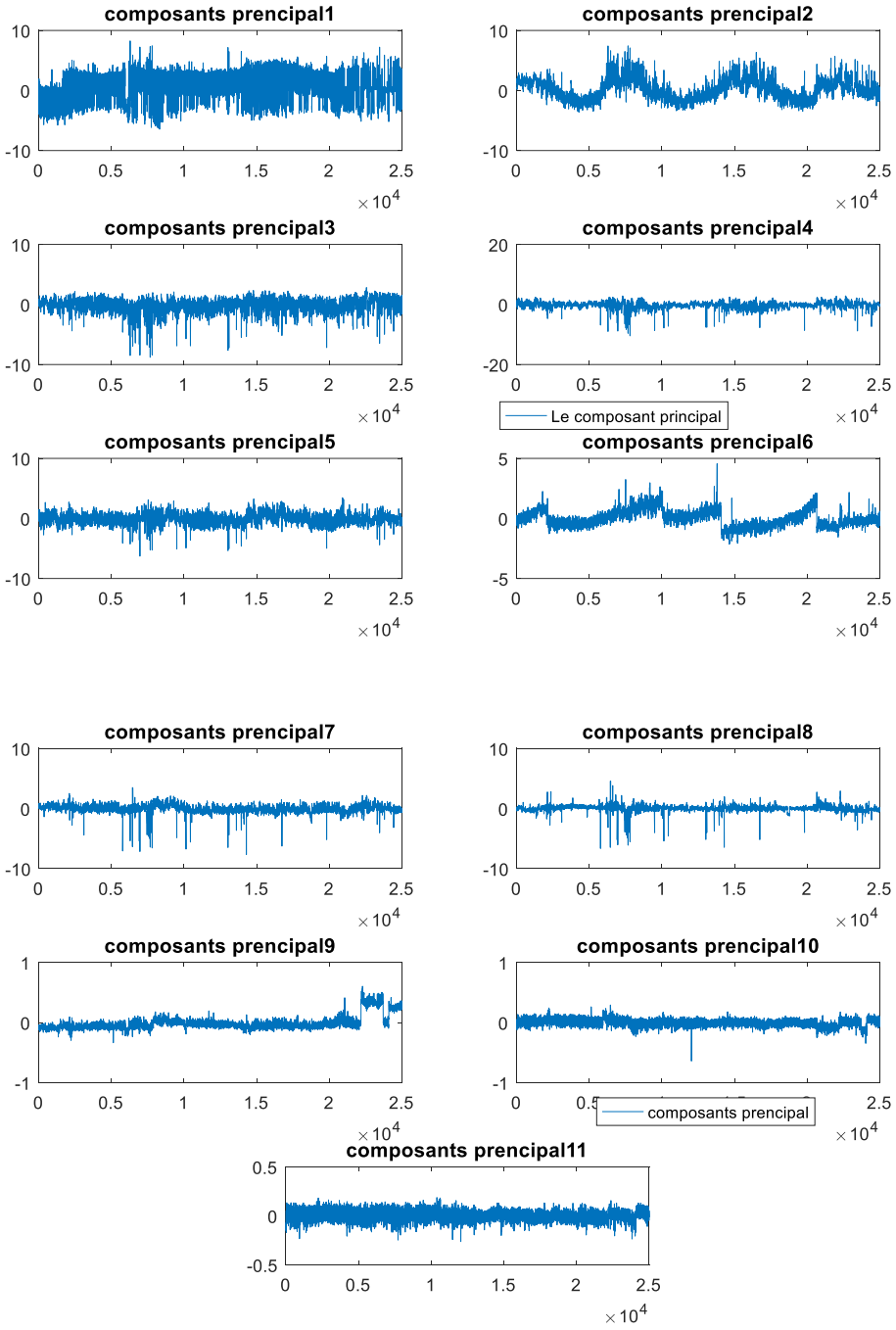

Figure 3.3 : Évolution des composantes principales

Les données et leurs estimations en utilisant le modèle ACP sont présentées sur la figure 3.4. Il est à noter que le nombre de composantes principales retenu par le modèle ACP et calculé par la méthode PCV est de 6 composantes.

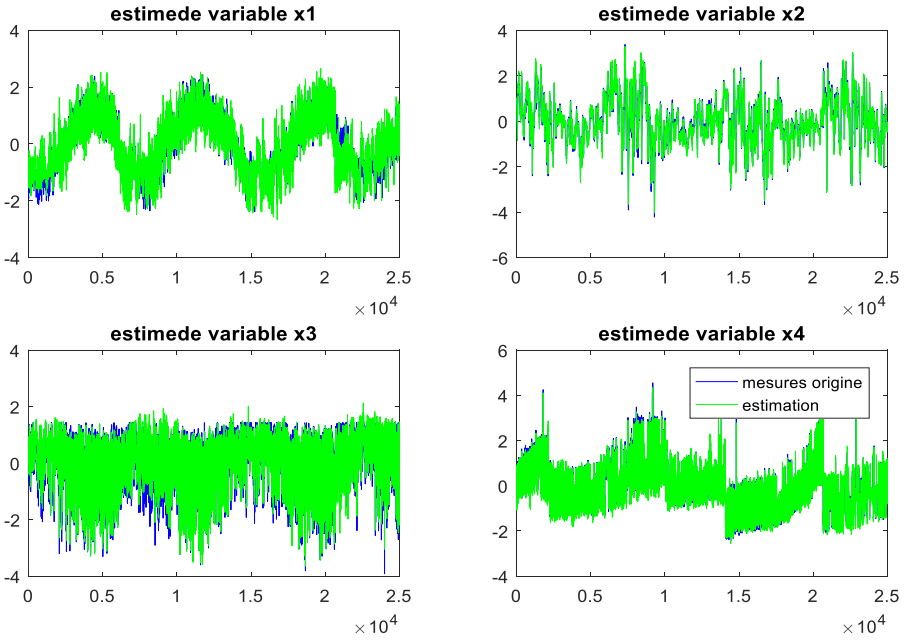

Figure 3.4 : Evolution des mesures et leurs estimations

La figure 3.4 illustre l'évolution des données ainsi que leurs estimations, démontrant clairement que le modèle ACP permet une estimation précise des données d'origine. Cette observation nous a conduits à considérer la méthode ACP comme une solution efficace pour la réduction de dimension, c'est-à-dire la transformation des données initiales en un ensemble de données réduit qui conserve la variabilité des données initiales.

3.2 La méthode des K-centres

K-centres (alias K-Menas Clustering) est un algorithme d'apprentissage non supervisé qui divise les données non gérées en groupes (ou clusters) distincts. Le K indique qu'il s'agit d'une référence au nombre de clusters/groupes (un cluster est un groupe d'observations/enregistrements similaires). Par exemple, dans l'exemple de la figure 3.5, la population non étiquetée a été regroupée en trois groupes en fonction de la valeur de K. [10]

3.2.1 Comment ça fonctionne ?

Pour traiter les données d'apprentissage, l'algorithme K- centres dans l'exploration de données commence par un premier groupe de centrioles sélectionnés au hasard, qui sont utilisés comme points de départ pour chaque cluster, puis effectue des calculs itératifs (répétitifs) pour optimiser les positions des centres. Il interrompt la création et

l'optimisation des clusters lorsque : Les centrioles se sont stabilisés - il n'y a pas de changement dans leurs valeurs car le regroupement a réussi, ou le nombre d'itérations défini a été atteint. Le fonctionnement de l'algorithme k-means est le suivant :

3.2.2 Étapes de l'algorithme de k-center

- Phase 1 : placez le centre du premier groupe dans l'espace 2d au hasard.
- Phase 2 : Attribut à la catégorie avec le centroïde le plus proche de chaque objet.
- Phase 3 : Mesurez les positions des centroïdes.
- Phase 4 : Si les positions des centroïdes n'ont pas changé, l'étape suivante est franchie.
- Phase 5 : Finition.[21]

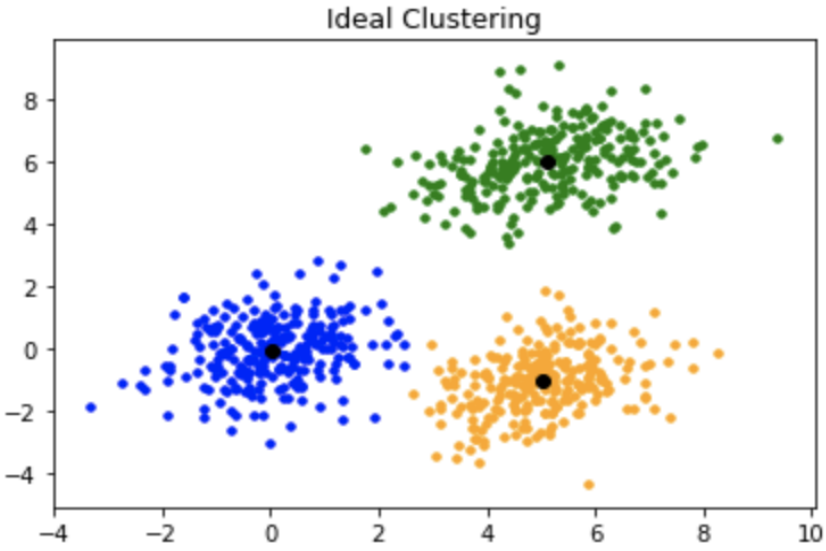

Figure 3.5 : Représentation graphique de K-centres

Exemple de simulation (k-moyen)

L'exemple suivant représente le clustering avec K-means sur l'ensemble de données "iris" de MTLAB. Le première sous-figure montre les données colorées en fonction de leurs affectations aux clusters, avec les centroïdes de chaque cluster indiqué par des croix noires. La deuxième sous-figure montre uniquement les données brutes sans

les marqueurs de couleur ou de groupe. L'objectif est de visualiser les résultats de clustering K-means avant et après clustering.

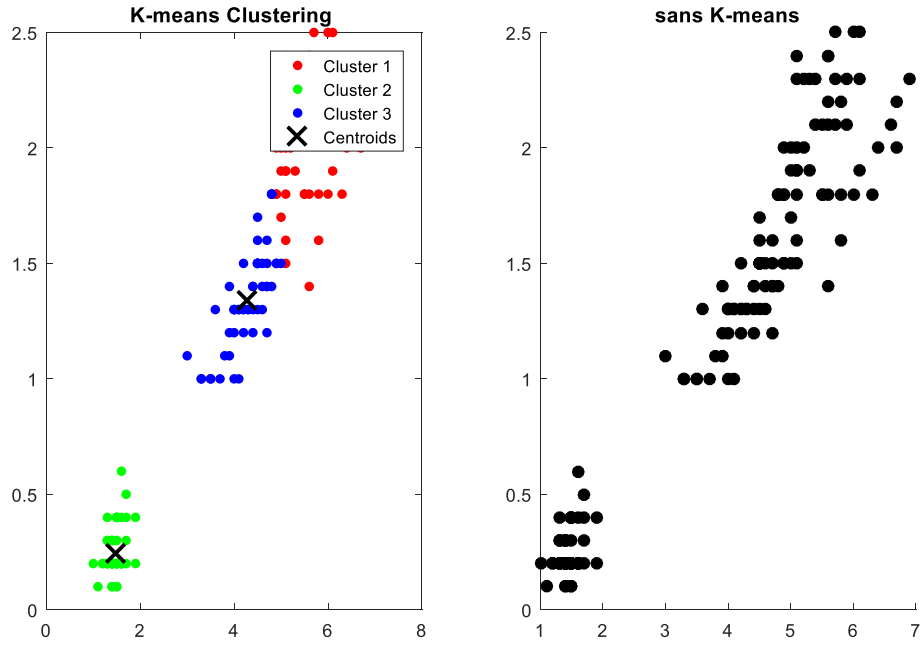

Figure 3.6 Exmpel de clustring par k-means

3.3 La méthode combinée PC-KNN pour prédiction de défauts

L'application de l'approche conventionnelle KNN pour la prédiction de défauts présente plusieurs faiblesses. C'est pourquoi l'algorithme kNN n'est pas largement utilisé dans sa forme originale, mais plutôt avec des versions améliorées limitant partiellement ces lacunes [18,19].

La combinaison de l'approche ACP avec l'algorithme KNN a pour objectif principal de réduire les problèmes liés à la complexité de calcul et à la consommation de l'espace mémoire. L'algorithme de la méthode combiné PC-KNN est illustré dans la figure suivante.

3.3.1 L'organigramme PC-KNN

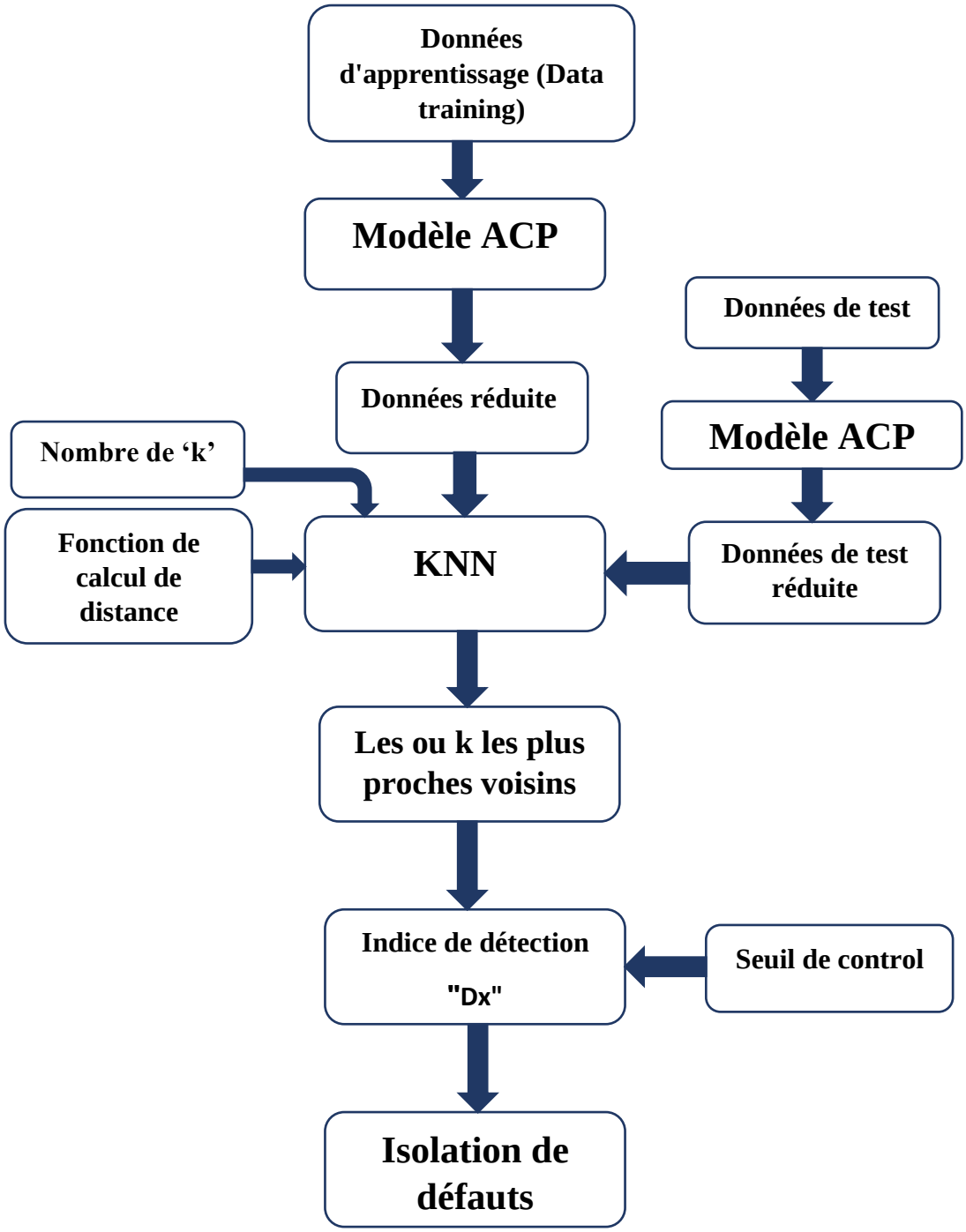

Figure 3.7:PC-KNN pour diagnostic de defaut

La méthode ACP consiste à réduire la dimensionnalité des données d'origine en les représentant avec leurs composantes principales. Les composantes principales des données d'apprentissage (données réduites) sont considérées comme des données d'entrées pour l'algorithme k-NN. Pour vérifier la crédibilité du nouvel échantillon de

données, l'algorithme cherche les k échantillons les plus proches de celui-ci dans l'ensemble des composantes principales d'apprentissage.

L'indice de détection d'erreur basé sur PC-kNN peut être dérivé comme suit. Tout d'abord, calculez les k plus proches voisins pour chaque échantillon dans la matrice de données réduite T. [9]

$$D^2{}_{PC-KNN}(i) = \frac{1}{k}\sum_{j=1}^{k} d^2{}_{PC-KNN}\,(i,j) \ldots\ldots\ldots\ldots (3.14)$$

Avec,

$$d^2{}_{PC-kNN}(\mathrm{i,\ j}) = \| t_i - t_j \|\, 2 \quad , \mathrm{j}=1\ldots, \mathrm{N};\ \mathrm{j}\neq\mathrm{i}\ldots\ldots\ldots\ldots (3.15)$$

Lors de la détection de défauts, un échantillon est projeté sur les premières vecteurs propres définissant le sous-espace principal afin d'obtenir un vecteur de données réduit. Ensuite, les k plus proches voisins de ce vecteur sont calculés et comparés à un seuil pour la détection de défauts.

Le calcul des valeurs des contributions de chaque variable peuvent également être calculées en utilisant la formulation suivante :

$$Cont_{PC-KNN}(i) = \sum_{j-1}^{k} \| \zeta_i(x - x_j) \|^{\,2}, = 1,2,\ldots,m \qquad \ldots\ldots\ldots\ldots (3.16$$

La mise en œuvre de l'algorithme PC-KNN nécessite une connaissance de l'ACP, de la réduction de dimensionnalité et de k-NN. Il est important de noter que les performances de l'algorithme dépendent de la sélection du nombre de composantes principales *(l)* et de la valeur de k dans l'algorithme k-NN.

3.4 La méthode PC-KNN et la méthode k-moyenne pour la détection de défauts

Après avoir examiné la méthode des k-centres, nous établirons ensuite un lien avec la méthode PC-KNN. Cette approche nous permet d'éliminer les calculs supplémentaires en utilisant les centres extraits de la méthode des k-centres au lieu de traiter l'intégralité des données réduite issues de la méthode ACP, tout en conservant les échantillons les plus significatifs. Cette approche de combinaison nous permet ainsi de simplifier le processus de calcul tout en

préservant les informations essentielles pour la prise de décision. L'organigramme descriptif de cette méthode est présenté dans la figure suivante.

3.4.1 Organigramme PC-KNN-Kmeans

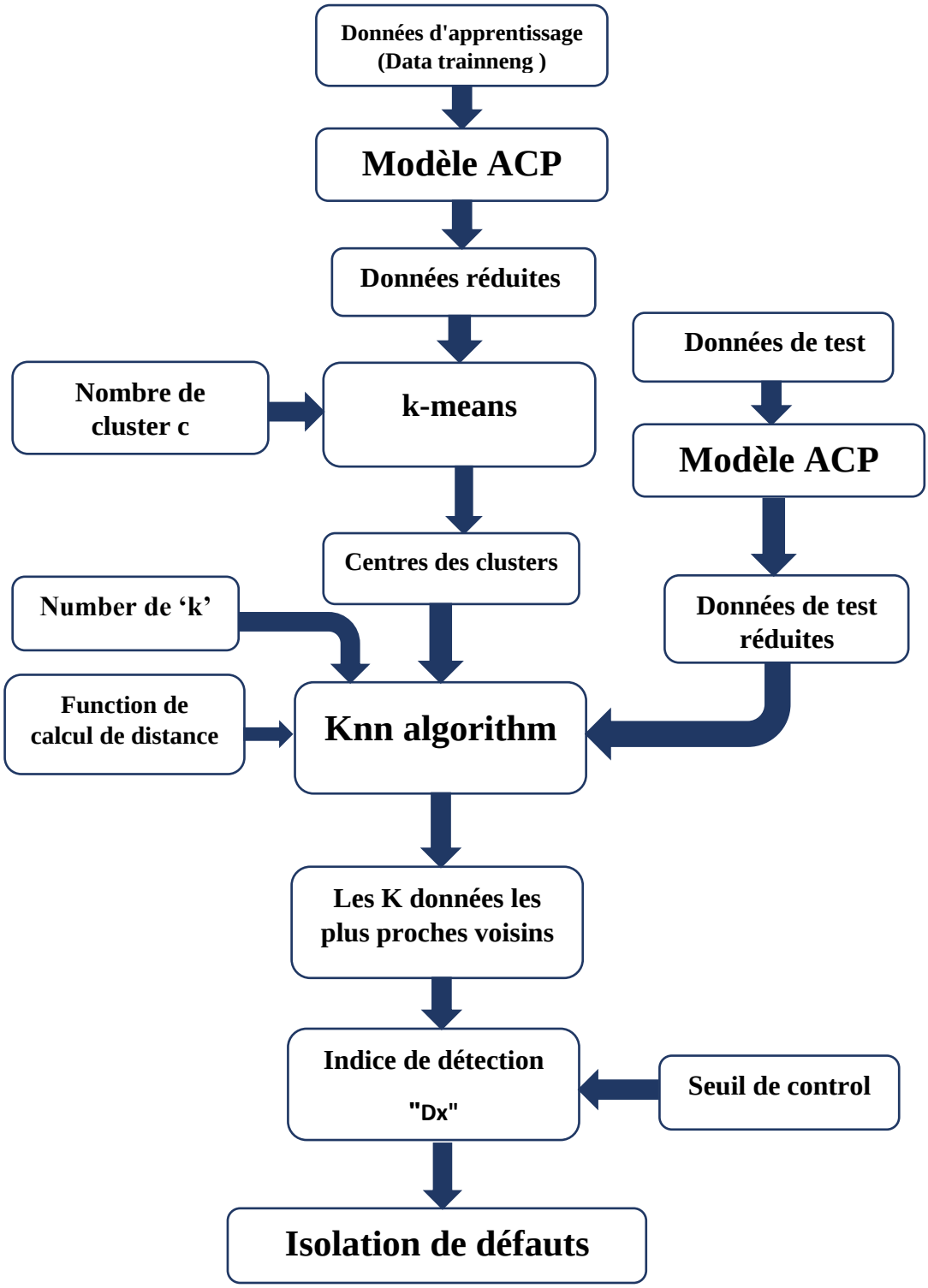

Figure 3.8:PC-KNN et K-means pour diagnostic de defaut

3.5 Fonctionnement de PC-KNN et K-means

Au début, nous utilisons la version réduite des données d'apprentissage obtenue à partir de l'algorithme ACP. Ensuite, nous passons à l'étape suivante où nous appliquons une analyse en cluster sur ces données réduite en utilisant la méthode des k-means en spécifiant un nombre de clusters c. L'algorithme des k-means calcule les centres des clusters, nous permettant d'obtenir de nouvelles données plus compactes qui conservent les éléments les plus importants. Ensuite, nous utilisons ces données (les centres des données réduites) dans l'algorithme kNN avec un nombre spécifié de k plus proches voisins pour obtenir un indice de détection et puis l'appliqué pour la localisation des défauts, tel que représenté dans la figure 3.12. Cette approche combinée permet d'obtenir des résultats plus efficaces en termes de qualité de diagnostic des défauts.

3.6 Exemple de simulation

On reprend le même exemple de simulations que nous avons étudies dans ACP pour simuler. Dans la simulation, nous étudierons 36733 échantillons de 11 variables extraites des simulations Monte Carlo de Turbofan, de sorte que nous utilisons 25712 comme données d'apprentissage pour l'algorithme k-NN et les 11733 restants comme échantillon de test. Nous choisissons un nombre k de voisins k=3 et k=4.

En appliquant l'algorithme ACP, nous avons réduit le nombre de variables à 5 variables représentative appelées composantes principales, et cette étape est dans le but de réduire et capturé toutes les relations entre les variables.

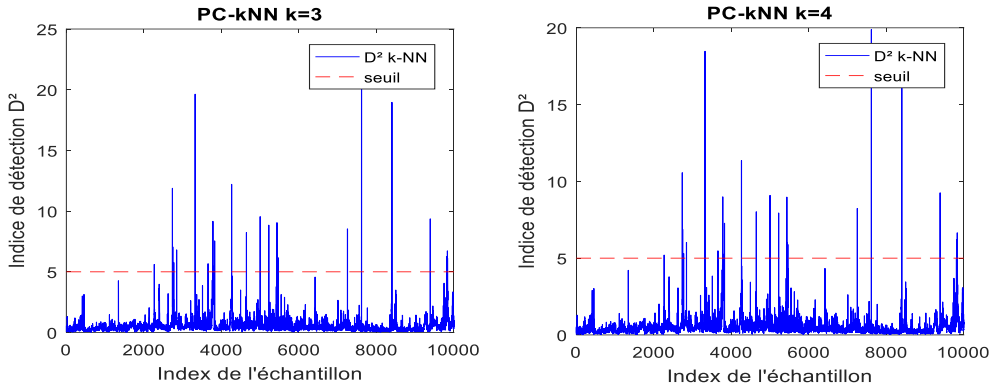

Figure 3.9:index de détection

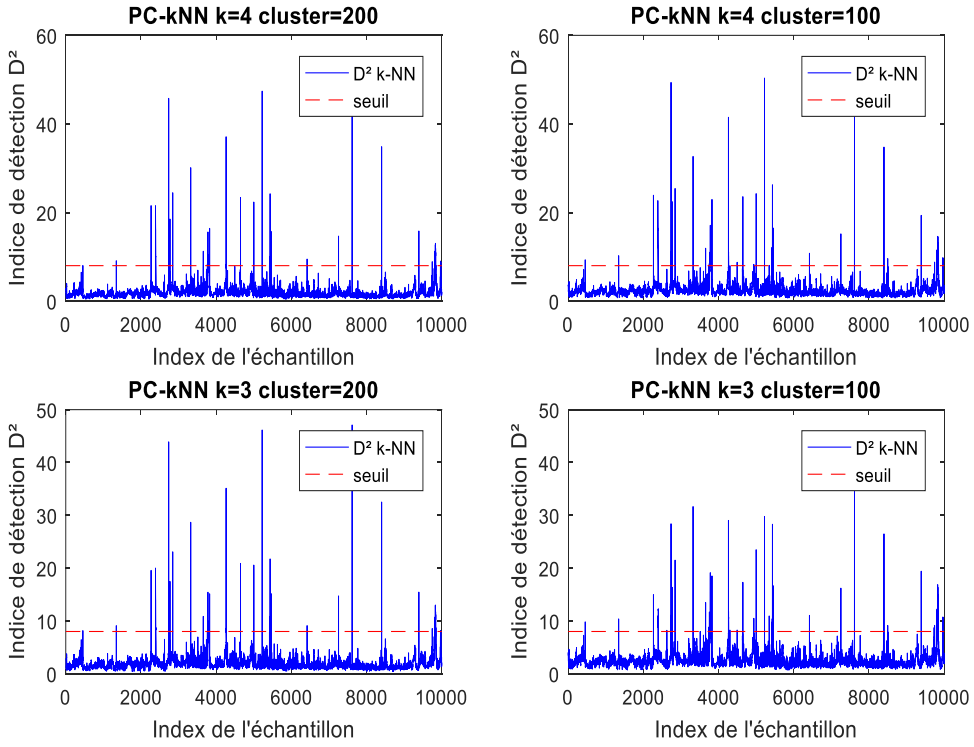

Figure 3.10:La détection de défauts à base de PC-kNN et k-center

Après avoir appliqué la corrélation entre les algorithmes ACP et KNN nommés PC-KNN en l'absence de défauts, l'application de l'algorithme aux données de test en le comparant échantillon par échantillon avec les données d'apprentissage à l'aide de la

distance euclidienne, permet d'extraire les données k-plus proches voisins pour chaque échantillon d'essai. Ces k données adjacentes sont ensuite utilisés pour calculer l'indice de détection. Les figures (3.9) (3.11) montre l'évolution de l'indice de détection en l'absence de défauts.

Phase de défaut

Pour la détection défaut , nous avons défini une erreur par défaut dans X2 (La poussée brute KN) produite avec une amplitude d'environ 40 % de sa plage de variance apparaissant du temps 2000 à 3973 considéré comme le dernier échantillon dans les données. Comme le montre la figure (4.11).

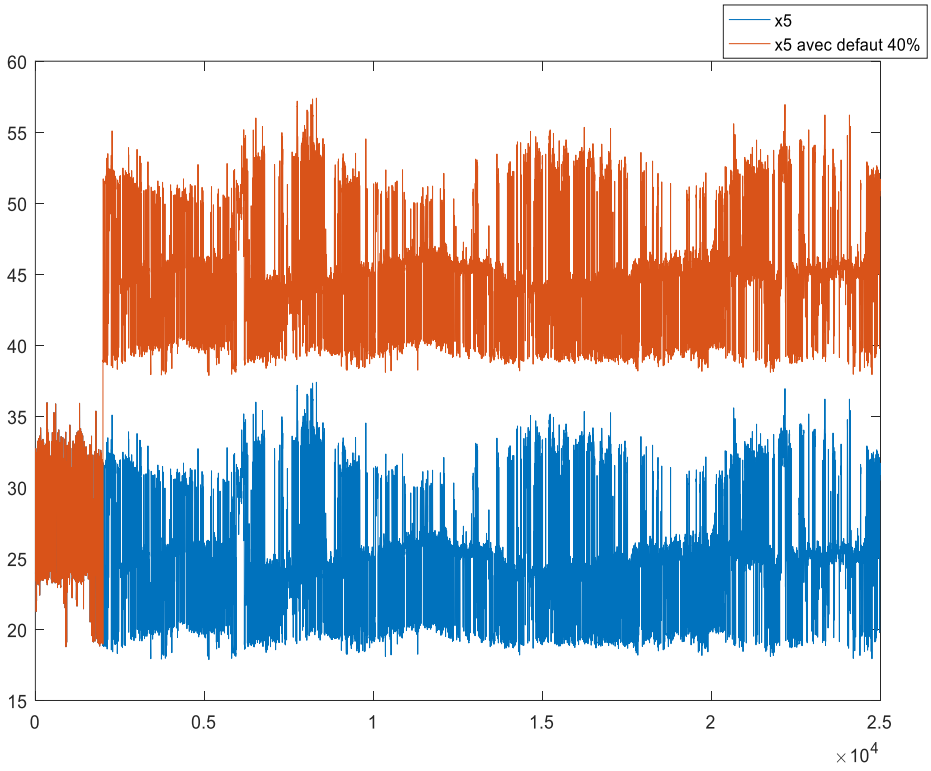

Figure 3.11 : le capteur x2 sans et avec le défaut

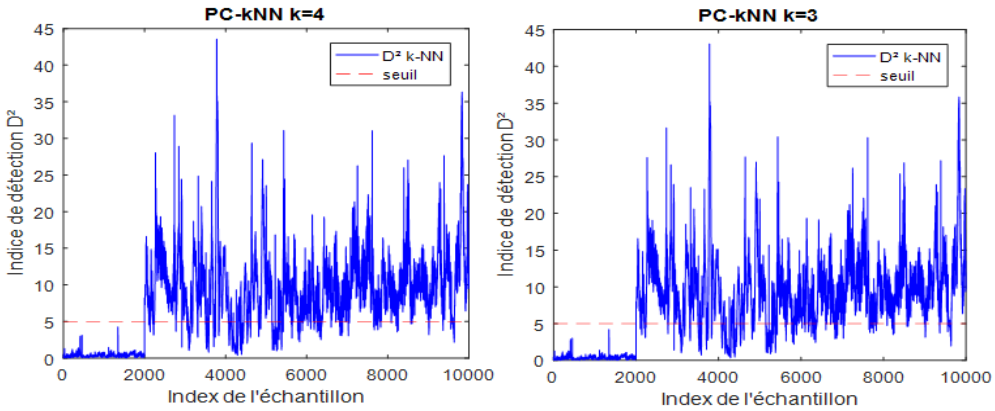

Figure 3.12:La détection de défauts à base de PC-kNN

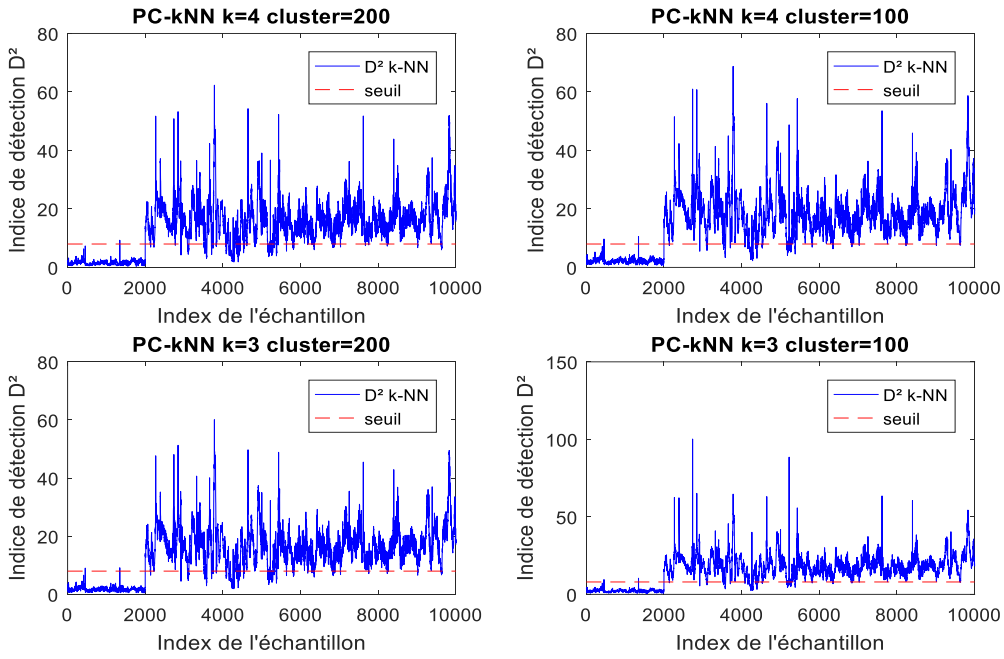

Figure 3.13:La détection de défauts à base de PC-kNN et k-center avec défaut dans x2

-En l'absence de défauts, l'application de l'algorithme PC-KNN-k-means aux données de test en le comparant échantillon par échantillon avec les données d'apprentissage à l'aide de la distance euclidienne permet d'extraire les données du k plus proche voisin pour chaque échantillon de test. Une fois l'erreur survenue, nous remarquons que le curseur est passé au-dessus du seuil, et cela est dû au défaut ajouté dans X2 C'est ce que représente la figure (3.12)(3.13)

Phase de localisation de défaut

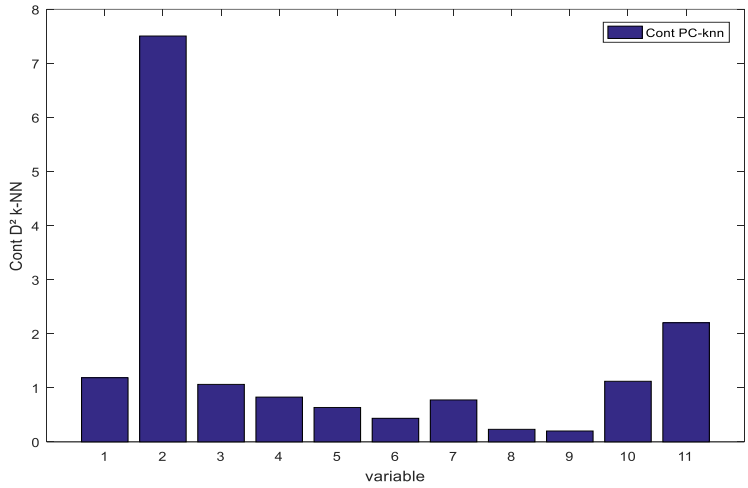

Figure 3.14:localisation de défaut dans la méthode de PC-KNN

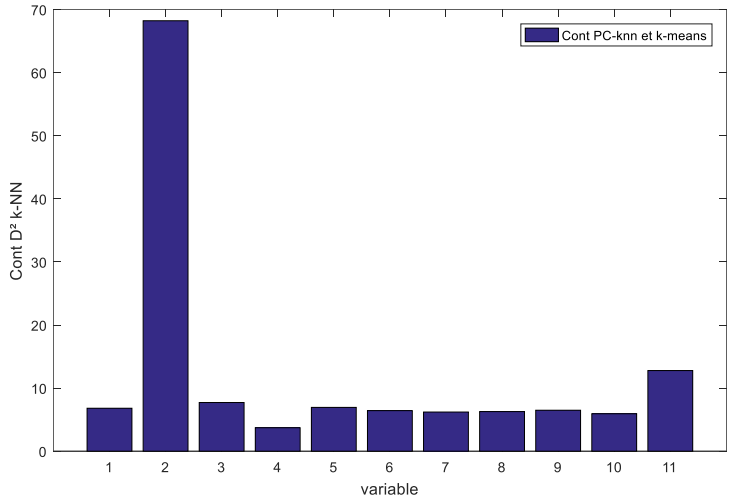

Figure 3.15 localisation de défaut dans la méthode de PC-KNN et k-means

En fait, la méthode de localisation de défauts par calcul des contributions, dont la variable qui contribue le plus est la variable en défauts. La Eq (3.10) montre le calcul des contributions à l'indice de détection pour chaque variable, où l'on remarque que la variable qui contribue le plus est la variable x2 alors que c'est la x2 qui est en défaut.

3.7 Conclusion

En résumé, la combinaison de l'Analyse en Composantes Principales (ACP) et de l'algorithme des K plus proches voisins (KNN) pour la détection de défauts présente plusieurs avantages. L'ACP permet de réduire la dimensionnalité des données et d'identifier les variables les plus pertinentes, tandis que le k-moyen classe les nouvelles observations en se basant sur les exemples les plus similaires. En combinant ces deux méthodes, appelée PC-KNN, on peut améliorer la performance de détection des défauts en prenant en compte à la fois la structure globale des données et la proximité des observations similaires.

4 Chapitre 04 : Cas d’étude

4.1 Introduction

La détection et la localisation d'objets sont des tâches critiques dans de nombreux domaines, tels que la vision par ordinateur, la robotique et la surveillance. La capacité de détecter et de localiser avec précision des objets dans une scène est essentielle pour de nombreuses applications, notamment la détection des piétons pour les véhicules autonomes, la reconnaissance faciale pour la sécurité, la localisation des défauts dans l'inspection industrielle, etc. Encore une fois.

Dans ce chapitre, nous appliquerons ces algorithmes à des données extraites de simulations de Monte Carlo et analyserons les résultats obtenus en appliquant les algorithmes.

4.2 Histoire de simulation Monte Carlo

John von Neumann et Stanislaw Ulam ont inventé la simulation de Monte Carlo, ou méthode de Monte Carlo, dans les années 1940. Ils l'ont nommé d'après le célèbre lieu de jeu à Monaco parce que la méthode partage la même caractéristique aléatoire qu'un jeu de roulette. [17]

4.3 Qu'est-ce qu'une simulation Monte Carlo méthode ?

La méthode de Monte Carlo reconnaît un problème pour toute technique de simulation : la probabilité de résultats variables ne peut pas être clairement identifiée en raison de l'interférence des variables aléatoires. Par conséquent, une simulation de Monte Carlo se concentre sur la répétition constante d'échantillons aléatoires.

Une simulation de Monte Carlo prend la variable qui a une incertitude et lui attribue une valeur aléatoire. Le modèle est ensuite exécuté et un résultat est fourni. Ce processus est répété encore et encore tout en attribuant de nombreuses valeurs différentes à la variable en question. Une fois la simulation terminée, les résultats sont moyennés pour arriver à une estimation. .[18]

variables	Description des variables	nité
X1	La poussée nette produite par le moteur.	KN
X2	La poussée brute produite	KN
X3	La poussée brute produite	KN
X4	Consommation spécifique de carburant	KN/S
X5	La poussée spécifique produite par le moteur	m/s
X6	La vitesse des gaz d'échappement	m/s
X7	Le rapport de pression à travers le noyau	
X8	La vitesse des gaz d'échappement à la dérivation	m / s
X9	Le rapport de pression à travers la dérivation	/
X10	L'efficacité du brûleur à convertir l'énergie du combustible en énergie cinétique des gaz d'échappement	/
X11	Le rapport de pression entre la station 5 et la station 2 dans le moteur	/
X12	La vitesse de rotation du tiroir haute pression en tours par minute	RPM
X13	La vitesse de rotation du tiroir basse pression en tours par minute	RPM
X14	Le débit de carburant à travers le moteur en kilogrammes par seconde	kg/s
X15	The pressure at the exit of the low-pressure turbine at station 5 in kilopascals	kPA
X16	The temperature at the exit of the low-pressure turbine at station 5 in Kelvin	K
X17	Le rendement isentropique du compresseur haute pression	/

Tableau 4.1: La description des variables du processus

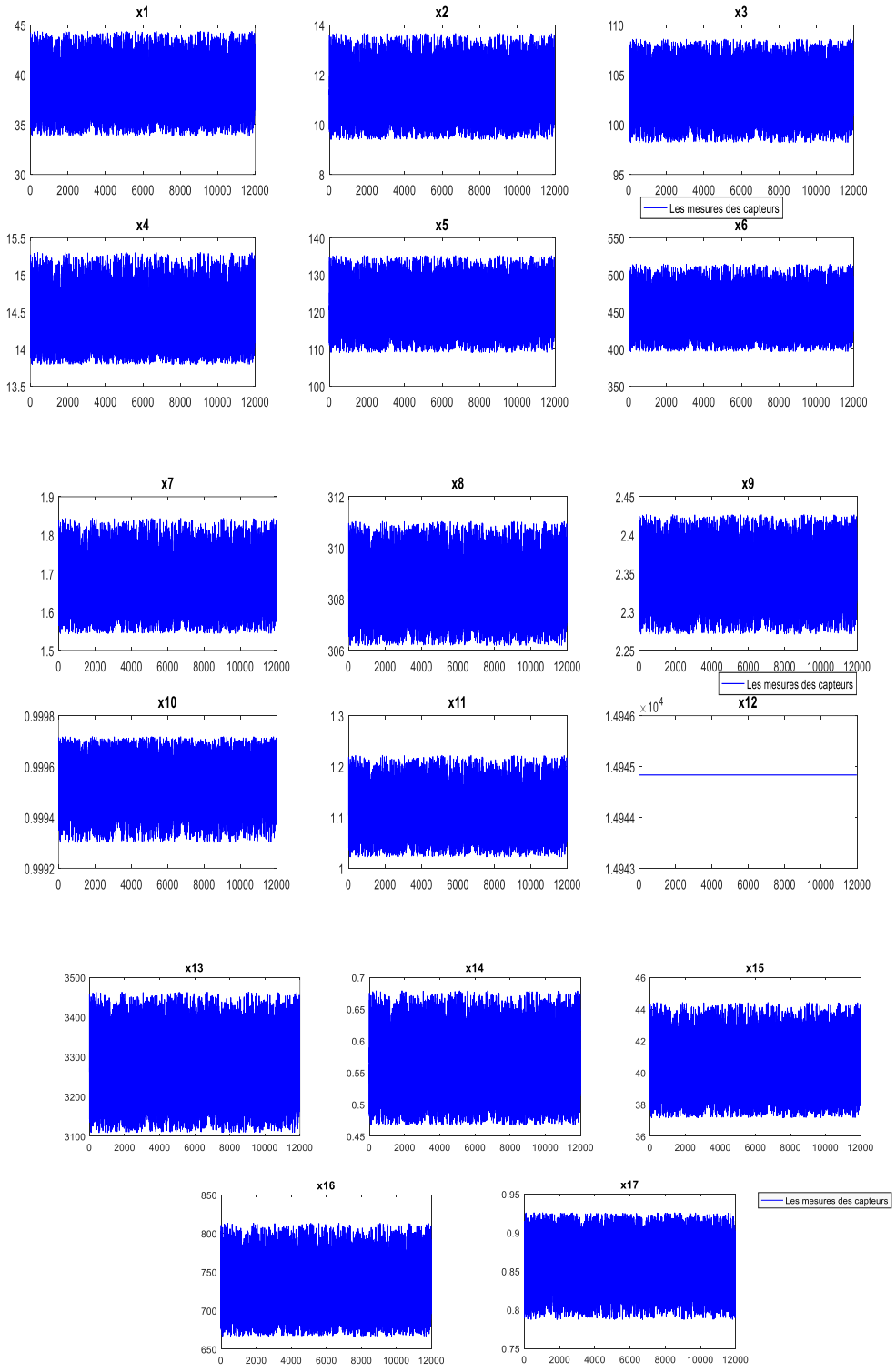

Figure 4.1 : L'évolution des mesures des 17 capteurs

Est-ce cet exemple que nous allons appliquer la méthode que nous obtenons 5 composant principal montré dans la figure correspondante

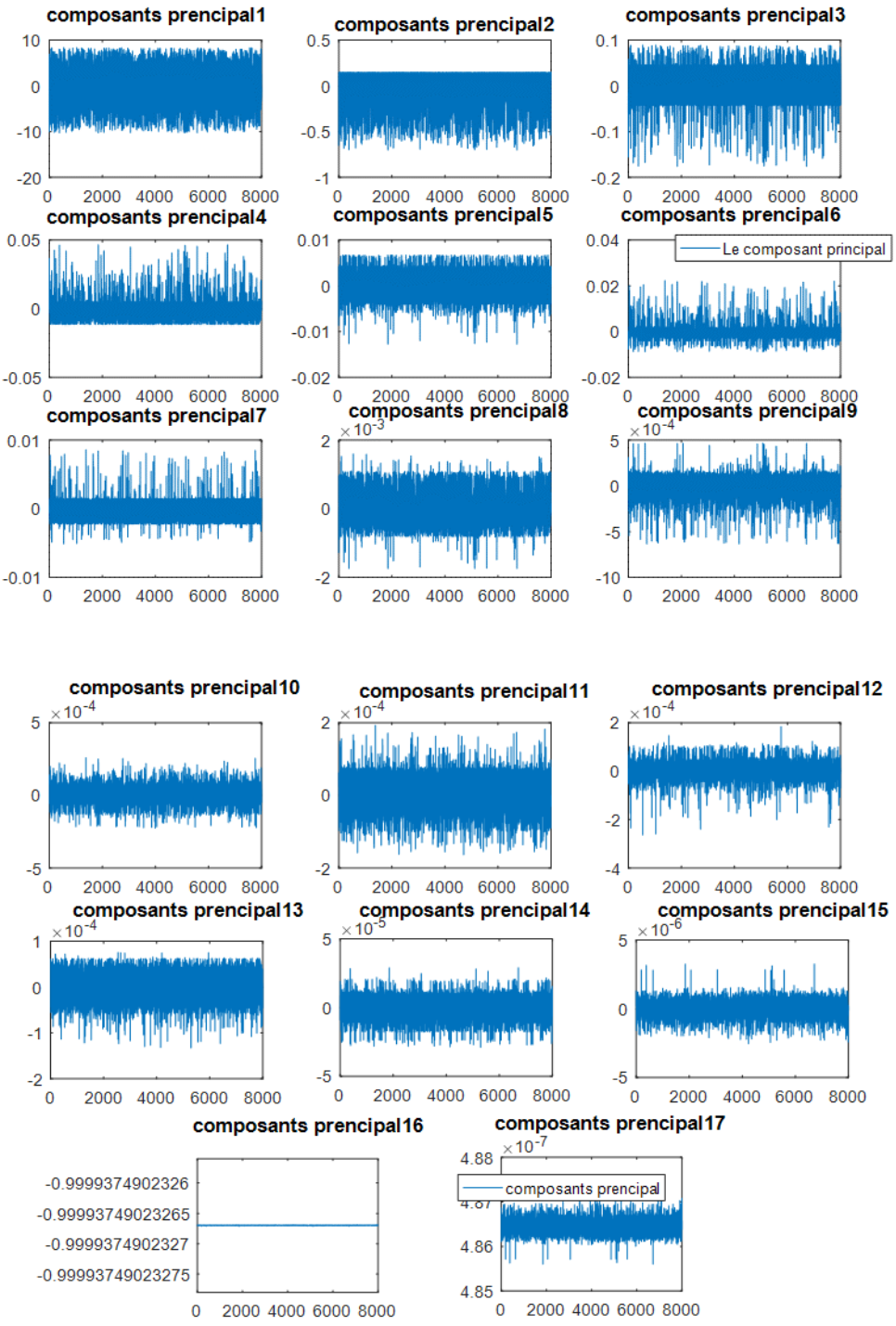

Figure 4.2 : les composants principaux

La figure (4.3) montre l'évolution des estimations d'approximation

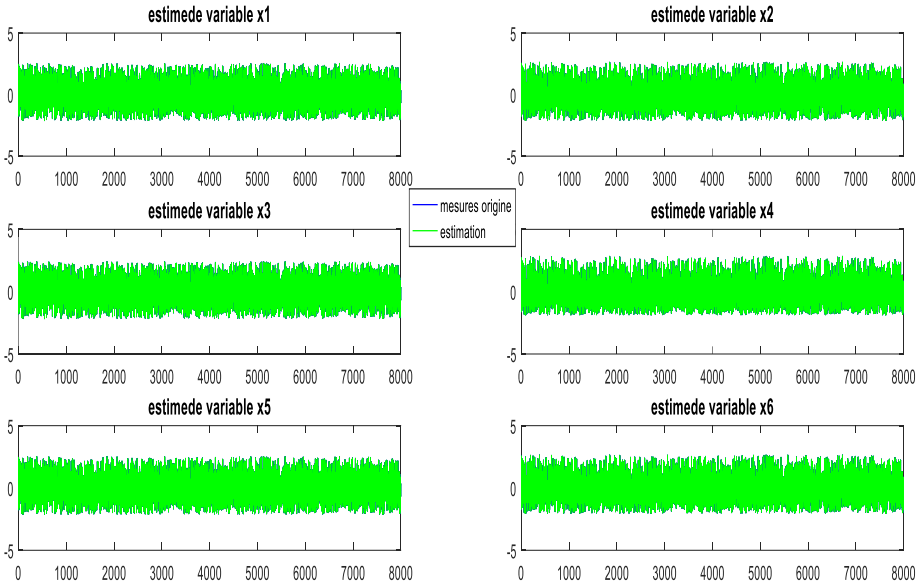

Figure 4.3 : L'estimation des différentes variables

Dans la simulation, nous étudierons 11972 échantillons de 17 variables extraites des simulations Monte Carlo de Turbo fan, de sorte que nous utilisons 7999 comme données d'apprentissage pour l'algorithme k-NN et les 3973 restants comme échantillon de test.

Nous choisissons le nombre de voisins k=3 et k=4.

Et lorsque nous avons utilisé l'algorithme ACP, nous avons réduit le nombre de variables de 17 à 5 variables, et cette étape est dans le but de rompre toutes les relations entre les variables, et pour cela nous avons obtenu 5 composant principal. Figure 4.2

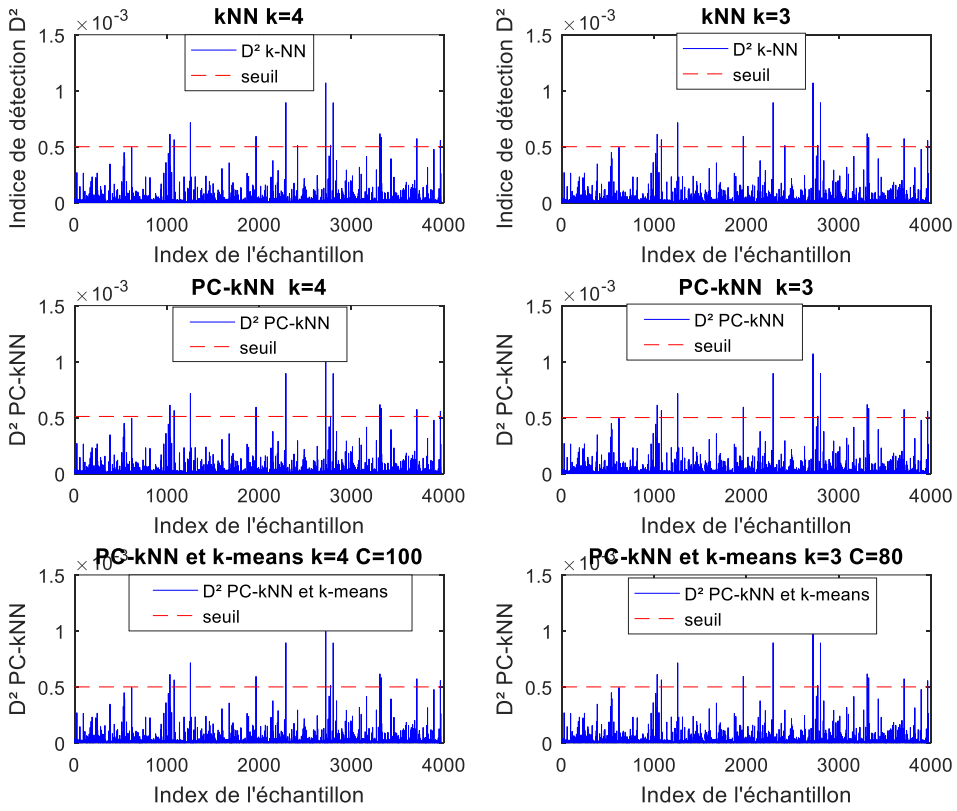

Figure 4.4 : index de détection

Après avoir appliqué la corrélation entre les algorithmes ACP et KNN nommés PC-KNN en l'absence de défauts, l'application de l'algorithme aux données de test en le comparant échantillon par échantillon avec les données d'apprentissage à l'aide de la distance euclidienne, permet d'extraire les données k-plus proches voisins pour chaque échantillon d'essai. Ces k données adjacentes sont ensuite utilisés pour calculer l'indice de détection. La figures (4.4) montre l'évolution de l'indice de détection en l'absence de défauts.

Phase de défaut

Pour la détection d'erreur, nous avons défini une erreur par défaut dans X2 (La poussée brute KN) produite avec une amplitude d'environ 20 % de sa plage de variance apparaissant du temps 2000 à 3973 considéré comme le dernier échantillon dans les données. Comme le montre la figure (4.5).

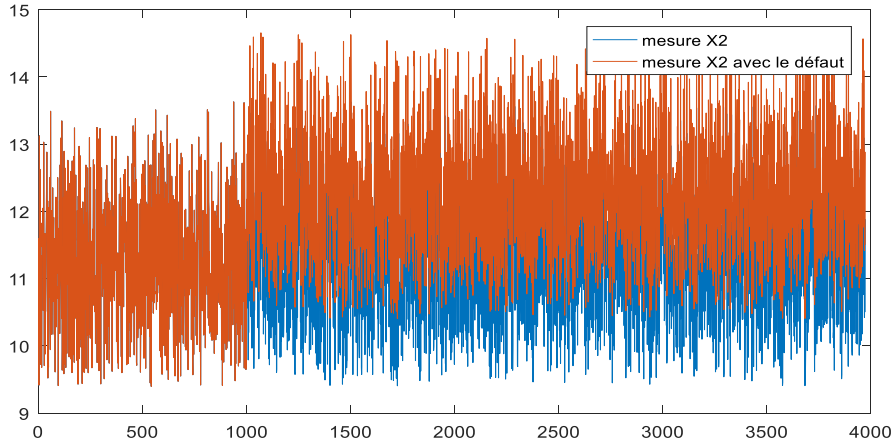

Figure 4.5 : le capteur x2 sans et avec le défaut

-D'après les figures obtenues, on constate qu'il y a une détection de défaut avec une grande sensibilité au défaut.

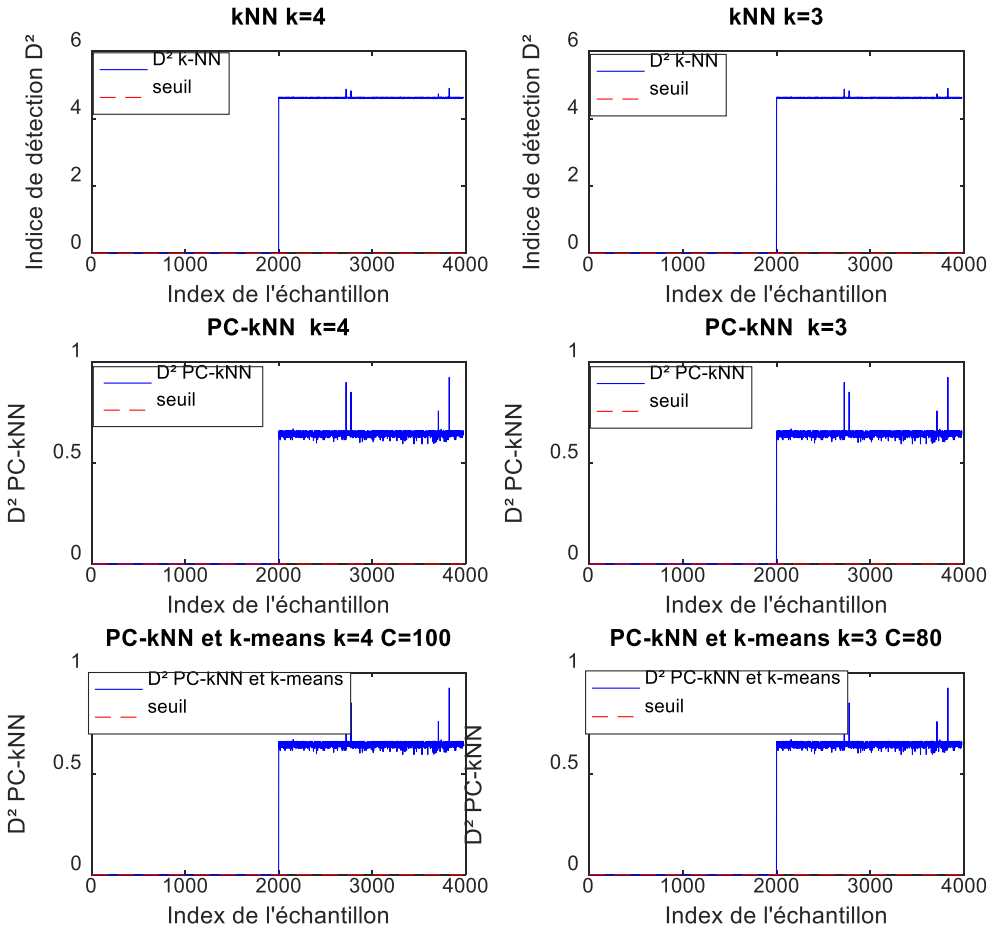

Figure 4.6 : index de détection avec défaut dans x2

En l'absence de défauts, l'application de l'algorithme KNN, PC-KNN et PC-KNN-k-means aux données de test en le comparant échantillon par échantillon avec les données d'apprentissage à l'aide de la distance euclidienne permet d'extraire les données du k plus proche voisin pour chaque échantillon de test. Une fois l'erreur survenue, nous remarquons que le curseur est passé au-dessus du seuil, et cela est dû au défaut ajouté dans X2 comme montré par la figure (4.6).

4.4 Phase de localisation

En fait, la méthode de localisation de défauts par calcul des contributions, dont la variable qui contribue le plus est la variable en défauts. **Les figures (4.7) (4.8) (4.9)** montre le calcul des contributions à l'indice de détection pour chaque variable, où l'on remarque que la variable qui contribue le plus est la variable x2 alors que c'est la x2 qui est en défaut.

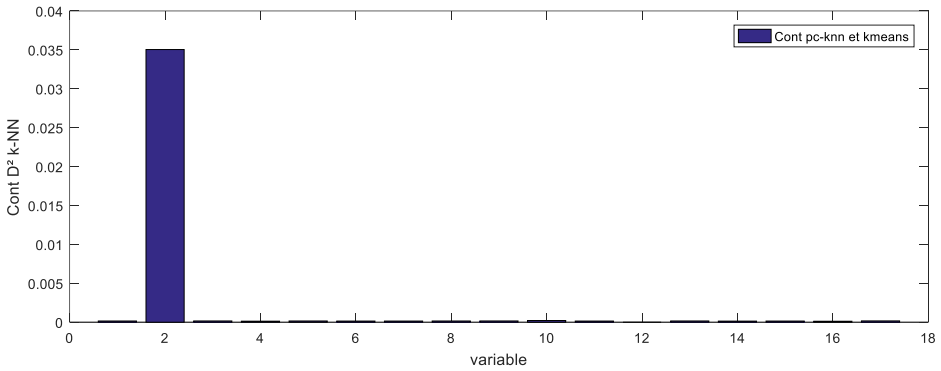

Figure 4.7 : localesation de défaut par method kNN

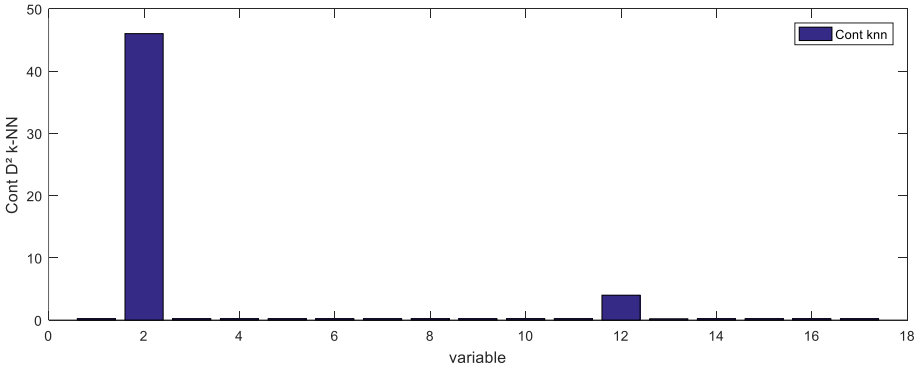

Figure 4.8 : localesation de défaut par method PC-kNN

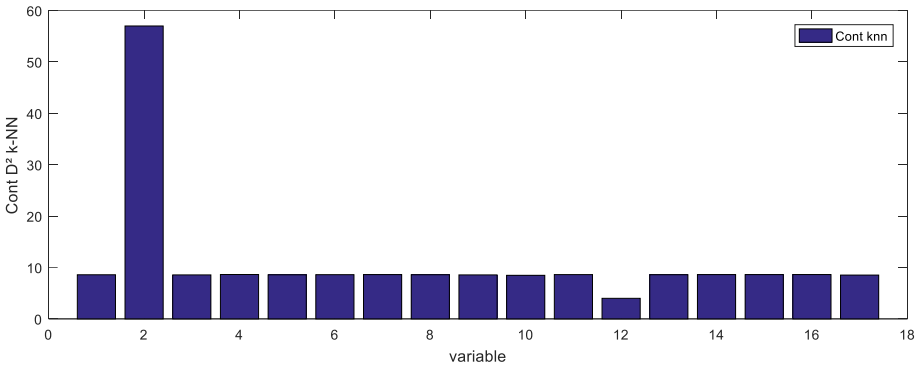

Figure 4.9 : localisation de défaut par méthode PC-Knn et k-center

Deuxième cas d'étude de défaut

Pour la détection d'erreur, nous avons défini une erreur par défaut dans X8 (La vitesse des gaz d'échappement à la dérivation m/s) produite avec une amplitude d'environ 20 % de sa plage de variance apparaissant du temps 2000 à 3973 considéré comme le dernier échantillon dans les données. Comme le montre la **figure (4.10).**

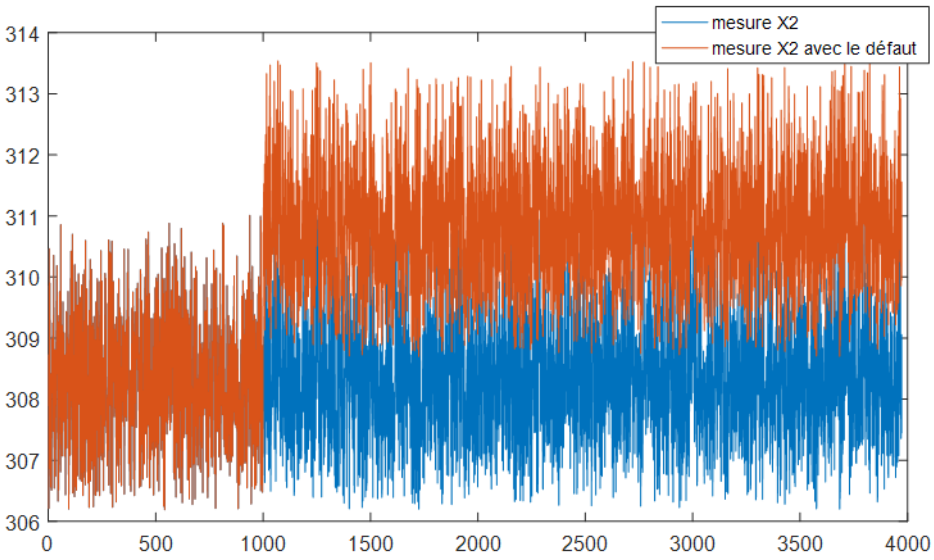

Figure 4.10 : le capteur x8 sans et avec le défaut

Lors de la simulation de l'erreur d'échantillonnage de 2000, nous remarquons que le dessin a dépassé la limite minimale dans toutes les méthodes étudiées KNN, PC-KNN et PC-KNN K-means en raison de défaut. Comme la représentation dans la **figure (4.11)**

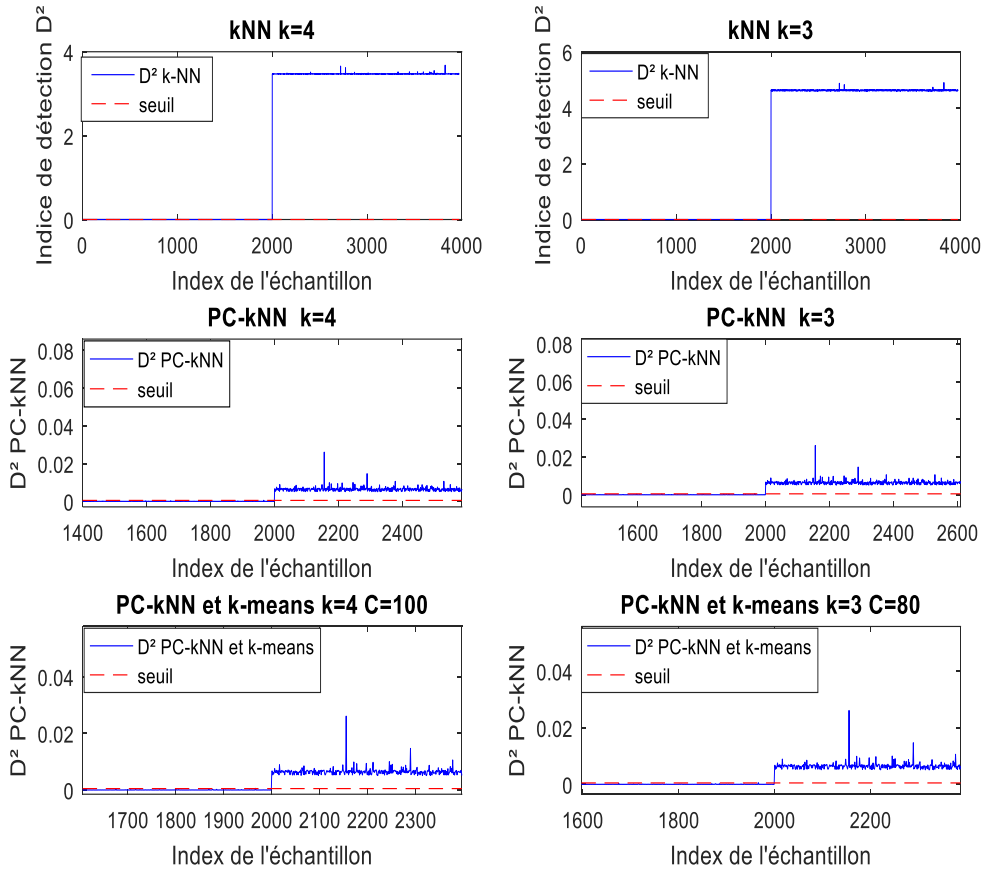

Figure 4.11 : index de détection avec défaut dans x8

En l'absence de défauts, l'application de l'algorithme KNN, PC-KNN et PC-KNN-k-means aux données de test en le comparant échantillon par échantillon avec les données d'apprentissage à l'aide de la distance euclidienne permet d'extraire les données du k plus proche voisin pour chaque échantillon de test. Une fois l'erreur survenue, nous remarquons que le curseur est passé au-dessus du seuil, et cela est dû au défaut ajouté dans X8 C'est-ce que représente la figure (4.11)

4.5 Phase de localisation

En fait, la méthode de localisation de défauts par calcul des contributions, dont la variable qui contribue le plus est la variable en défauts. **Les figures (4.12) (4.13) (4.14)** montre le calcul des contributions à l'indice de détection pour chaque variable, où l'on remarque que la variable qui contribue le plus est la variable x8 alors que c'est la x8 qui est en défaut.

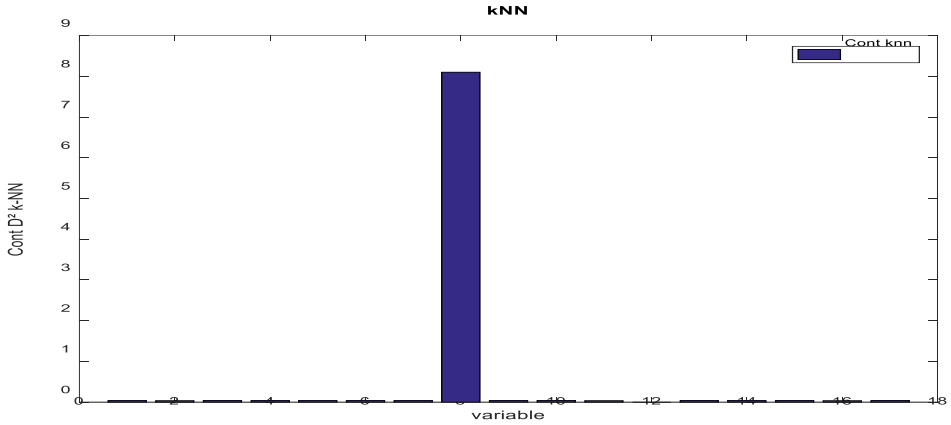

Figure 4.12 : localesation de défaut par method kNN

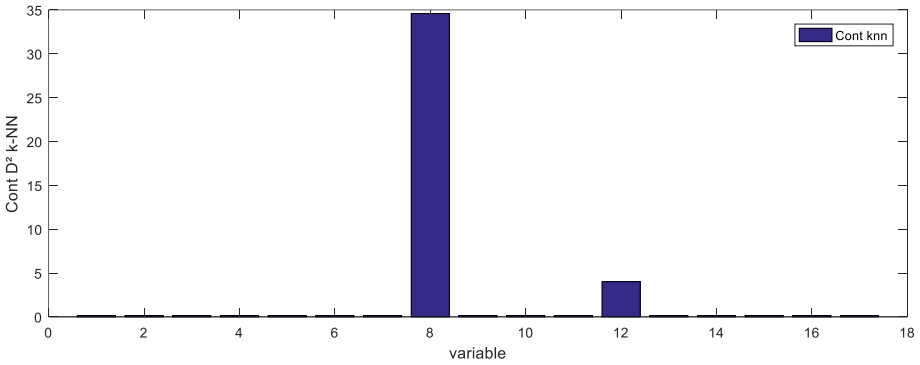

Figure 4.13 : localesation de défaut par method PC-kNN

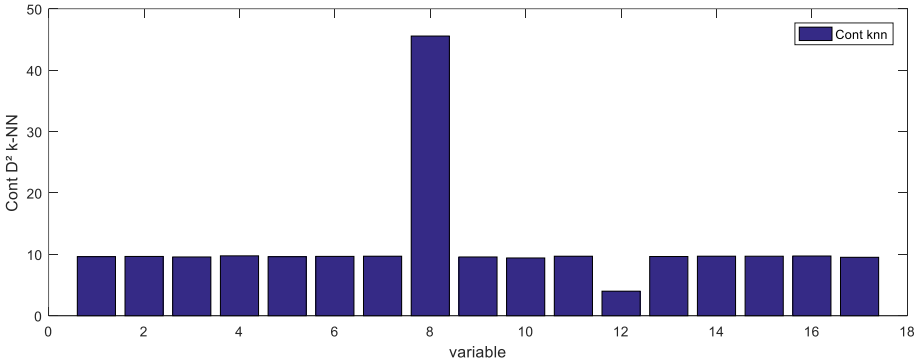

Figure 4.14 : localisation de défaut par méthode PC-Knn et k-center

	kNN	PC-kNN	PC-kNN et k-means	
			c=100	c=80
K=4	3.127384 s	1.311068 s	1.154143s	1.114846s
K=7	3.213125 s	1.241260 s	1.171585s	1.137250s
K=15	3.320164 s	1.247121 s	1.178102s	1.147070s

Tableau 4.2 pour chaque méthode temps

	kNN	PC-kNN	PC-kNN et k-means	
			c=100	c=80
K=4	0.01%	0.01%	0.01%	0.01%
K=7	0.25%	0.23%	0.20%	0.20%
K=15	1.78%	1.75%	1.75%	1.75%

Tableau 4.3 pour chaque méthode

D'après les résultats présentés dans les tableaux (4.2) et (4.3), on remarque qu'en augmentant le nombre de k dans l'algorithme knn, le temps pris augmente, mais lorsqu'on utilise la méthode ACP avec knn, le temps passé devient moindre, et lors de l'ajout de la troisième méthode, k-means est devenu moins, tandis que dans le deuxième tableau Concernant les fausses alarmes, nous ne remarquons pas qu'il y a des changements majeurs, et cela indique que l'algorithme après l'avoir appliqué de trois manières, il n'y a pas eu de changement, c'est-à-dire qu'il a réduit la taille des données tout en préservant les variables les plus importantes dans les données.

4.6 Conclusion

Dans ce chapitre, nous avons appliqué les méthodes KNN, K-means, ACP et PC-KNN, pour la prédiction de défauts dans le cadre de la simulation Monte Carlo pour une turbine. Nous avons réalisé une analyse comparative des résultats que nous avons obtenus, en examinant attentivement les différences en termes de temps de calcul, de détection de défauts, ainsi que les résultats encourageants extraits de cette analyse. De plus, nous avons constaté que la taille des données volumineuses est devenue plus petite tout en préservant les variables importantes.

Conclusion général

En conclusion, l'algorithme k-NN (k plus proches voisins) et l'algorithme k-means sont des techniques d'apprentissage automatique simples mais puissantes pour la classification et la régression et aussi pour la prédiction de défauts. L'algorithme k-NN présente de nombreux avantages, notamment sa simplicité de compréhension et de mise en œuvre, son absence d'hypothèses sur la distribution des données sous-jacentes et sa capacité à gérer à la fois des données continues et catégorielles.

Cependant, certaines limites doivent également être prises en compte. Premièrement, l'algorithme peut être coûteux en calcul, en particulier pour les grands ensembles de données, car il nécessite de calculer les distances entre chaque point de données. Deuxièmement, k-NN est sensible aux caractéristiques non pertinentes dans les données, ce qui signifie que les caractéristiques qui ne sont pas pertinentes pour la prédiction peuvent influencer les résultats. Enfin, il est nécessaire de sélectionner avec soin l'échelle de distance et la valeur de k, car cela peut avoir un impact significatif sur les performances de l'algorithme.

Il est important de considérer leurs avantages et leurs limites propres à chaque application. En matière de détection d'erreurs, la combinaison de l'analyse en composantes principales (PCA) et de k-NN (appelée PC-KNN) peut offrir des avantages significatifs. L'ACP réduit les dimensions des données et identifie les variables les plus importantes, tandis que k-NN classe les nouvelles observations sur la base des exemples les plus similaires. Cette combinaison peut améliorer les performances de détection des erreurs en tenant compte à la fois de la structure globale des données et de la proximité d'observations similaires.

Cependant, il est important de noter que l'efficacité de cette approche peut varier selon le contexte de l'application et la qualité des données utilisées. Une évaluation minutieuse de la méthode PC-KNN est donc recommandée pour s'assurer de sa pertinence dans un cas particulier.

Bibliographie

[1] Support de Cours Diagnostic des Systèmes Dr. CHAKOUR Chouaib Faculté des Nouvelles Technologies d'Information et de Communication Département d'Electronique et des Télécommunications Courriel : chakour.chouaib@univ-ouargla.dz

[2] Diagnostic et surveillance des procédés industriels et de leur environnement sur la base de l'analyse de données THÈSE Présentée en vue de l'obtention du diplôme de DOCTORAT 3eme CYCLE

[3 Salowa METHNANI ,] DOCTORAT - Commande Automatique et Informatique Industrielle, Diagnostic, reconstruction et identification des défauts ,capteurs et actionneurs : application aux station ,d'épurations des eaux usées.

[4] Karim Chabir. Diagnostic de défauts des systèmes contrôlés via un réseau. Autre [cs.OH]. Université Henri Poincaré - Nancy 1, 2011. Français. ffNNT : 2011NAN10044ff. fftel-01746194f

[5] https://hal.univ-lorraine.fr/tel-01750473

[6] https://www.ibm.com/topics/knn.

[7] Lakshmikantha Reddy Somula, M. Meena Computer Science2022 IEEE 4th International Conference on Cybernetics, Cognition and Machine Learning Applications (ICCCMLA)

[8]https://moncoachdata.com/

[9]L.M. Elshenawy, C. Chakour and T.A. Mahmoud, Fault detection and diagnosis strategy based on k-nearest neighbors and fuzzy C-means clustering algorithm for industrial processes, Journal of the Franklin Institute, https://doi.org/10.1016/j.jfranklin.2022.06.022

[11] https://fr.wikipedia.org/

[12] https://biblio.univ-annaba.dz/ingeniorat/wp-content/uploads/2019/09/Rachedi-Temer-Abdelatif.pdf

[13] Mohamed-Faouzi Harkat. Détection et localisation de défauts par analyse en composantes principales.

[14] https://archive.ics.uci.edu/ml/datasets/Gas+Turbine+CO+and+NOx+Emission+Data+Set

[17] https://www.kaggle.com/datasets/muniryadi/gasturbine-co-and-nox-emission-data

[18] https://archive.ics.uci.edu/ml/datasets/Gas+Turbine+CO+and+NOx+Emission+Data+Set

[19]https://aws.amazon.com/what-is/monte-carlo-simulation/#:~:text=History%20of%20the%20Monte%20Carlo,characteristic%20as%20a%20roulette%20game.

[20] Arar K , T adjine S , Modélisation et Diagnostic des systèmes par l'Analyse en Composantes Principales Multi-Echelle (MSPCA) Faculté des Sciences de l'Ingénieur , Département d'Electronique BP. 12, Sidi Amar 23000 Annaba , 2007

[21] Revathi Vankayalapati* , Kalyani Balaso Ghutugade, Rekha Vannapuram, Bejjanki Pooja Sree Prasanna Department of CSE, K-Means Algorithm for Clustering of Learners Performance Levels Using MachineLearning Techniques School of Technology, GITAM (Deemed to be University), Hyderabad 502329, Telangana, India