Advanced Probabilistic & Statistical/Machine learning Models For Anomaly Detection: Application in Telecommunication Industry

Michel Kamel

N°d'ordre NNT : xxx

THESE de DOCTORAT
opérée au sein de
l'Ecole des Mines de Saint-Etienne

Ecole Doctorale N° 488
Sciences, Ingénierie, Santé

Spécialité de doctorat :
Discipline : Mathématiques Appliquées

Soutenue publiquement/à huis clos le 18/07/2023, par :
KAMEL Michel

Advanced Probabilistic & Statistical/Machine learning Models For Anomaly Detection: Application in Telecommunication Industry

Devant le jury composé de :

M. Stéphane Chrétien, Professeur, Université Lyon 2	Président
M. Gille Ducharme, Professeur, Université Montpellier	Rapporteur
M. Stéphane Girard, Directeur de recherche, INRIA Grenoble	Rapporteur
M. Anis Hoayek, Maitre de conférence, Mines Saint-Etienne	Examinateur
Mme Mireille Batton-Hubert, Professeur, Mines Saint-Etienne	Directrice de thèse
Mme Aline Mefleh, Docteur, Université Libanaise	Invitée
M. Kinan Jarrah, Ingénieur, B-yond	Invité

Acknowledgment

First and foremost, I would like to express my deepest gratitude to my thesis supervisor, professor Mireille BATTON-HUBERT, for their continuous guidance, support, and invaluable advice throughout my research journey. Their expertise, patience, and mentorship have been indispensable in helping me navigate the challenges of this research, and I am truly grateful for the opportunity to learn and grow under their tutelage.

I would also like to thank my co-supervisor, Anis HOAYEK, for their insightful feedback, constructive criticism, and encouragement during the various stages of my research. Their expertise and experience have greatly enriched my understanding of the subject matter.

I extend my heartfelt appreciation to the members of my thesis committee, [Committee Members' Names], for their constructive suggestions, insightful comments, and encouragement that helped refine my work and shape my research direction.

My sincere thanks go to the faculty and staff of the GMI and Centre Fayol at Ecole Des Mines de Saint-Etienne for providing a stimulating and nurturing environment for my academic growth. Special thanks to Professors Olivier BOISSIER and Rodolphe LE RICHE for their assistance and support throughout my time at the university.

I would like to acknowledge the financial support provided by B-yond, which enabled me to focus on my research and complete this thesis.

Last but not least, I would like to express my deepest gratitude to my family: my parents, Georges & Mona, for their unwavering love, support, and belief in my abilities, and my partner, Rakel, for their patience, understanding, and constant encouragement throughout this challenging journey. Without their support, this achievement would not have been possible.

Résumé

Les réseaux de dispositifs connectés connaissent une croissance exponentielle à travers le monde et les opérateurs de télécommunication sont chargés de gérer des réseaux vastes et complexes. Ainsi, il est nécessaire de disposer de systèmes intelligents et performants pour aider les ingénieurs dans la maintenance de ces réseaux. Les dispositifs, également appelés éléments de réseaux ou capteurs, rapportent en continu des indicateurs clés de performance (KPI) et utilisent les données associées, combinées à des modèles de détection des anomalies (DA) intelligents, pour prioriser la maintenance de production du réseau.

Une anomalie est une observation ou un événement dans une série temporelle qui a une faible probabilité de se produire dans des circonstances normales. Par conséquent, la survenue d'anomalies est généralement accompagnée de symptômes qui peuvent perturber le fonctionnement d'un système sous-jacent. Les anomalies sont généralement rares et importantes ; elles sont rencontrées dans différents domaines tels que le contrôle du trafic réseau, la détection de fraudes, les dommages industriels, le traitement d'image, etc. La détection des anomalies est bien développée dans la littérature et plusieurs approches et méthodes ont été appliquées pour détecter de tels événements. Ces travaux impliquent différents domaines de recherche tels que l'apprentissage automatique, l'analyse de données, les statistiques paramétriques/non paramétriques, la théorie de l'information, la théorie spectrale, etc. Cependant, la majorité de ces approches et outils sont confrontés à des défis et à des lacunes de recherche de différentes natures qui sont difficiles à surmonter.

Les principales lacunes de recherche auxquels sont confrontés les modèles de détection des anomalies les plus populaires peuvent être résumés comme suit : un temps d'entraînement relativement long ; l'incapacité à quantifier la contribution des différentes variables sous-jacentes à un comportement anormal lors de l'exploration dans un espace multidimensionnel ; l'incapacité à distinguer les anomalies des valeurs aberrantes qui ne sont pas causées par des dysfonctionnements du système; l'incapacité à hiérarchiser les anomalies détectées selon leur niveau de gravité et la difficulté de traiter des variables de différentes typologies (e.g., textuelles, catégorielles, numériques).

Pour surmonter ces limites, nous avons développé plusieurs approches (modèles/algorithmes) de détection des anomalies qui utilisent des données de différentes typologies et de nature diversifiée : probabiliste, statistique et apprentissage automatique. Nos différentes méthodes détectent les comportements anormaux, génèrent des scores d'anomalies, quantifient l'importance des variables sous-jacentes et constituent une première étape essentielle pour déterminer la cause fondamentale des anomalies détectées.

Premièrement, nous proposons un nouvel algorithme de détection d'anomalies, appelé Abnormality Hyper-Cubic Algorithm (AHCA). L'algorithme est conçu pour être non supervisé, indépendant de la distribution des variables sous-jacentes, et il peut mesurer la contribution de chaque variable à un score d'anomalie global. De plus, il peut distinguer entre les vraies anomalies et les valeurs aberrantes qui ne sont pas causées par des dysfonctionnements du système. Le score d'anomalie est calculé dans un contexte multidimensionnel en se basant sur une approche probabiliste et géométrique ayant pour but d'isoler les anomalies dans des zones cubiques ayant une faible masse de probabilité. Ensuite, nous introduisons une mesure, aussi probabiliste, de l'importance et de la contribution de chaque variable à un comportement anormal. Enfin, la théorie de l'information, particulièrement l'entropie de Shannon, nous aide à décider si une observation avec un score d'anomalie élevé est une véritable anomalie ou une valeur aberrante.

En résumé, l'algorithme AHCA présente plusieurs innovations par rapport aux algorithmes de détection d'anomalies existants. Il combine une approche probabiliste avec une topologie géométrique spécifique pour calculer un score d'anomalie pour chaque point de données. De plus, il montre l'importance de chaque variable sous-jacente dans le score global d'anomalie d'une observation, tout en préservant les corrélations entre les variables. De plus, c'est un algorithme qui aide à la discrimination entre anomalie et valeur aberrante.

Deuxièmement, nous proposons une approche adaptée à la détection des anomalies en se basant sur des données catégorielles (des alarmes) provenant d'un réseau de télécommunication. Le modèle consiste à calculer et agréger quatre caractéristiques/signaux pour définir un score d'anomalie sur un certain intervalle de temps. Les caractéristiques sont : le nombre des alarmes ; le temps d'attente entre deux alarmes consécutives ; la fréquence de transition entre les alarmes (modèle markovien) ; la fréquence d'apparition historique des alarmes.

Le score d'anomalie final est obtenu en agrégant les caractéristiques précédemment calculées en utilisant une moyenne pondérée avec des poids optimisés suite à des interactions avec des experts du domaine (SME) et d'une approche de recherche en grille supervisée.

Dans un troisième temps, nous nous concentrons sur les données textuelles qui sont une source d'information cruciale pour la performance des équipements d'un réseau de télécommunications. Ces données textuelles contiennent des informations sur les événements qui se produisent pendant le fonctionnement du système. L'objectif principal est de détecter un intervalle de temps présentant une forte probabilité de comportement anormal en se basant uniquement sur les données textuelles du système. Cette méthode est particulièrement utile lorsque les seules données disponibles sont les journaux système, ce qui est le cas dans de nombreuses applications réelles.

La méthodologie se décompose en deux étapes principales : 1) Le prétraitement des données textuelles en utilisant des techniques de traitement du langage naturel (NLP) ; 2) Le regroupement des données textuelles à l'aide d'un nouvel algorithme de regroupement non supervisé qui tient compte des spécificités des données et évite une étape de vectorisation intermédiaire.

Les trois principales contributions de cette méthode sont:

1) L'adaptation d'un nouvel algorithme de regroupement à la structure des données textuelles en évitant la perte d'information due à une étape de vectorisation;
2) l'extraction d'une structure optimale de l'espace de décision grâce à une technique de regroupement non supervisé pour obtenir un système de notation d'anomalie plus précis et homogène ;
3) L'optimisation des hyper paramètres de l'algorithme proposé en utilisant une approche supervisée innovante qui prend en compte l'interaction avec les experts du domaine.

Quatrièmement, nous proposons une nouvelle approche basée sur la théorie des valeurs extrêmes et des records pour la détection des anomalies. Cette approche se concentre sur le comportement des queues des variables sous-jacentes, plutôt que sur l'ensemble de la distribution, et introduit un nouveau système de notation des anomalies capable de distinguer entre les événements rares et courants.

Après une formalisation mathématique de la théorie des records adaptée à notre contexte, nous avons créé un outil de sélection de variables

permettant de choisir celles ayant le plus d'importance pour être utilisées dans un algorithme de détection des anomalies. Notons que certaines approches de réduction de dimension classiques peuvent ne pas être adaptées à certains contextes d'application, en particulier lorsque l'accent est mis sur le comportement de la distribution des queues des variables. La méthode proposée est donc innovante et spécifiquement adaptée au cas de la détection des anomalies. Elle se concentre principalement sur le comportement des événements extrêmes, en particulier des records supérieurs, pour déterminer quelles variables doivent être sélectionnées.

Ensuite, nous proposons un système de notation pour la détection des anomalies, applicable en une ou plusieurs dimensions. Ce système peut détecter objectivement les anomalies et suggérer des valeurs seuils pour les différentes variables (KPI) sans l'aide d'experts.

Le système de notation proposé pour la détection des anomalies est un algorithme simple qui ne repose sur aucune distribution ou paramètre spécifique. Il est conçu pour être utilisé en tant que système en ligne pour détecter les anomalies avec une complexité de calcul minimale sans risque de sur apprentissage. De plus, le système peut estimer automatiquement la valeur seuil nécessaire pour classer les observations en tant qu'anomalies, assurant ainsi une performance optimale de l'algorithme.

Les quatre modèles/algorithmes ont été testés sur des données réelles de télécommunications et ont démontré d'excellentes performances dans la détection d'anomalies avec des taux d'erreur très faibles.

En conclusion, dans cette thèse, pour répondre aux limites des modèles de détection d'anomalies les plus populaires, nous avons proposé un nouveau modèle géométrique multidimensionnel probabiliste pour rechercher les comportements anormaux dans l'espace de données, générer des scores d'anomalie et quantifier les facteurs d'anomalie. Nous avons introduit également un algorithme pour générer un score d'anomalie final basé sur quatre caractéristiques dérivées des données historiques pour les données d'alarme. En outre, un algorithme pour aider à prétraiter les données textuelles, les regrouper en classes et étiqueter dynamiquement chaque classe comme une anomalie ou non a été développé. Enfin, nous avons proposé une méthode couplant la réduction de dimension et la détection des anomalies basée sur la théorie des records. Dans l'ensemble, cette thèse fournit des méthodes innovantes pour détecter et prioriser les anomalies dans les réseaux de télécommunications et propose des outils puissants pour l'analyse de données et la maintenance des réseaux.

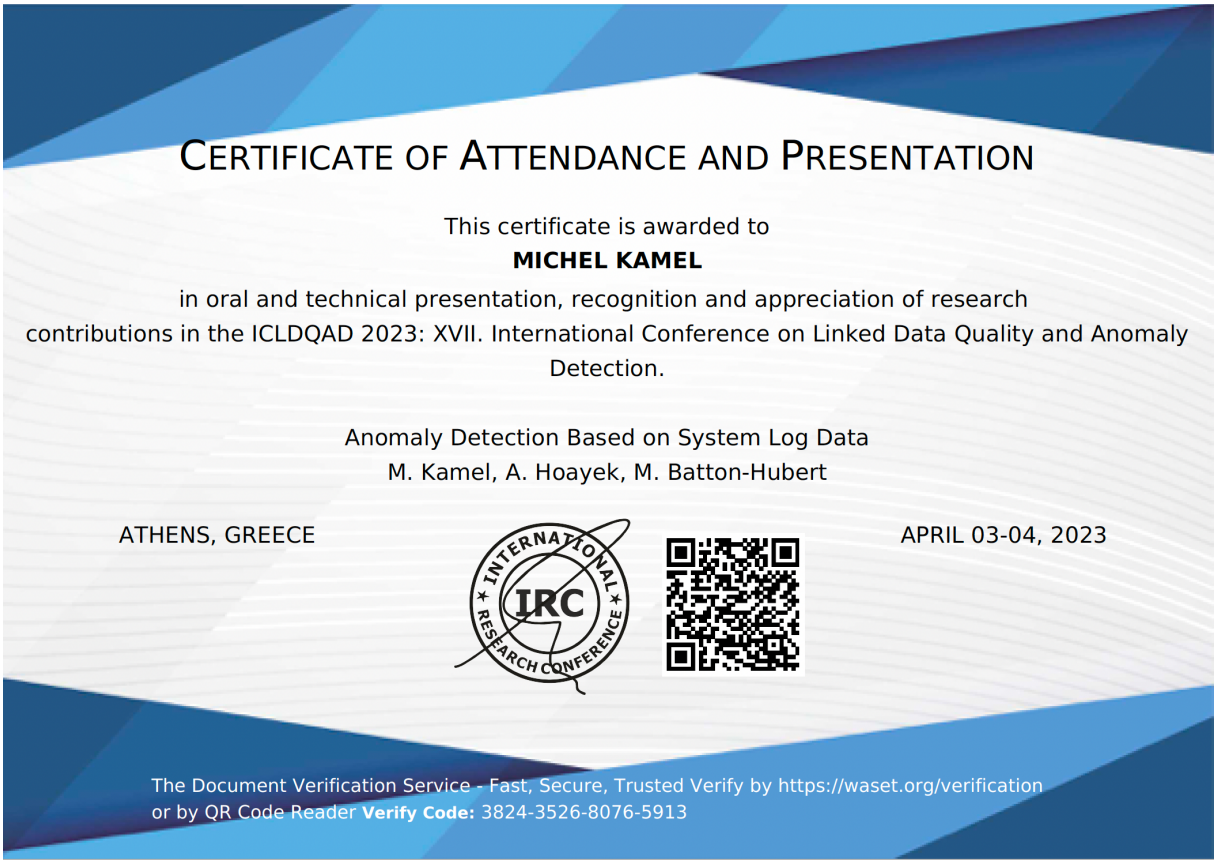
List of Publications

[1] Michel Kamel, Anis Hoayek and Mireille Batton-Hubert - Anomaly Isolation Agents - (Submitted, under second review in Engineering Applications of Artificial Intelligence 2022)

[2] Michel Kamel, Anis Hoayek, Mireille Batton-Hubert - Coupling Variable Selection and Anomaly Detection:... - (Submitted, under first review in Engineering Applications of Artificial Intelligence 2023)

[3] Michel Kamel, Anis Hoayek, Mireille Batton-Hubert - Anomaly Detection based on alarms data - (Submitted and published David C. Wyld et al. (Eds): AI, AIMLNET, BIOS, BINLP, CSTY, MaVaS, SIGI - 2022 pp. 103-114, 2022. CS & IT - CSCP 2022)

[4] Michel Kamel, Anis Hoayek, Mireille Batton-Hubert - Anomaly Detection Based on System Log Data - (Submitted and published in ICLDQAD 2023 : International Conference on Linked Data Quality and Anomaly Detection, Apr 2023, Athenes, Greece.)



CERTIFICATE OF ATTENDANCE AND PRESENTATION

This certificate is awarded to

MICHEL KAMEL

in oral and technical presentation, recognition and appreciation of research contributions in the ICLDQAD 2023: XVII. International Conference on Linked Data Quality and Anomaly Detection.

Anomaly Detection Based on System Log Data
M. Kamel, A. Hoayek, M. Batton-Hubert

ATHENS, GREECE

APRIL 03-04, 2023

The Document Verification Service - Fast, Secure, Trusted Verify by https://waset.org/verification
or by QR Code Reader **Verify Code:** 3824-3526-8076-5913

Table of Contents

Acknowledgment *3*

Résumé *5*

List of Publications *9*

Table of Contents *11*

Chapter 1: Introduction To Our Research *15*

1.1 Background and Motivation **17**

1.2 Research Questions and Objectives **18**

1.3 Literature Review **21**

1.4 Contributions and Novelty **21**

1.5 The importance of anomaly detection in telecommunication industry **22**

1.5.1 Limitations of Traditional Anomaly Detection Techniques 22

1.5.2 Advantages of Machine Learning Algorithms for Anomaly Detection 23

1.5.3 Machine Learning Approaches for Anomaly Detection 23

Chapter 2: B-Yond & Data Description *25*

2.1 Data Overview **28**

2.2 Data value **30**

2.3 Challenges **32**

2.4 Network Topology **34**

Chapter 3: Anomaly Isolation Agents *45*

Abstract **45**

3.1 Introduction **46**

3.2 One-dimensional abnormality score **49**

3.3 Multi-dimensional abnormality hyper-cubic (AHCA): **51**

3.3.1 Abnormality score 51

3.3.2 Individual abnormality contribution 51

3.3.3 Abnormality vs. Outliers 52

3.4 Application **55**

3.4.1 Data description 55

3.4.2 Results and analysis 56

3.4.3 Sampling method 61

3.4.4 Advantages of the proposed algorithm 62

3.5 Conclusion **63**

Appendix: **64**

References **66**

Chapter 4: Combining Numeric KPI and Categorical Alarm Data ***69***

The Complementary Nature of Numeric KPIs and Categorical Alarms **70**

Challenges in Combining Numeric KPIs and Categorical Alarms **70**

Conclusion **71**

Chapter 5: Anomaly Detection Based on Alarms/Events Data ***73***

Abstract **73**

5.1 Introduction **75**

5.2 Methodology **76**

5.2.1 Number of Alarms 77

5.2.2 Inter-arrival time 78

5.2.3 Transition probability 78

5.2.4 Historical frequency 79

5.2.5 Final score and individual contributions 80

5.2.6. Validation and optimization 80

5.3 Application **81**

5.3.1. Data description 81

5.3.2 Results and analysis 82

5.3.3 Advantages of the proposed algorithm 88

5.3.4 Test of independence between different type of alarms 88

5.4 Conclusion and Perspectives **89**

References **90**

Chapter 6: Expanding the Anomaly Detection Horizon to Syslogs ***93***

The Role of Syslogs in Telecommunication Networks **95**

Challenges in Integrating Syslog Data **95**

Conclusion **95**

Chapter 7: Anomaly detection based on SysLogs textual data ***97***

Abstract **97**

7.1 INTRODUCTION **99**

7.2 Methodology **101**

7.2.1 Preprocessing textual data 101

7.2.2 Clustering and structure extraction algorithm 102

7.3 Application **106**

7.3.1 Data description 106

7.3.2 Results and discussions 106

7.3.3 Particularities of the methodology 108
7.4 Conclusion and perspectives 109
References 111
Chapter 8: From whole distribution approach to a tail distribution approach 115
Chapter 9: 117
Record theory for Anomaly detection & information selection 117
9.1 Introduction 119
9.2 Records, an Introduction 120
9.3 Mathematical Formalization 121
9.3.1 Independent but Not Identically Distributed Observations 123
9.3.2 Dependent and Not Identically Distributed Observations 125
9.3.3 Record Model Selection 126
9.4 Variable Selection Based on Record Behavior 127
9.5 Anomaly Scoring System Based on Records Distribution 130
9.5.1 One Dimensional Abnormality Score 130
9.5.2 Multidimensional Abnormality Score 132
9.6 Real-World Data Application 133
9.7 Conclusion 138
References 139
Chapter 10: Conclusion and research perspective 141

Chapter 1: Introduction To Our Research

In recent years, the rapid growth of the telecommunications industry has led to an unprecedented expansion of network infrastructures and an exponential increase in data traffic. The complexity of these network systems and the ever-increasing volume of data generated by users and applications have made it imperative for network operators to maintain a high level of service quality and reliability. As a result, the identification and resolution of network anomalies have become critical challenges for telecommunications operators worldwide.

Anomaly detection is a fundamental process that involves the identification of unusual behavior or data patterns in networks, which might signal malicious activities, system failures, or other forms of network degradation. Traditional methods for anomaly detection, such as threshold-based and rule-based systems, have become less effective due to the limitations in scalability, adaptability, and accuracy when dealing with the dynamic and evolving nature of modern networks.

Anomaly detection has become a crucial component in many fields, including cybersecurity, healthcare, finance, and manufacturing. The increasing complexity and scale of data generated in these domains have necessitated the development of more sophisticated techniques for identifying unusual patterns or outliers. This PhD research thesis aims to explore novel methods for anomaly detection, evaluate their effectiveness in various real-world scenarios, and contribute to the existing body of knowledge by addressing specific limitations in current approaches.

1.1 Background and Motivation

The importance of anomaly detection has grown exponentially with the rise of big data and the need to maintain high-quality datasets in various industries. Real-world problems that can be addressed through effective anomaly detection include fraud detection, network intrusion detection, fault detection in manufacturing processes, and early identification of disease outbreaks in healthcare settings. The potential impact of this research is significant, as advancements in anomaly detection can contribute to cost reduction, improved efficiency, and enhanced decision-making in multiple domains.

In the following, the most popular application of anomaly detection in different domains:

Fraud Detection:
- Credit card fraud detection
- Insurance fraud detection
- Tax evasion detection

Network Intrusion Detection:
- Detecting unauthorized access
- Identifying distributed denial-of-service (DDoS) attacks
- Discovering malware activity

Fault Detection in Manufacturing Processes:
- Equipment malfunction identification
- Quality control and defect detection
- Predictive maintenance

Early Identification of Disease Outbreaks:
- Monitoring disease spread patterns
- Detecting emerging epidemics
- Identifying potential outbreaks from electronic health records

1.2 Research Questions and Objectives

The main research question guiding this thesis is: How can novel anomaly detection techniques be developed and applied to improve performance, scalability, and adaptability in diverse real-world scenarios? The objectives of the research are as follows:

- Investigate the limitations of existing machine learning anomaly detection methods in handling large-scale, diverse, and dynamic datasets.

- Develop novel techniques that address these limitations and demonstrate their effectiveness in a range of applications.

- Evaluate the performance of the proposed methods against established benchmarks and state-of-the-art approaches.

To address the research question and objectives, a comprehensive and systematic approach will be taken. This section provides a more detailed overview of the three main objectives.

Objective 1:

Investigate the limitations of existing machine learning anomaly detection methods:

The first objective of this research is to investigate the limitations of current machine learning-based anomaly detection methods. This will involve conducting a thorough literature review to identify and understand the common issues faced by researchers and practitioners in various fields when using these methods. The limitations of existing methods can include:

- High dimensionality: As the dimensionality of a dataset increases, many existing methods become less effective in detecting anomalies. The curse of dimensionality is a known issue that makes it challenging for traditional distance-based methods to identify outliers in high-dimensional spaces.

- Dynamic data: In many real-world applications, data can be dynamic and constantly changing. Some existing methods are not well-suited for detecting anomalies in such situations, as they may require periodic retraining or adaptation to accommodate the evolving data distribution.

- Noisy and imbalanced data: Anomaly detection methods can be sensitive to noisy or imbalanced data. Noisy data can lead to false positives, while imbalanced data can result in many false negatives due to the rarity of actual anomalies.

- Scalability: As the volume of data continues to grow, the scalability of anomaly detection methods becomes increasingly important. Many existing methods may struggle to process large datasets efficiently, making it difficult to apply them in real-world scenarios.

Objective 2:

Develop novel techniques that address these limitations:

The second objective of this research is to develop novel anomaly detection techniques that address the limitations identified in Objective 1. This will involve:

- Designing novel algorithms: To overcome the limitations of existing methods, new algorithms will be developed that can effectively detect anomalies in large-scale, diverse, and dynamic datasets. These algorithms will take into consideration the specific challenges identified in Objective 1 and will be designed to perform well in a variety of situations.

- Leveraging advanced machine learning techniques: The proposed methods may involve the use of advanced machine or/and statistical learning techniques to enhance their ability to detect anomalies. These techniques can offer improved performance and adaptability compared to traditional methods.

- Feature engineering and dimensionality reduction: To tackle the issue of high dimensionality, the proposed methods may involve feature engineering and dimensionality reduction techniques to simplify the data and make it more manageable. This can help improve the performance of the anomaly detection algorithms and reduce computational complexity.

- Incorporating domain knowledge: Domain knowledge can be incorporated into the design of the proposed anomaly detection methods to improve their effectiveness in specific application areas. This can involve tailoring the methods to the unique characteristics and requirements of each domain, resulting in more accurate and robust detection of anomalies.

Objective 3:

Evaluate the performance of the proposed methods:

The third objective of this research is to evaluate the performance of the proposed methods against established benchmarks and state-of-the-art approaches. This will involve selecting appropriate datasets: To ensure a comprehensive evaluation, various datasets from different domains will be selected, representing diverse data characteristics and real-world challenges. These datasets may include both synthetic and real-world data, allowing for a thorough assessment of the proposed.

1.3 Literature Review

A comprehensive review of the existing literature on anomaly detection will be conducted, examining the various techniques used in different domains. This review will cover statistical, machine learning, and deep learning-based approaches, discussing their strengths, limitations, and potential for improvement. Additionally, the literature review will explore the specific challenges faced by anomaly detection in various industries, providing a context for the development of novel methods in this research.

The literature review will be introduced gradually at the beginning of each chapter based on the developed methodology and the typology of the application data.

1.4 Contributions and Novelty

The key contributions of this thesis will be the development of novel anomaly detection methods, offering improvements in performance, scalability, and adaptability. These new techniques will address the limitations identified in the literature review and demonstrate their effectiveness in diverse applications. Furthermore, the research will contribute to the understanding of how different techniques perform under varying conditions and how they can be combined or adapted to optimize results.

This thesis will introduce four novel anomaly detection algorithms and their contributions.

1.5 The importance of anomaly detection in telecommunication industry

Telecommunication networks have become the backbone of modern society, enabling seamless connectivity and communication across the globe. Network telco operators are responsible for managing and maintaining the infrastructure that supports these networks, ensuring they remain reliable, secure, and efficient. One of the critical aspects of managing these networks is the detection and resolution of anomalies, which can lead to service degradation, system failures, or cyber threats. With the rapid advancement of technology and the increasing complexity of networks, the importance of utilizing machine learning algorithms for anomaly detection has become more evident than ever before. In the following sub-sections, we explore the reasons why telco operators are increasingly interested in adopting machine learning-based approaches for network anomaly detection.

1.5.1 Limitations of Traditional Anomaly Detection Techniques

Traditional anomaly detection techniques, such as threshold-based and rule-based systems, have several limitations in the context of modern telecommunication networks. These limitations include:

- Scalability: As networks grow in size and complexity, the volume of data generated by the network also increases exponentially. Traditional methods struggle to keep up with this rapid expansion, leading to longer processing times and decreased accuracy in detecting anomalies.

- Adaptability: Telecommunication networks are highly dynamic, with constantly changing network topologies and traffic patterns. Rule-based and threshold-based techniques often require manual intervention to update rules and thresholds, making it difficult to adapt to these changes in real-time.

- High False Positive and False Negative Rates: Traditional techniques may not always accurately distinguish between normal and anomalous behavior, resulting in false alarms or undetected threats.

1.5.2 Advantages of Machine Learning Algorithms for Anomaly Detection

Machine learning algorithms offer several advantages over traditional techniques in addressing the limitations outlined above:

- Scalability: Machine learning algorithms can handle large amounts of data, enabling them to scale with the growing network size and complexity. This is particularly important given the increasing adoption of 5G, IoT, and other emerging technologies that generate vast amounts of data.

- Adaptability: Machine learning models can automatically learn from the data, adapting to changing network conditions and traffic patterns without requiring manual intervention. This ensures that the anomaly detection system remains effective even as the network evolves.

- Improved Accuracy: By leveraging the power of advanced algorithms, machine learning techniques can better model complex relationships between network features, leading to more accurate anomaly detection and reduced false positive and false negative rates.

1.5.3 Machine Learning Approaches for Anomaly Detection

Machine learning techniques can be broadly classified into supervised, unsupervised, and reinforcement learning. Each of these approaches has its advantages and can be applied to different aspects of anomaly detection in telecommunications:

- Supervised Learning: Supervised learning techniques are trained using labeled data, where the model learns to predict the output (anomaly or normal) based on input features. These techniques, such as Support Vector Machines (SVM) and Artificial Neural Networks (ANN), have shown promising results in detecting known anomalies in network traffic.

- Unsupervised Learning: Unsupervised learning techniques do not require labeled data and instead identify anomalies based on the structure and patterns within the data itself. Techniques such as

clustering and autoencoders have been used to uncover previously unknown anomalies or to detect outliers in the data.

- Reinforcement Learning: Reinforcement learning techniques optimize decision-making based on rewards and penalties received for different actions. These algorithms can be applied to dynamic network management tasks, such as adaptive traffic routing or resource allocation, to minimize the occurrence of network anomalies.

As network telecom operators continue to face challenges in maintaining reliable and secure telecommunications infrastructures, the adoption of machine learning algorithms for anomaly detection becomes increasingly important. By leveraging the advantages of machine learning techniques, operators can enhance their services and customer experience, hence the importance of our research and methodologies for telecom operators.

Chapter 2: B-Yond & Data

Description

Radio Access Networks (RANs) are essential components of modern telecommunication systems, enabling seamless connectivity between end-users and core networks. The performance and stability of RANs are of utmost importance to ensure reliable and high-quality communication services. As network operators strive to maintain and optimize RAN infrastructure, it is crucial to have a comprehensive understanding of the various types of data generated within these networks. Specifically, the analysis of performance Key Performance Indicators (KPIs), alarms, and system logs can provide valuable insights into the overall health and functioning of the RAN.

Performance KPIs are quantifiable metrics that provide an objective assessment of the RAN's performance, covering aspects such as availability, capacity, and quality of service. These KPIs are typically collected periodically and aggregated over specified time intervals to facilitate monitoring, troubleshooting, and optimization efforts. Examples of performance KPIs include call setup success rate, dropped call rate, and data throughput.

Alarms, on the other hand, are generated by network elements within the RAN to indicate potential issues or failures. These alarms can be triggered by a variety of events, ranging from equipment malfunctions and configuration errors to external factors such as environmental conditions or network attacks. Timely detection and resolution of alarms are crucial in maintaining the integrity and availability of the RAN infrastructure.

System logs (Syslogs) are event logs produced by network elements that record operational events and other relevant information. Syslogs provide a detailed view of network activity, offering insights into both normal and anomalous behavior. The analysis of syslogs can aid in identifying potential issues or inefficiencies, as well as assisting in troubleshooting and root cause analysis.

In this chapter, we aim to provide a comprehensive analysis of the data generated from RAN networks, including performance KPIs, alarms, and syslogs. We will explore the characteristics of these data types, their interdependencies, and their significance in maintaining and optimizing RAN performance. Additionally, we will investigate the application of advanced data processing and analytics techniques, such as machine learning and big data platforms, to derive actionable insights that can contribute to the improvement of RAN operations and management. By developing a deeper understanding of RAN-generated data, network operators will be better

equipped to maintain a high level of service quality and reliability for their customers.

2.1 Data Overview

Here's some information on KPIs (Key Performance Indicators) and alarms in network communication data and flows, including what they are and how they work.

In the field of network communication, KPIs are used to measure the performance of various network elements and systems. They are a set of quantifiable measurements that indicate how well a network element is functioning. KPIs can be used to monitor network performance, identify potential issues, and track progress toward performance goals.

KPIs can vary depending on the network element or system being monitored. For example, KPIs for a router might include measures of latency, packet loss, and throughput. KPIs for a cellular network might include measures of call quality, handover success rates, and data transfer rates. Regardless of the specific KPIs being monitored, they are generally used to provide an overall picture of the performance of the network element in question.

In addition to KPIs, network elements may also provide alarms when certain conditions are met. Alarms are used to indicate when something has gone wrong or when a specific threshold has been reached. For example, a router might generate an alarm when packet loss exceeds a certain threshold, indicating that there is a problem with the network connection. Alarms can be used to alert network administrators to potential issues so that they can take corrective action before the problem becomes more serious.

Alarms can be generated by a wide range of network elements, including core and transmission elements. Core elements are typically used to route traffic across a network, while transmission elements are used to transport data over long distances. Both types of elements are critical to the overall performance of a network and can generate alarms when issues arise.

Overall, KPIs and alarms are important tools for monitoring and maintaining network performance. By providing insight into how well network elements are functioning and alerting administrators to potential issues, they help ensure that networks operate efficiently and effectively.

Going deeper into the data, in cellular networks, cells are the basic units that provide radio coverage in a particular geographic area. Each cell is typically served by a base station or cell site, which transmits and receives radio signals to and from mobile devices within the cell. The size of each cell can vary depending on factors such as population density and terrain.

Counters are a type of KPI that are used to track specific aspects of cellular network performance. They are typically associated with individual cells and can provide detailed information on factors such as call drops, handover success rates, and data throughput. Counters are often used to identify potential issues with a particular cell or set of cells and can be used to guide troubleshooting and optimization efforts.

In addition to counters, cellular networks also generate large amounts of raw data that can be used to monitor and optimize network performance. This data can include information such as call logs, location data, and signal strength measurements. Raw data is typically collected by network elements such as base stations, switches, and routers, and can be processed and analyzed using specialized software tools.

By monitoring and analyzing this data, network operators can gain insights into how their networks are performing and identify potential areas for improvement. For example, by analyzing call logs and location data, operators can identify areas of poor coverage and take steps to improve network performance in those areas. Similarly, by analyzing signal strength data, operators can identify potential interference sources and take steps to mitigate the impact of that interference.

Overall, cells, counters, and raw data generated in network communication are critical tools for monitoring and optimizing network performance. By providing detailed insights into network performance, they help ensure that networks operate efficiently and effectively, and can help network operators to identify and address potential issues before they become more serious.

Time series KPIs, counters, and raw data generated by network communication equipment can vary depending on the equipment vendor and the specific type of equipment being used.

Different equipment vendors may have different approaches to how they measure and report on network performance. For example, one vendor may emphasize certain KPIs or counters over others, or may use different terminology to describe the same performance metrics. Additionally, vendors may use different algorithms or methodologies to calculate performance metrics, which can result in variations in reported values.

Furthermore, the specific KPIs, counters, and raw data generated by different types of network equipment can vary as well. For example, the KPIs and counters used to measure the performance of a base station in a cellular network may be different from those used to measure the performance of a router in a wired network.

It is important for network operators to be aware of these variations and to ensure that they have a thorough understanding of the specific KPIs, counters, and raw data generated by the equipment they are using. This can help operators to monitor and optimize network performance, and to ensure that they are getting the most out of their equipment investment more effectively.

2.2 Data value

KPIs and counters are important in network communication because they provide valuable information on network performance and help network operators to identify potential issues and optimize network performance. Here are some specific reasons why KPIs and counters are important:

- They provide insight into network performance: KPIs and counters provide a way to measure and monitor the performance of network elements and systems. By tracking performance over time, operators can gain insight into how the network is functioning and identify potential areas for improvement.

- They help identify issues and troubleshoot problems: When network performance falls below a certain threshold, alarms and alerts can be triggered, indicating that there is an issue that needs to be addressed. KPIs and counters can help operators to quickly identify the source of the problem and take corrective action before the problem becomes more serious.

- They can guide optimization efforts: By tracking performance metrics over time, operators can identify areas where network performance is suboptimal and take steps to optimize the network. This might involve adjusting network configurations, upgrading equipment, or implementing new technologies to improve network performance.

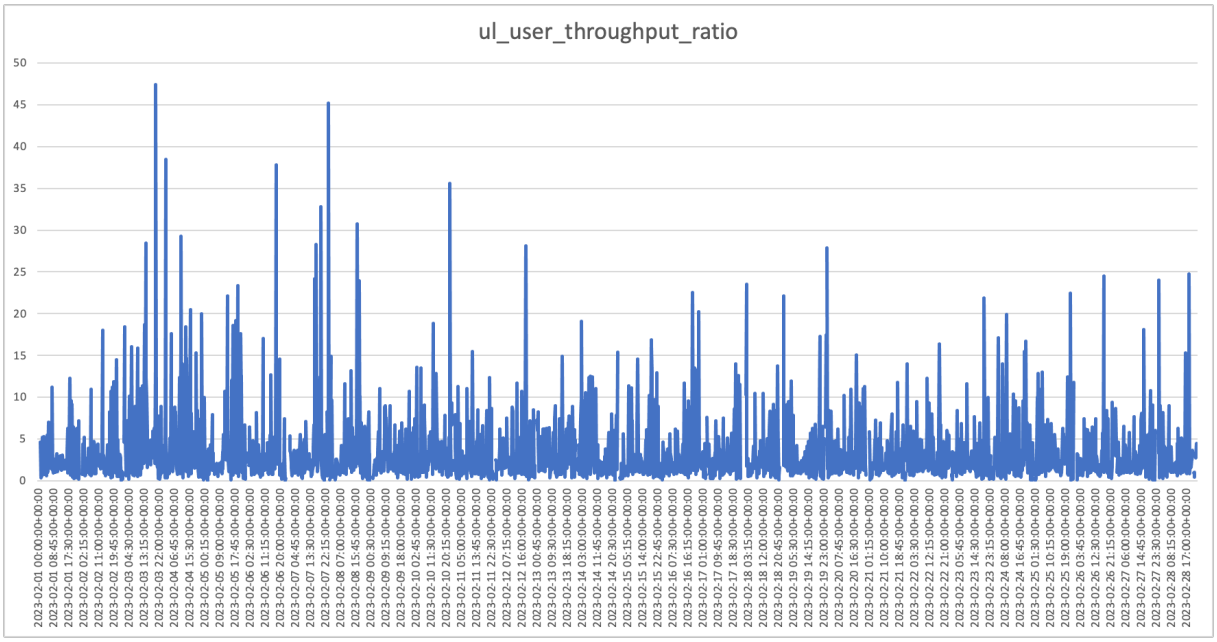
- They can help ensure quality of service: KPIs and counters can be used to track the performance of specific aspects of the network, such as call quality or data throughput. By ensuring that these metrics meet certain thresholds, network operators can help to ensure that their customers receive a high-quality experience when using the network.

Here is a sample of KPIs generated by a real network and provided for our research:

datetime_key	cell_id	site_name	initial_e_rab_set up_succ _rate_ra tio	rach_suc c_rate_r atio	rrc_esta b_succ_r ate_rati o	s1_conn _succ_ra te_ratio	session_ drop_ra tio	average _ue_pdc p_dl_lat ency_rat io	dl_packet _loss_rat e_overall _ratio	volte_er ab_drop _ratio	dl_user_ through put_rati o	ul_user_ through put_rati o	dl_prb_ utilizatio n_ratio	cell_pagi ng_disca rded_rat io_ratio	s1_pagin g_discar ded_rati o_ratio	initial_e _rab_set up_succ _rate_d	rach_suc c_rate_d
2023-02-07 09:30:00+00:00	52091670	IBS0355_BB	0.99824715	0.63592233	0.99704142	1	0.00085985	3.71325938	0.017323746	0	7.73285178	0.26840102	0.17333333	0		1141	1236
2023-02-09 04:30:00+00:00	192746795	GD855_BB2	1	0.5312085	1	1	0	3.73308864	0.000164912	0	110.046245	1.15810956	0	0		493	753
2023-02-09 04:30:00+00:00	192888353	GD1444_BB2	1	0.4587156	1	1	0	4.71314143	0.016800378	0	24.7616137	2.225	0	0		158	218
2023-02-09 04:00:00+00:00	187396889	OD1300_BB3	1	0.23463687	1	1	0	7.48753224	0.002935626		13.8011594	0.88015979	0	0		39	179
2023-02-09 04:30:00+00:00	187396889	OD1300_BB3	1	0.68707483	1	1	0	5.22593954	0.000427767	0	99.2996902	5.74418605	0	0		121	147
2023-02-09 04:45:00+00:00	192888353	GD1444_BB2	1	0.61373391	1	1	0	5.6392225	0.012441055		23.9279314	4.94849785	0	0		195	233
2023-02-09 04:45:00+00:00	192566571	GD492_BB2	1	0.54133333	1	1	0.00518135	4.80774777	0.000169593	0	21.3050837	0.07187918	0	0		182	375
2023-02-09 04:45:00+00:00	187396889	OD1300_BB3	1	0.71111111	1	1	0	5.02842656	0.002777582		40.3486391	5.87830688	0	0		135	180
2023-02-09 04:30:00+00:00	192566571	GD492_BB2	1	0.88505747	1	1	0	4.12228695	7.75E-05		27.1313646	19.3261477	0.17333333	0		145	174
2023-02-09 08:30:00+00:00	52091670	IBS0355_BB	1	0.64801298	0.99722992	1	0.00164204	3.93228034	0.007715863	0	9.39737909	2.99849576	0.18666667	0		1208	1233
2023-02-09 08:45:00+00:00	187396889	OD1300_BB3	1	0.75366569	1	1	0	4.07899405	0.002361134	0	63.7226107	29.9192406	0	0		403	341
2023-02-09 08:00:00+00:00	192566571	GD492_BB2	1	0.96244132	1	1	0	3.89926081	-0.00035837	0	40.7523054	8.71551724	0	0		210	213
2023-02-09 08:15:00+00:00	192746795	GD855_BB2	1	0.5971564	1	1	0	3.81709621	-5.02E-05	0	64.2381032	5.87331081	0.02666667	0		337	422
2023-02-09 08:00:00+00:00	192888353	GD1444_BB2	1	0.50909091	1	1	0	7.7178122	0.003141748	0	47.1005747	5.56637168	0	0		113	165
2023-02-09 08:45:00+00:00	192888353	GD1444_BB2	1	0.54216868	1	1	0	5.42629009	0.007166451	0	66.3493235	38.445395	0	0		145	249
2023-02-09 08:30:00+00:00	192566571	GD492_BB2	1	0.97747748	1	1	0	4.72419188	-3.26E-05		47.2263151	4.78070175	0	0		248	222
2023-02-09 08:30:00+00:00	192888353	GD1444_BB2	1	0.57990868	1	1	0	5.03879969	0.005096247	0	85.3466721	1.06576439	0.00666667	0		147	219
2023-02-09 08:30:00+00:00	192746795	GD855_BB2	1	0.83121019	1	1	0	3.45971753	-0.00031369	0	74.3793076	6.50925926	0.01333333	0		343	314
2023-02-09 08:45:00+00:00	192746795	GD855_BB2	1	0.73218673	1	1	0	4.18597025	2.25E-05	0	59.1579041	4.36862442	0	0		387	407
2023-02-09 08:15:00+00:00	52091670	IBS0355_BB	0.99767261	0.71636953	1	1	0.00075415	4.41185632	0.011675438	0	15.2483833	1.62125585	0.13333333	0		1289	1234
2023-02-09 08:00:00+00:00	192746795	GD855_BB2	1	0.90041494	1	1	0	3.41160351	-0.0011736	0	64.7940796	28.7361809	0.00666667	0		244	241
2023-02-09 08:45:00+00:00	192566571	GD492_BB2	1	0.9760479	1	1	0.00544959	6.17835963	0.018185802	0	13.5033181	0.21637417	0	0		348	334
2023-02-09 08:15:00+00:00	192888353	GD1444_BB2	1	0.69148936	1	1	0	5.34083869	0.000573432		139.295589	3.06849315	0	0		183	188
2023-02-09 08:45:00+00:00	52091670	IBS0355_BB	1	0.57757576	0.99760479	1	0.0020548	4.07139868	0.004461133	0	12.5998018	1.20767777	0.26666667	0		1429	1650
2023-02-09 08:00:00+00:00	52091670	IBS0355_BB	0.99749373	0.71521336	0.99855073	1	0	3.92723511	0.01153177	0	7.98001394	1.17915321	0.14	0		1197	1078
2023-02-07 10:45:00+00:00	52091670	IBS0355_BB	1	0.54779169	0.99742268	1	0.00291758	4.11097647	0.016670145	0	14.3484788	0.92995807	0.14	0		1352	1517
2023-02-07 10:30:00+00:00	52091670	IBS0355_BB	0.99919355	0.59728814	0.99863946	1	0.00387898	4.24240777	0.005999698	0	14.7749115	4.93429272	0.11333333	0		1240	1475
2023-02-07 10:00:00+00:00	52091670	IBS0355_BB	0.99909091	0.53455571	0.99840764	1	0.00087719	3.85395962	0.013325489	0	6.69183469	0.94467028	0.22	0		1100	1418
2023-02-07 10:00:00+00:00	187396889	OD1300_BB3	1	0.7687747	1	1	0.0019305	4.23451118	0.003415305	0	47.2937018	1.77314728	0.01333333	0		556	506

Columns are KPIs measured at 15 min level for each cell. More details for each column will be elaborated in each chapter based on its usage.

We can plot each KPI to see its trend as a time series as the following:



2.3 Challenges

There are several challenges that network operators may face when working with this data. We are dividing it into 2 categories, first one about general ones and then detailing one specific challenge about size of the data. Here we will start listing five challenges with their description:

- Data integration: Network operators often use a variety of different equipment from different vendors, and each vendor may use different data formats and standards to report KPIs and counters. Integrating data from multiple sources can be challenging, and network operators may need to invest in specialized software or data integration tools to ensure that data can be easily accessed and analyzed.

- Data visibility: With so much data being generated, it can be difficult to identify the most important KPIs and counters to monitor. Network operators need to prioritize their data monitoring efforts to focus on the metrics that have the greatest impact on network performance and user experience.

- Data interpretation: Analyzing KPIs and counters requires expertise in both network communication and data analysis. Network operators may need to invest in training for their staff to ensure that they have the necessary skills to analyze and interpret this data effectively.

- Network complexity: Modern networks are becoming increasingly complex, with a growing number of connected devices, applications, and services. This complexity can make it more difficult to identify the root cause of network performance issues, and can make it harder to optimize network performance.

- Automation: As the volume of data generated by KPIs and counters continues to grow, it may become more difficult for human operators to analyze and interpret all of the data. Network operators may need to invest in automated tools and machine learning algorithms to help identify performance issues and optimize network performance.

There are challenges associated with the size of the data generated by KPIs and counters in network communication. With the proliferation of connected devices and the growth of data-intensive applications, the amount of data

generated by network elements has increased exponentially in recent years. Here are some specific challenges associated with the size of this data:

- Storage: The amount of data generated by KPIs and counters can be massive, and storing all of this data can be a significant challenge. Network operators may need to invest in large-scale data storage solutions, such as cloud-based storage or dedicated storage systems, to accommodate this data.

- Processing: Once data has been stored, processing it can also be a challenge. Analyzing massive data sets can be computationally intensive, and network operators may need to invest in specialized hardware or software to process this data efficiently.

- Data quality: With so much data being generated, it can be difficult to ensure that all of the data is accurate and of high quality. Network operators may need to implement quality control measures to ensure that data is accurate and usable for analysis.

- Security: Storing and processing large amounts of data also raises security concerns. Network operators need to ensure that the data is stored securely and that it is only accessible to authorized personnel.

Overall, the size of the data generated by KPIs and counters can pose significant challenges for network operators. However, with the right storage, processing, and security solutions in place, network operators can effectively leverage this data to optimize network performance and deliver a high-quality user experience.

In summary, while the data generated by KPIs and counters can be extremely valuable for optimizing network performance, there are several challenges that network operators may face when working with this data. By investing in the right tools, technologies, and training, however, network operators can overcome these challenges and effectively leverage this data to improve network performance and deliver a better user experience.

2.4 Network Topology

Network topology refers to the way in which devices are connected in a network. There are several main characteristics and types of network topologies, including:

- Physical topology: This refers to the physical layout of devices and cables in the network. The physical topology can affect network performance and reliability. Here are some key details about physical topology:
    a. Devices: Devices in a physical network topology can include computers, routers, switches, hubs, and other network equipment.
    b. Cables: Cables are used to connect devices in the network. The type of cable used can affect the speed and reliability of data transmission.
    c. Connectors: Connectors are used to attach cables to devices. The type of connector used can affect the reliability and performance of the connection.
    d. Network interface cards (NICs): NICs are used to connect devices to the network. The type of NIC used can affect the speed and reliability of data transmission.
    e. Network layout: The physical layout of devices and cables can affect network performance and reliability. For example, if devices are spread out over a large area, data transmission speeds may be slower, and signal interference may be more likely.
    f. Network topology diagram: A network topology diagram is a graphical representation of the physical layout of devices and cables in a network. This can be used to plan and troubleshoot network configurations.
    g. Maintenance: Maintenance of physical network topology involves ensuring that cables, connectors, and devices are in good working condition, and replacing any components that are damaged or malfunctioning.

- Logical topology: This refers to the way in which data is transmitted between devices in the network. The logical topology is independent of the physical topology and can be modified by network protocols. Logical topology is concerned with the way that devices and networks

communicate with each other, rather than how they are physically connected. Here are some key details about logical topology:

a. Communication protocols: Communication protocols determine how data is transmitted between devices on the network. For example, the Transmission Control Protocol (TCP) is a common protocol used for transmitting data between devices.
b. Network addressing: Network addressing refers to the way that devices on the network are identified. IP addressing is a common method used to identify devices on a network.
c. Routing: Routing refers to the process of directing data between devices on the network. Routers are typically used to direct data between networks.
d. Switching: Switching refers to the process of directing data between devices on the same network. Switches are typically used to direct data between devices on a LAN.
e. Network topology diagram: A logical network topology diagram is a graphical representation of the way that data flows between devices on the network.

- Scale: The scale of a network topology refers to the size of the network, in terms of the number of devices and the geographic area it covers. Network topology can vary greatly in scale, from a small LAN (Local Area Network) with just a few devices to a large WAN (Wide Area Network) spanning multiple cities or even countries. The scale of a network topology can impact its design and implementation, as well as its performance and management. Some of the key considerations for network topology at scale include:
    a. Scalability: The ability to add more devices and users to the network without significant performance degradation or management difficulties.
    b. Redundancy: The ability to provide backup connections and hardware to ensure continuous network operation, even in the case of equipment failures or network outages.
    c. Security: The need to protect the network and its data from unauthorized access, and to ensure the integrity and confidentiality of the data being transmitted.
    d. Management: The need to monitor and manage the network efficiently and effectively, including tasks such as performance monitoring, configuration management, and troubleshooting.

e. Cost: The cost of implementing and maintaining the network topology can vary greatly depending on its scale, and can be a significant factor in decision-making.

- Fault tolerance: Fault tolerance refers to the ability of a network topology to continue functioning even in the presence of device failures or network disruptions. In the context of network topology, fault tolerance is a key consideration, especially for large-scale networks that support critical applications or services. Here are some common strategies used to achieve fault tolerance in network topology:
    a. Redundancy: Redundancy involves having duplicate or backup components, such as routers, switches, and servers, that can take over in the event of a failure. This helps ensure that the network continues to operate even if a component fails.
    b. Load balancing: Load balancing involves distributing network traffic across multiple devices, which helps ensure that no single device becomes overloaded and prone to failure.
    c. Automatic failover: Automatic failover is the process of automatically switching to a backup component when a primary component fails. This can be achieved through technologies such as clustering and virtualization.
    d. Monitoring and alerts: Regular monitoring of network components can help detect potential issues before they become major problems. Automated alerts can be set up to notify network administrators of issues, allowing them to take action before a failure occurs.
    e. Disaster recovery planning: In the event of a major failure or outage, having a disaster recovery plan in place can help minimize downtime and ensure that critical services can be restored as quickly as possible.

- Performance: Performance refers to the speed and efficiency of data transmission in the network, which can be affected by factors such as network congestion and bandwidth limitations. A well-designed network topology can help ensure optimal network performance and can provide a reliable and responsive communication platform for users and devices. Here are some key factors that can impact network performance:
    a. Bandwidth: Bandwidth is the amount of data that can be transmitted over a network in a given time period. A network

with higher bandwidth can typically transmit more data in a shorter amount of time, which can result in better performance.

b. Latency: Latency is the amount of time it takes for a packet of data to travel from one point in the network to another. Lower latency can result in faster network performance and better user experience.
c. Network congestion: Network congestion occurs when too many devices or users are competing for the available bandwidth, which can result in slower network performance. Effective network design and management can help minimize congestion and ensure optimal performance.
d. Network topology: The topology of a network can impact its performance, as certain topologies may be better suited for specific types of applications or use cases. For example, a mesh topology may be better suited for high-availability applications that require redundant connections, while a star topology may be more appropriate for small LANs.
e. Quality of Service (QoS): QoS refers to the ability of a network to prioritize certain types of traffic over others, which can help ensure that critical applications or services receive the necessary bandwidth and performance. Effective QoS management can help ensure optimal network performance for all users and devices.

Some common types of network topologies include:

- Bus topology (Figure I): In a bus topology, devices are connected to a single cable that acts as a backbone for the network. Data is transmitted to all devices on the network, and each device filters out data that is not intended for it. The bus network topology is a simple and widely used network topology. In a bus topology, all devices are connected to a single communication line, called a bus. The bus is typically a coaxial cable or twisted pair cable that serves as the main backbone for the network. In a bus topology, all devices on the network receive all data transmitted on the bus. Each device has a unique address, and when a device wants to send data, it broadcasts the data onto the bus. The other devices on the network then examine the address to determine whether the data is intended for them. If the data is not intended for a particular device, that device simply ignores it. The advantages of the bus topology include its simplicity and low cost. Because all devices are connected to a single

communication line, the topology is easy to set up and requires minimal cabling. However, there are also some disadvantages to the bus topology. One disadvantage is that the entire network can be affected if there is a break or fault in the bus. Because all devices rely on the bus for communication, a break in the bus can cause the entire network to fail. Another disadvantage is that the network can become congested if there are too many devices connected to the bus. This can lead to reduced performance and slower data transmission speeds. Overall, the bus topology is a simple and inexpensive network topology that can be effective for small networks with relatively few devices. However, it may not be the best choice for larger networks or those that require high levels of performance or fault tolerance.

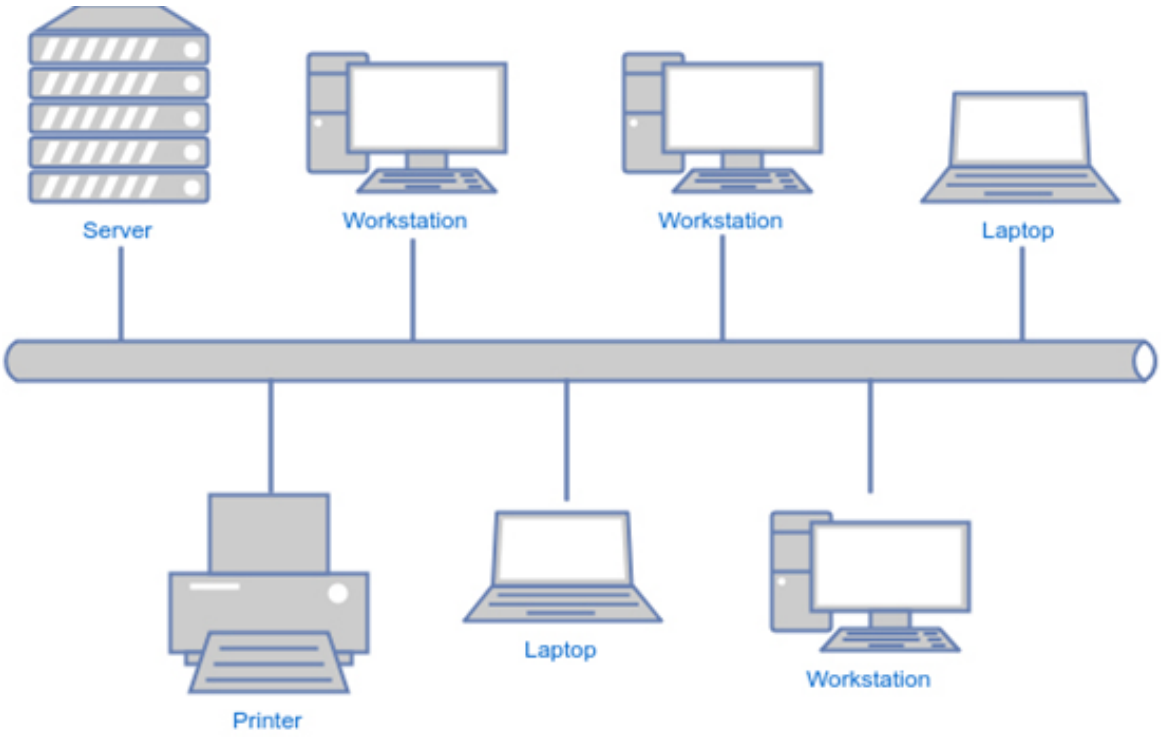


Figure I: Bus Topology Network

- Star topology (Figure II): In a star topology, devices are connected to a central hub or switch. Data is transmitted through the hub, which manages data transmission to the appropriate devices. The star

network topology is another commonly used network topology. The hub or switch serves as a central point of connection for all devices and facilitates communication between them. In a star topology, each device has a dedicated connection to the hub or switch, which allows for individual communication between devices. When a device wants to send data, it sends the data to the hub or switch, which then forwards the data to the intended recipient. The advantages of the star topology include its simplicity, scalability, and fault tolerance. Because each device has a dedicated connection to the hub or switch, the topology is easy to set up and can be scaled easily by adding additional devices to the network. Additionally, if a single device or cable fails, the rest of the network remains unaffected, which makes the topology highly fault-tolerant. However, there are also some disadvantages to the star topology. One disadvantage is that the hub or switch can become a bottleneck for the network if it is not able to handle the amount of traffic being transmitted. Additionally, because each device has a dedicated connection to the hub or switch, the topology can require more cabling than other topologies, which can increase installation costs. Overall, the star topology is a versatile and reliable network topology that is well-suited for a wide range of applications. It is particularly useful for larger networks that require fault tolerance and scalability, and it is often used in enterprise and data center environments.

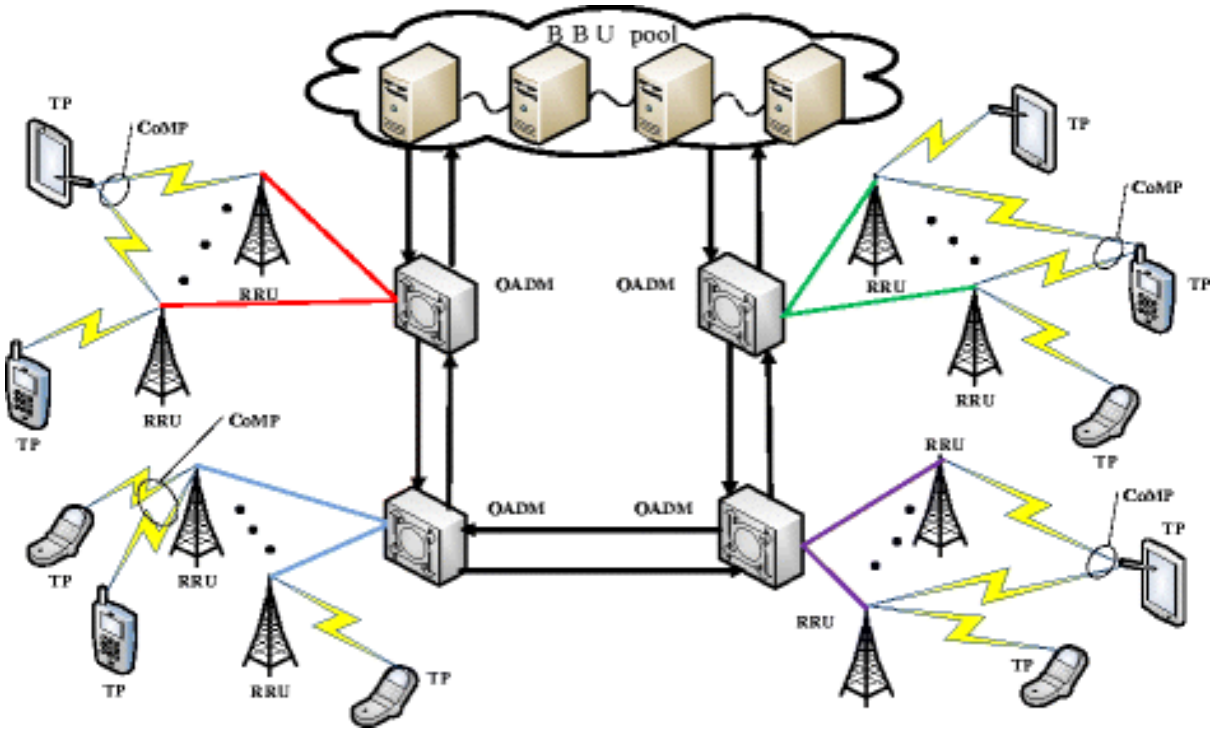


Figure II: Star Topology Network

- Ring topology (Figure III): In a ring topology, devices are connected in a closed loop, with data transmitted in one direction around the loop. Each device acts as a repeater, amplifying and forwarding data to the next device. In a ring topology, each device is connected to two neighboring devices, forming a continuous ring of devices. In a ring topology, data is transmitted around the ring in a unidirectional manner, with each device receiving data from its upstream neighbor and forwarding data to its downstream neighbor. When a device wants to transmit data, it sends the data onto the ring, where it is transmitted from device to device until it reaches its intended recipient. One advantage of the ring topology is that it is a highly fault-tolerant topology. Because the devices are connected in a continuous loop, if one device or cable fails, the network can still function by rerouting data through the remaining devices on the ring. Additionally, because data is transmitted in a unidirectional manner, the topology can provide a high level of performance and reduce the risk of collisions and other communication errors. However, there are also some disadvantages to the ring topology. One disadvantage is that the topology can be difficult to scale, particularly for larger networks. Additionally, the topology can be vulnerable to performance degradation if there are too many devices on the network, as each device must forward data to its downstream neighbor to transmit data around the ring. Overall, the ring topology is a reliable and fault-tolerant network topology that is particularly useful for applications that require high levels of performance and fault tolerance. However, it may not be the best choice for larger or more complex networks.

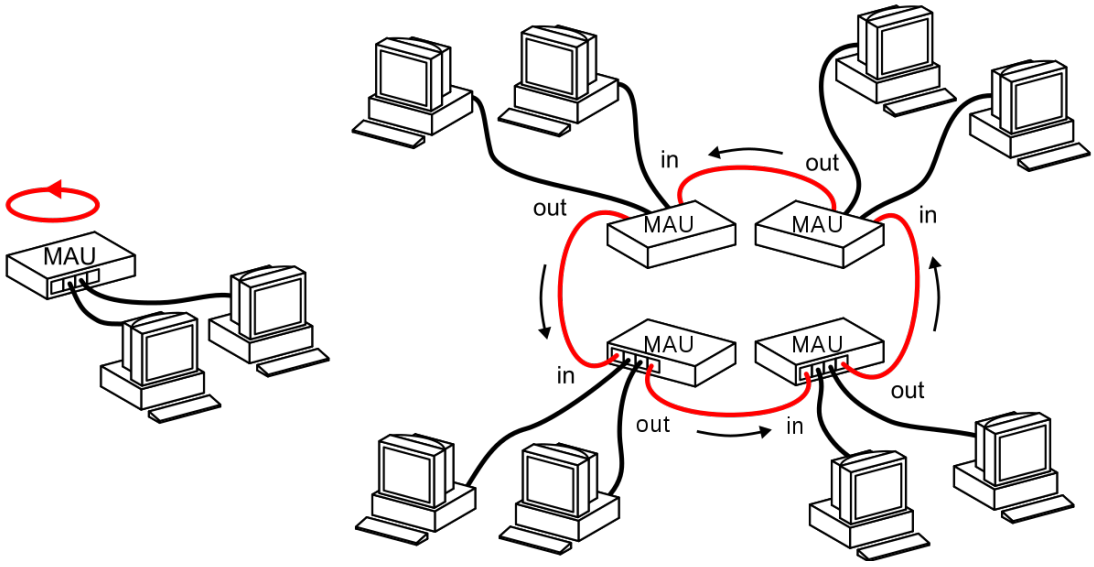


Figure III: Ring Topology Network

- Mesh topology (Figure IV): In a mesh topology, each device is connected to multiple other devices, creating multiple paths for data transmission. This can provide high fault tolerance and network redundancy, but can be complex to manage. The mesh network topology is a type of network topology in which all devices are connected to every other device in the network. In a mesh topology, data is transmitted between devices in a variety of different paths, depending on the specific routing algorithms used by the network. This provides a high degree of fault tolerance, as data can be rerouted around failed devices or connections. One advantage of the mesh topology is its high level of fault tolerance. Because every device is connected to multiple other devices, the topology can continue to function even if multiple devices or connections fail. Additionally, the topology can provide high levels of performance, particularly for applications that require a high degree of reliability or real-time communication. However, there are also some disadvantages to the mesh topology. One disadvantage is that the topology can be difficult to set up and manage, particularly for larger networks. Additionally, the topology can require many connections and devices, which can increase installation and maintenance costs.

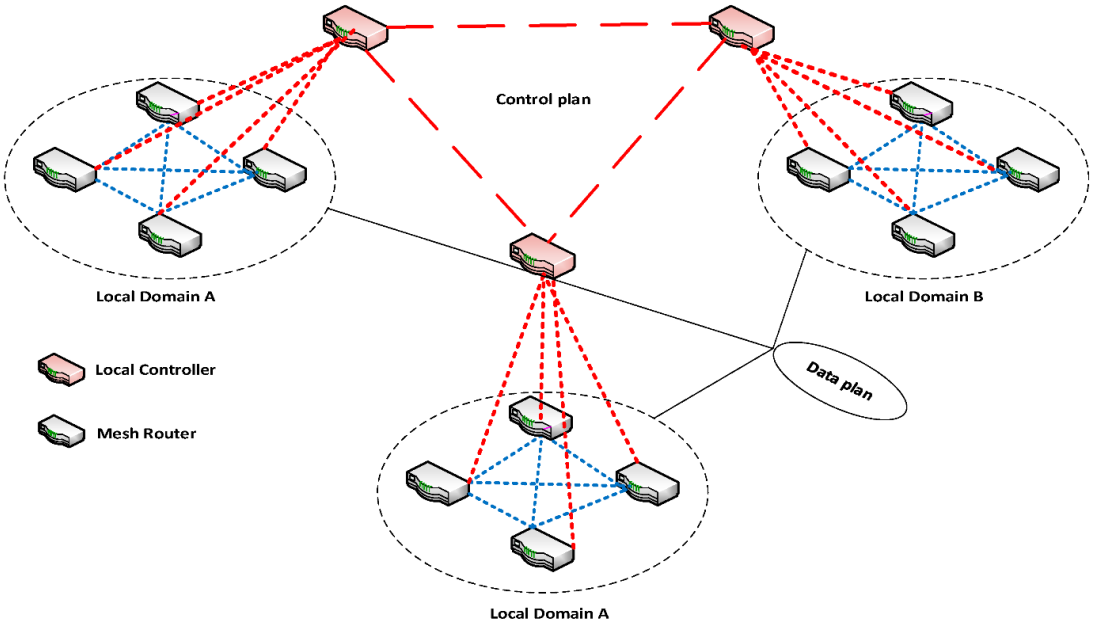


Figure IV: Mesh Topology Network

- Hybrid topology (Figure V): A hybrid topology is a combination of two or more topologies. For example, a network might combine a star topology with a bus topology, creating a hybrid star-bus topology. The goal of a hybrid topology is to combine the advantages of different topologies while minimizing their disadvantages. For example, a hybrid topology that combines the fault tolerance of a mesh topology with the simplicity of a star topology could provide a highly reliable and easy-to-manage network. One advantage of a hybrid topology is that it can provide a high degree of flexibility and customization. By combining different topologies, network designers can create networks that are optimized for their specific needs and requirements. Additionally, a hybrid topology can provide a high degree of fault tolerance, as multiple redundancy mechanisms can be implemented. However, there are also some disadvantages to a hybrid topology. One disadvantage is that the topology can be more complex to set up and manage, particularly for larger networks. Additionally, the topology can require a larger number of devices and connections, which can increase installation and maintenance costs.

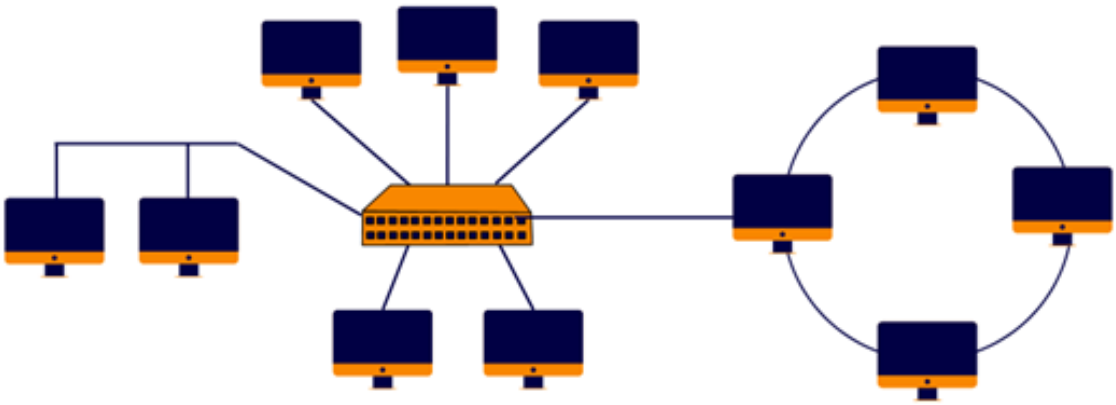


Figure V: Hybrid Topology Network

These are just a few examples of network topologies, and there are many other types that can be used depending on the specific needs of the network. The choice of topology depends on factors such as the number of devices, the geographic area of the network, and the required level of fault tolerance and performance.

The choice of network topology depends on the specific needs and requirements of the network. There is no one "most used" topology that applies universally to all networks.

That being said, some network topologies are more commonly used than others. For example, the star topology is a very popular choice for local area networks (LANs), as it provides a centralized point of control and is relatively easy to manage. The bus topology is also used in some LANs, as it is simple and inexpensive to set up, but it is less fault-tolerant than other topologies.

In wide area networks (WANs), mesh topologies are often used to provide high levels of fault tolerance and redundancy. However, mesh topologies can be complex and expensive to set up and maintain.

Ultimately, the choice of network topology depends on the specific needs of the network, and each topology has its own advantages and disadvantages. It is up to the network designer to choose the topology that best meets the requirements of the network.

All contribution of this thesis and methodologies are performed to be applicable on all above network topologies so we are not limited to any special cases.

Chapter 3: Anomaly Isolation Agents

Abstract

The Networks of connected devices are growing exponentially across the world and telecommunication operators are managing big, complex networks; thus, there is a need for intelligent and highly performing systems to support humans (network engineers) in network maintenance. Those devices, also called network elements or sensors, constantly report key performance indicators (KPIs) and utilize data on these combined with smart anomaly detection (AD) models to help prioritize production maintenance of the network. The main research gaps and challenges facing the most popular AD models, can be summarized as follows: a relatively long training time; unable to quantify the drivers of abnormal behavior when mining in a multidimensional space; unable to distinguish between anomalies and outliers that are not caused by system malfunctions; unable to prioritize detected anomalies by their severity level. To tackle these limitations, we propose a new geometrical multidimensional probabilistic model to search in the data space for abnormal behavior, generate anomaly scores and quantify anomaly drivers, which is a first and essential step to determining the root cause of detected anomalies. Furthermore, we propose a data-driven definition of an outlier score to be coupled with the anomaly score, to prioritize devices anomalies first and then tackle data observations that have a higher probability of being outliers. We also propose a sampling approach to speed up scoring newcomer observations, which gives our model the specificity of real-time intelligent systems. Our new model is tested on real-world data-set and compared with classical AD approaches. The results were reviewed by telecommunication network experts for validation, who concluded that our new approach works efficiently in detecting anomalies and identifying their drivers.

3.1 Introduction

An anomaly is an observation or event in a time series that has a small probability of occurring under normal circumstances. Consequently, the occurrence of anomalies is generally accompanied by symptoms that may disturb the operation of an underlying system. Anomalies are usually rare and prominent; they are experienced in various domains such as network traffic [21], fraud detection [6], medical and public health [22], industrial damage [2], image processing [17], etc. The detection of anomalies is well-developed in the literature, and several approaches and methods have been applied to detect such events. These works involve different areas of research such as machine learning [4], data mining [19], parametric/non-parametric statistics [25, 24], information theory [16], spectral theory [20], etc. However, the majority of these approaches and tools face challenges and have research gaps of different natures that are difficult to overcome [8, 10]. Though not exhaustive, some of the faced challenges when using the most popular AD algorithms are listed as follows: 1) Some of the approaches are more adapted to supervised learning where one has historical labels about anomalies. Unfortunately, most of the AD problems are based on unsupervised data where no labels about anomalies are available. Thus, it remains up to the user to confirm the performance of a model with a subject matter expert. 2) Many approaches, such as variational auto-encoders [14, 5], make assumptions about the prior probability distributions of the different features or the corresponding latent space (in the case of dimension reduction), which are not always reasonable to consider. 3) Almost all approaches provide a global score about the abnormality of a given observation without giving more information about which of the underlying features contributes the most to that observation being abnormal [7]. 4) Nearly all AD algorithms lack the ability to distinguish between real anomalies and outliers that are not caused by system malfunction [9], which may lead to critical false positive alarms.

A large number of studies have proposed models that help to overcome the first two challenges. The most popular unsupervised approaches are listed as follows: isolation forest [15, 23], deep auto-encoder [1, 11], and the density-based spatial clustering of applications with noise (DBSCAN) [13]. However, to the best of our knowledge, no anomaly detection algorithm can explicitly measure the contribution of each feature, observation by observation, or discriminate between real-abnormal behavior and outliers that are not caused by system malfunctions. Thus, these two challenges are considered the main research gaps in existing AD algorithms. Notably,

overcoming these gaps is the main motivation point of this chapter. The first contribution of this work is that we propose a new, unsupervised, and distribution-free anomaly detection algorithm that can be adapted to measure the contribution of each feature to the global anomaly score at the observation level. A second contribution is the ability of the proposed algorithm to discriminate between real anomalies and outliers, which is also a point that remains an open question in the AD field. For the first time, we present an algorithm that simultaneously overcomes the four challenges while showing significant computational advantages. In addition to the points, the proposed algorithm can prioritize detected anomalies and avoid considering all of them in the same basket, which is the case for classical AD models. Notably, the algorithm is built based on basic mathematical tools and objects that are easy to understand and manipulate without being obliged to create "black boxes" with a high level of complexity.

On the other hand, the AD problem that we are attempting to solve can be seen in the context of machine learning (ML) recognition problems [3]. The proposed algorithm can be assigned to such a class of models based on the following criteria: 1) The algorithm has the aim of recognizing future abnormal behavior based on a learning process from historical behavior in the dataset; 2) The proposed model transforms the AD process from manual work to an entirely automated procedure ready to be used in an online learning mode; 3) The computational complexity of the underlying learning process proposed by the algorithm was optimized through a sampling approach in order to have the capacity of implementing such a model on machines of reasonable computational power. Hence, based on all of the previous standards, this chapter aimed to develop an ML recognition model for anomaly detection.

In short, within the context of this work, an anomaly is an abnormal behavior in the data generated by a system when performing certain operations. Domain experts can add to data scientists' dimensions: "impact" and "significance" of an anomaly. The "impact" refers to how badly an abnormal behavior is affecting the underlying operations of a system. The "significance" refers to the level of impact of an anomaly on the underlying system, which can be measured by time (the longer, the worse), the impacted users or customers of a system, or the loss of opportunity for having the system streaming. One of the most important drawbacks we have observed in the existing unsupervised AD models is related to not considering domain experts' interests by investigating the impact and significance. For example, nearly all of the AD models treat data

outliers as abnormal behavior. However, such anomaly types are likely insignificant and have a low impact.

The main field of application of our method is detecting the abnormal behavior of different elements of a telecommunication network. Key performance indicators (KPIs) are generated by all of the elements of the network, and the number of such features describing the performance of the different services provided is massive, which makes the manual analysis of these observations extremely difficult or even impossible. In addition, detecting anomalies on the fly and without significant delays requires advanced correlation analysis and deep data mining on the generated data to identify hidden patterns and relationships.

The chapter is organized as follows: Section 3.2 introduces the mathematical formalization of the algorithm in a simple one-dimensional case. In Section 3.3, the hyper-cubic approach for high dimensions is introduced. Some applications on real-world data and comparisons with classical anomaly detection models are shown in Section 3.4. Finally, we conclude the chapter.

3.2 One-dimensional abnormality score

Let $\mathcal{I}$ be a finite set of indexes. In this work, $(\Omega, \mathcal{F}, \mathbb{P})$ is a probability space and for any $i \in \mathcal{I}$, X_i is a $\mathbb{R}$-valued random variable defined on Ω with a cumulative distribution function (CDF) $F_i(\cdot)$, and a probability density function $f_i(\cdot)$, with respect to the Lebesgue measure in $\mathbb{R}$.

These random variables X_i, $i \in \mathcal{I}$ have distinct distributions as they correspond to different KPIs. Without loss of generality, we start our formalization of an abnormality score by considering a simple case of one KPI represented by a real random variable X with a CDF $F(\cdot)$ and a pdf $f(\cdot)$. A realization x of X is said to be abnormal with an abnormality score $\theta > 0$ if for a very small, predefined value $\varepsilon > 0$:

$$\mathbb{P}[|X - x| < \theta] < \varepsilon.$$

A first challenge raise by the latter definition of an abnormality score is the difficulty of setting the value of θ and controlling it according to the behavior and the underlying distribution. To get past this problem, we propose the following method:
for $\varepsilon > 0$ (very small) and $\forall n \in \mathbb{N}^*$, let:

$$A_n = \left\{ x \in \Omega | \mathbb{P}\left[|X - x| < \frac{1}{n} \right] < \varepsilon \right\}.$$

By construction, the sequence of sets $(A_n)_n$ is increasing and converging to a set covering all Ω, i.e.,

$$A_1 \subseteq A_2 \subseteq \cdots \subseteq A_n \subseteq \cdots,$$

and,

$$\lim_{n \to +\infty} A_n = \Omega.$$

Note that the selection of the upper bound in the definition of the set A_n is not restricted to $\frac{1}{n}$. This upper bound can be substituted by any non-negative and decreasing function of n to verify the increasing nested property for the sequence $(A_n)_n$.

Then, $\forall x \in \Omega, \exists n(x) \in \mathbb{N}^*$ such that $x \in A_{n(x)}$ and $x \notin A_{n(x)-1}$. Therefore, $\theta(x) = \frac{1}{n(x)}$ represents a measure of abnormality (abnormality score) for x. In other words, if x_1 and x_2 are two realizations of the rv X, and $\theta(x_1) > \theta(x_2)$, then one can say that x_1 has a higher score with which to be considered as an abnormal observation. In other words, the pdf of the rv around this x shows much less density and the area under the pdf bounded by $x - \theta$ and $x + \theta$ is very small.

So, for a very small given value of ε, the rv $\theta(X)$ can be considered as a measure of abnormality for the realization of the underlying rv X. In addition, among all the possible forms of $\theta(\cdot)$, we must determine the one that is best overall at highlighting the abnormal behavior. In other words, the aim is to find the function $\theta^*(\cdot)$ verifying:

$$\theta^* = \underset{\theta \in \mathcal{F}_\theta}{\operatorname{argmax}} E[\theta(X)],$$

where $\mathcal{F}_\theta$ is the family of functions used as an upper bound in the definition of A_n to verify the increasing nested property for the sequence $(A_n)_n$.

Notably, to implement the above algorithm, a predefined small value of ε is fixed and the step function is considered to be $\theta = \frac{1}{n}$ with $n = 1, \ldots, N$, where N is the total number of observations in the dataset. Then, the probability defined within the set A_n is empirically computed by counting the number of observations among N that fall within the corresponding interval. If, for an observation x, the condition $x \in A_{n(x)}\ and\ x \notin A_{n(x)-1}$ is not verified for all of the considered values of n, then we consider that the observation is in a dense area and cannot be considered abnormal.

3.3 Multi-dimensional abnormality hyper-cubic (AHCA):

3.3.1 Abnormality score

Now, in the same context, we suppose that we are dealing with $m = Card(\mathcal{I})$ KPIs modeled by m real valued rv $X_i, i = 1, \dots, m$. In a similar way, a realization $x = (x_1, x_2, \dots, x_m)$ of $X = (X_1, X_2, \dots, X_m)$ is said to be abnormal with an abnormality score vector $\theta = (\theta_1, \theta_2, \dots, \theta_m)$, with $\theta_i > 0 \; \forall i \in \{1, 2, \dots, m\}$, if for a very small, predefined, value $\varepsilon > 0$:

$$\mathbb{P}[|X_1 - x_1| < \theta_1, |X_2 - x_2| < \theta_2, \dots, |X_m - x_m| < \theta_m] < \varepsilon. \quad (1)$$

Accordingly, for $\varepsilon > 0$ (very small) and $\forall n = (n_1, n_2, \dots, n_m) \in \mathbb{N}^{m*}$, let:

$$A_n = \left\{ x \in \Omega^m | \mathbb{P}\left[|X_1 - x_1| < \frac{1}{n_1}, |X_2 - x_2| < \frac{1}{n_2}, \dots, |X_m - x_m| < \frac{1}{n_m} \right] < \varepsilon \right\}.$$

Note that sequence A_n has the same properties as in the univariate case by considering $n + 1 = (n_1 + 1, n_2 + 1, \dots, n_m + 1)$. In addition, $\forall \; x = (x_1, x_2, \dots, x_m) \in \Omega^m$, $\exists n(x) = \big(n_1(x), n_2(x), \dots, n_m(x)\big) \in \mathbb{N}^{m*}$ such that $x \in A_{n(x)} \; and \; x \notin A_{n(x)-1}$. So, in this case, the volume of the hyper-cube of m edges of length $2 \times \theta_1(x) = \frac{2}{n_1(x)}$, $2 \times \theta_2(x) = \frac{2}{n_2(x)}, \dots, 2 \times \theta_m(x) = \frac{2}{n_m(x)}$, respectively, given by $\theta(x) = 2^m \prod_{i=1}^{m} \theta_i(x)$, represents an abnormality score for x.

The practical implementation of the AHCA scoring system is the same as the one described at the end of Section 3.3, with an adaptation to the multi-dimensional context.

3.3.2 Individual abnormality contribution

To measure the contribution of each of the variables in a multi-dimensional context, one should be able to assess the individual effect of an rv $X_i, i = 1, \dots, m$ in making a realization $x = (x_1, x_2, \dots, x_m)$ more or less abnormal. In addition, while assessing the individual effect of each X_i, one should consider the interaction with the other underlying variables, given that the assumption of a mutually independent rv is not always reasonable.

To achieve such a target, let us recall that $\forall \; x = (x_1, x_2, \dots, x_m) \in \Omega^m$, one can compute an abnormality score $\theta(x) = 2^m \prod_{i=1}^{m} \theta_i(x)$ based on the method described in Subsection 3.3.1. Then, by considering each of the $\theta_i(x), i = 1, \dots, m$, one may compute the contribution of each variable X_i by calculating the following quantity:

$$Imp_i(x) = 1 - \mathbb{P}[|X_i - x_i| < \theta_i(x)]. \quad (2)$$

A variable X_i less impacts the abnormality score of an observation x when the component x_i of x is in a dense area of the probability density function

of the rv X_i, e.g., the probability $\mathbb{P}[|X_i - x_i| < \theta_i(x)]$ is high, and consequently, $Imp_i(x)$ is low and closer to zero. Then, by sorting the elements of the vector $Imp\,(x) = \left(Imp_i(x)\right)_{i=1,\dots,m}$ in a decreasing order, one will be highlighting the impact of each variable from the highest to the lowest on the abnormal behavior of x.

Now, in the next subsection, we will suggest a method based on Shannon's Entropy [18] and use the probability distribution of $Imp\,(x)$ to help the user decide if an observation x, with a high anomaly score, is really an anomaly or can be considered as an outlier. Such a differentiation method, to the best of the authors' knowledge, has not been presented in the literature before.

From an implementation perspective, one should start by computing the abnormality score as described in Subsection 3.3.1. Next, each component $\theta_i(x)$ is used to compute in an empirical manner (as described in the one-dimensional case in Section 3.2) $\mathbb{P}[|X_i - x_i| < \theta_i(x)]$, leading to $Imp_i(x)$.

3.3.3 Abnormality vs. Outliers

Assuming that a given observation $x = (x_1, x_2, \dots, x_m) \in \Omega^m$ has a high abnormality score $\theta\,(x)$ and is likely to be an anomaly with a vector of feature importance $Imp\,(x) = \left(Imp_i(x)\right)_{i=1,\dots,m}$. To decide if x represents more an outlier than a behavioral abnormality, one needs to extract information about how much the distribution of importance is close to a uniform distribution. When it is reasonable to say that the abnormality contribution of the underlying variables is uniformly distributed, one can assume that such an observation is more likely to be an anomaly. Otherwise, when some variables contribute significantly more to the abnormal behavior, one can say that it is more an outlier observation than an anomaly. In summary, we are proposing a new way to measure the likelihood of having an outlier versus an anomaly under the following assumption: if only one rv is showing abnormal behavior, then there is a high probability of an outlier. On the other hand, if all variables are showing abnormal behavior, this is a sign that the probability of having an outlier is low. To do this, we start by transforming the vector of importance into a probability distribution by applying the following transformation:

$$Imp(x)^* = \left(\frac{Imp_i(x)}{\sum_{j=1}^{m} Imp_j(x)}\right)_{i=1,\dots,m}.$$

Next, we compute Shannon's entropy of the discrete probability distribution defined by $Imp(x)^*$:

$$\mathcal{H}(Imp(x)^*) = -\sum_{i=1}^{m} \frac{Imp_i(x)}{\sum_{j=1}^{m} Imp_j(x)} log_2 \frac{Imp_i(x)}{\sum_{j=1}^{m} Imp_j(x)}.$$

Now, one can use $\mathcal{H}(Imp(x)^*)$ as a goodness of fit score for a uniform distribution. In other words, when $\mathcal{H}(Imp(x)^*)$ is closer to $log_2 m$ than to zero, we can say that x is more likely to be an anomaly than an outlier, as in this case, the contribution is more uniformly distributed across all the underlying KPIs instead of being concentrated and dominated by a few KPIs. Note that in our context, an outlier is an observation that is abnormal for a reason other than a malfunction of an element (e.g., a cell) in a network. To provide an example, a bad experimental measurement can be the cause of an outlier. Based on this new ranking of observations, users of this algorithm output can now classify occurrences of these random variables in a 2D space using the score of abnormality and the score of being an outlier. Then, domain experts can start investigating cases that are most likely anomalies and less likely outliers; the latter should be treated by other teams such as data quality teams.

Then, in summary, from technical and methodological perspectives, the novel elements of AHCA with respect to state-of-the art AD algorithms are presented as follows: 1) Through the combination of the sequence of sets A_n and the corresponding empirical probability, AHCA is the first AD algorithm to consider a probabilistic approach applied to a particular geometric topology to compute an anomaly score for each data point. 2) AHCA has the ability to show the importance of each underlying variable in the global abnormality score of an observation through the computation of the function $Imp\,(x)$, which presents the importance of the variables while preserving the correlations between them. 3) Finally, by applying Shannon's entropy on the vector $Imp(x)^*$ showing the importance probability distribution, AHCA has the novelty of discriminating between real-anomalies and outliers.

3.4 Application

3.4.1 Data description

Without loss of generality and as an application of our approach, we consider the data representing the observations of two particular KPIs describing the performance of one cell on a given network. These KPIs were selected by SMEs as the most informative drivers of abnormal behavior in a telecommunication network. Data were collected from a virtual telecommunication network during a period of time covering multiple traffic scenarios and congestion. Key performance indicators (KPIs) measure several performance aspects of cells like call success rates, handover success rates and other confidential KPIs. Data are aggregated by n minutes by cell and we assume that every time interval represents an occurrence of considered random variables. In this context, abnormal behaviors are described and characterized as follows: Anomalies are important events for network operation center (NOC) teams within operators since they indicate critical network failures in hardware or software that can lead to impacts on business, lost opportunities for service usage, and customer churn.

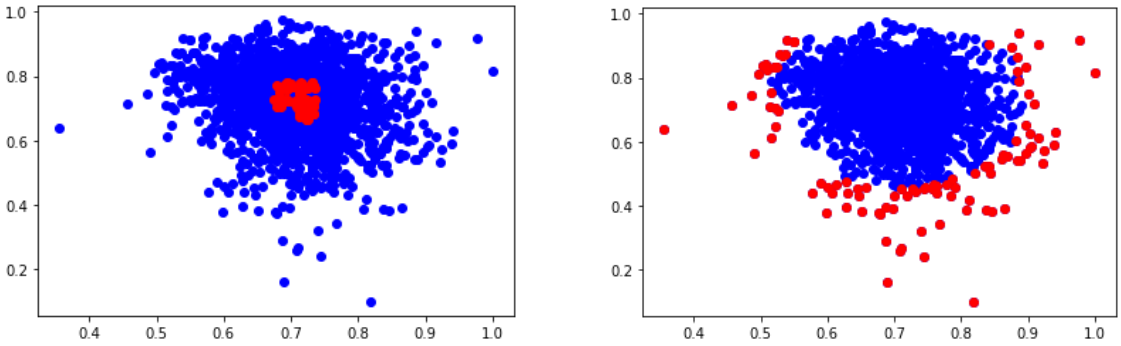
Anomalies in part of a network might be small operation issues or major failures, and their consequences can be very costly for network operators since the impact of these anomalies can result in the total disruption of the network. As such, customers will be heavily impacted and may look at changing their network operators. Therefore, NOC teams must heavily prioritize seeking and rapidly detecting (proactively if possible) any anomalies and fixing them as soon as possible. In the next section the results of AHCA will be compared to two of the most popular anomaly detection models: Isolation Forest and Auto-Encoder.

3.4.2 Results and analysis

3.4.2.1 Comparative analysis with classical AD models

This section aims to compare the anomaly scores generated by AHCA, after being implemented in Python, with other popular anomaly detection algorithms, and interpret and analyze the results with the help of telecommunication subject matter experts (SMEs). Note that the values of the KPIs were scaled in a way to get values between 0 and 1 before applying the algorithms. Scaling the features is important in our context as we are comparing measurements that have different units and are measured at different scales. Then, KPIs do not contribute equally to the analysis, whereas without scaling, they might end up creating a bias.

First, the isolation forest anomaly detection model is applied. Figure I shows this in red. Notably, for Figures 1-5 x – axis and y – axis designate the scaled values of the first and second considered KPIs, respectively.



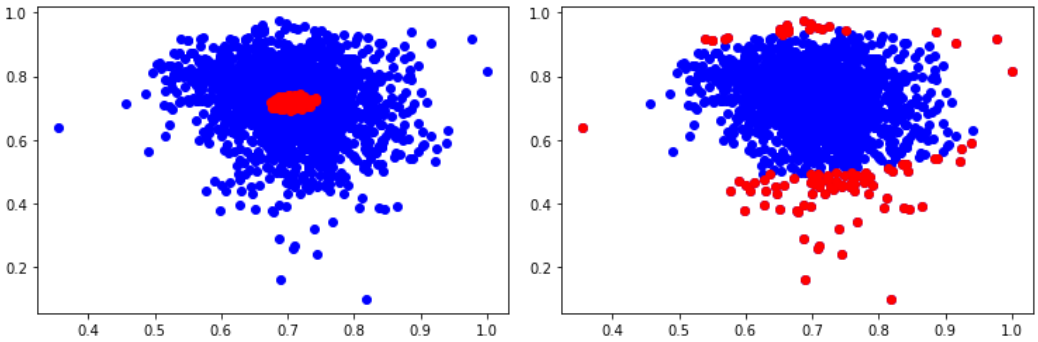
(a)100 most non-anomalous observations

(b) 100 most anomalous observations

Figure I: Anomaly scores based on Isolation Forest algorithm

It's clear that the non-anomalous observations are concentrated in the center of the region representing the interaction between the considered KPIs. Anomalous observations are located on the edge of the plot, except for the upper part of the plot.

Second, the auto-encoder (AE) anomaly detection model is applied. The results are represented in Figure II.



(a)100 most non-anomalous observations

(b) 100 most anomalous observations

Figure II: Anomaly scores based on AE algorithm

In our context, the results of the AE are not stable as the performance and the robustness of the algorithm depend on controlling a long list of hyperparameters with a risk of over-fitting. In addition to that, it is well known that deep machine-learning algorithms come with a high cost in terms of computational time. Then, this anomaly detection approach is not really adapted to our context where a fast and stable algorithm, performing in an online manner, is needed. In summary, AE was directly rejected by the SMEs.

Finally, the results of AHCA, with a fixed $\varepsilon = 0.001$, are represented in Figure III, where one can see that the anomalies/outliers cover all the border of the scatter plot (191 observations out of 1649) without keeping any of the borderline points beyond suspicion, which was the case in the two previous models.

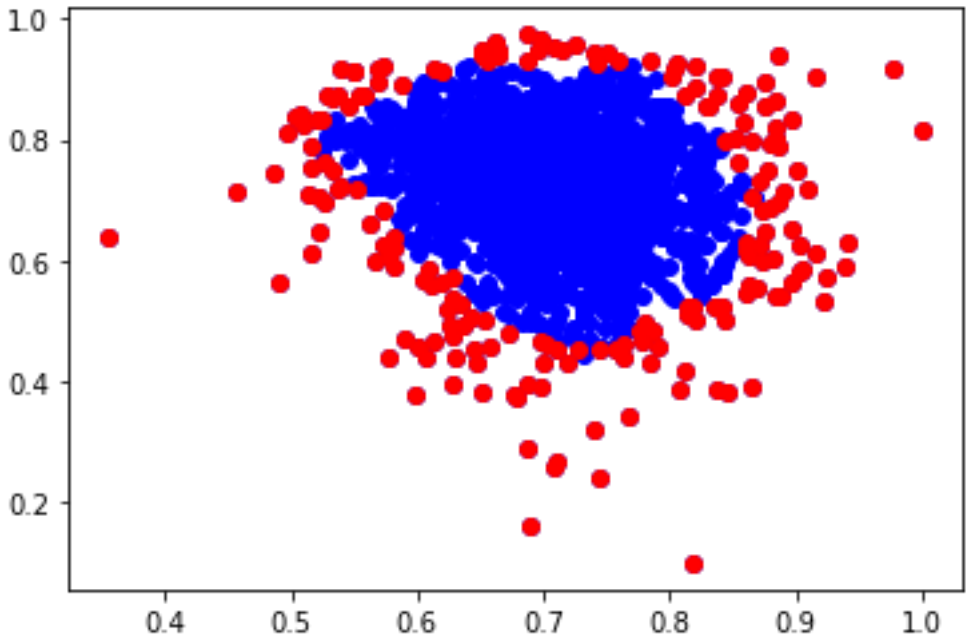


Figure III: Anomalous observations based on AHCA algorithm

Then, a question arises here, among all these observations considered as anomalies, which are real anomalies and which are more outliers?

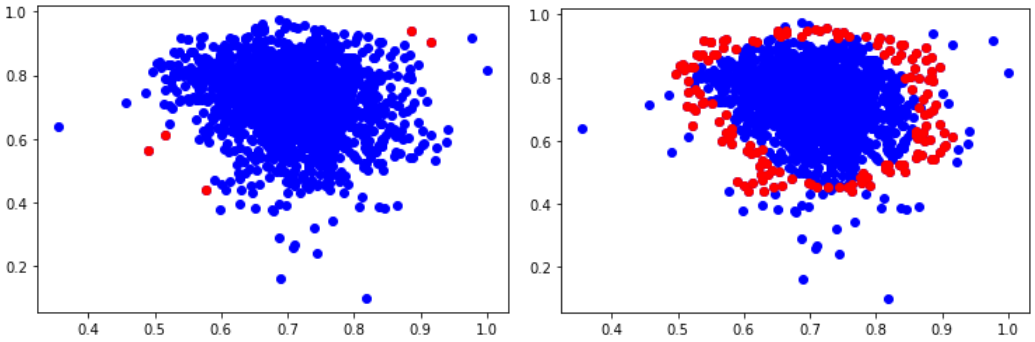
To answer this question, one may require the help of SMEs to carry out a "manual analysis" of these observations and classify them. Another solution, completely automated and less time consuming, is to apply the approach described in Subsection 3.3.3. Then, the two-level discrimination of our dataset in normal/abnormal behavior and anomaly/outlier classification generates four possible scenarios that will be presented in the next subsection.

3.4.2.2 Beyond AD: Innovative AHCA results

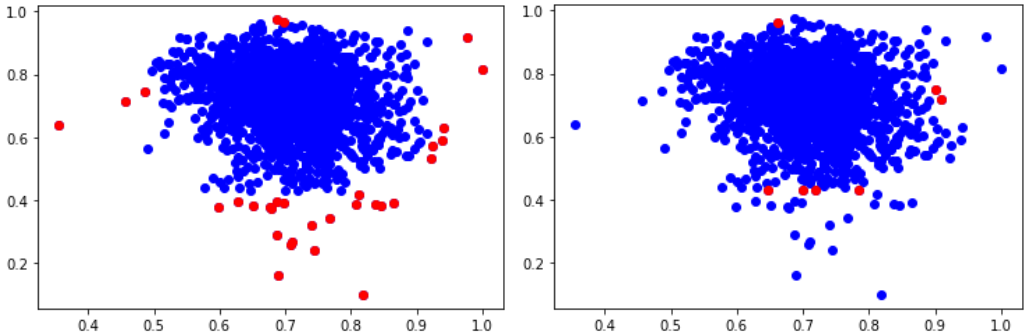
In this subsection, the aim is to go beyond anomaly detection and show the ability of the proposed algorithm to provide results for both anomaly vs outlier discrimination and variable importance analysis.

As a starting point, Figure IV shows four cases that should be considered by the expert: 1) High priority cases, showing the observations with high abnormality scores (above the empirical average) and a low probability of being outliers (below the empirical average), Figure IVa; 2) Medium-I priority cases, i.e., observations with high abnormality scores and a high probability of being outliers, Figure IVb; 3) Medium-II priority cases, i.e., observations with low abnormality scores and a low probability of being outliers, Figure IVc; and 4) Low priority cases, i.e., observations with low abnormality scores and a high probability of being outliers, Figure IVd. One can remark that the number of high priority cases among the observations

that are originally considered as anomalies is small, which is very reasonable in practice and a result commonly encountered by SMEs.



(a) Anomalies with High Priority (b) Anomalies with Medium-I Priority

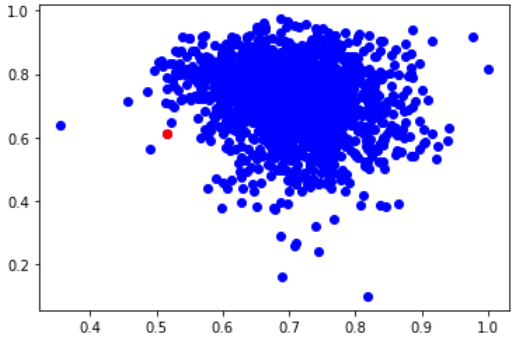


(a) Anomalies with Medium-II Priority (b) Anomalies with Low Priority

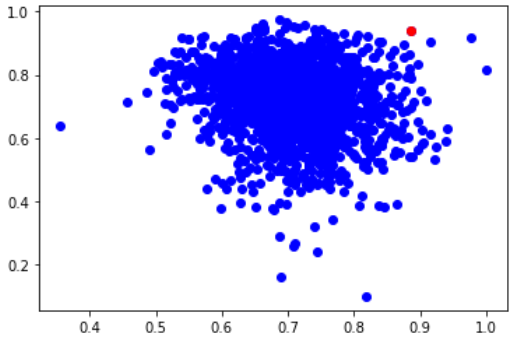
Figure IV: Anomalies vs. outliers in different scenarios

On the other hand, among the observations that were selected as high priority, we select some points to highlight the contribution of each of the underlying KPIs. The first observation in red in Figure Va shows that the anomaly is more due to the first KPI (on x-axis) then the second one (on y-axis). The projection of the data points on each axis and the analysis of the probability density of each KPI around the selected point, based on the method described in Subsection 3.3.2, show that the contribution of the first KPI has a proportion of 65.4% and 34.6% for the second one. A second example is shown if Figure Vb; by applying the same analysis as for the first point we get a contribution of 52.6% for the first KPI and 47.4% for the second. So, in the second case, both KPIs almost equally contribute to the detected abnormality.

Based on all these previous analyses, it is important to mention that AHCA can be seen also as a clustering algorithm that can be used not just for online AD but to classify a dataset into five different clusters, four for abnormal behaviors with different priority levels and a fifth one for normal observations.



(a) Anomaly with unequal underlying contributions



(b) Anomaly with equal underlying contributions

Figure V: Importance and contribution analysis of particular anomalies

Based on all of these comparisons and analyses, Table I presents the capacity of the different algorithms to overcome the list of challenges listed in the Introduction. These challenges will be labeled in Table I as follows: "Challenge 1" for the model is adapted to unsupervised learning; "Challenge 2" for the algorithm is distribution-free; "Challenge 3" for the model provides information about feature importance at the observation level; "Challenge 4" for the model has the ability to distinguish between real anomalies and outliers. AHCA overcomes all of the classical challenges of AD models, with the additional power of classifying the detected anomalies into four priorities instead of placing all anomalies into one basket, which is the case for classical AD models.

Models	Challenge 1	Challenge 2	Challenge 3	Challenge 4
Random Forest	X	X		
Auto-Encoder	X	X		
AHCA	X	X	X	X

Table I: Comparison of different algorithms on the capacity to overcome challenges

3.4.3 Sampling method

Notably, AHCA was designed in a way (see Section 3.3 for more details) to come complete with a computational complexity of order less than $O(N^2 \times m)$ where N is the total number of observations already existing in the considered dataset and m is the number of underlying variables i.e., dimension of the space. In fact, to compute the abnormality score of any new upcoming observation, one is not obliged to go through all the possible values of θ that are N. One stops at the largest value of θ verifying Equation (1).

To reduce the computational complexity of the algorithm, especially in case of a high-dimensional dataset, we propose a sampling method adapted to the context of anomaly detection. Among all the observations $x = (x_1, x_2, \dots, x_m)$, where m is the number of variables, we start by selecting those verifying the following property:

$$\exists j \in \{1, \dots, m\} \text{ such that } x_j \leq \mu_{X_j} - 1.5\sigma_{X_j} \text{ or } x_j \geq \mu_{X_j} + 1.5\sigma_{X_j},$$

with μ_{X_j} and σ_{X_j} respectively representing the empirical mean and standard deviation of the variable X_j. We denote the set of selected observations by S and $card(S) = N_1 \ll N$. In a second step, and from the remaining observations, we select kN_1 observations from the kernel region of the data, i.e., verifying:

$$\forall j \in \{1, \dots, m\} \text{ such that } \mu_{X_j} - 1.5\sigma_{X_j} \leq x_j \leq \mu_{X_j} + 1.5\sigma_{X_j}.$$

This second set of observations is denoted by S' and $card(S') = kN_1$. Then, the final sample that should be considered to apply the algorithm is $S \cup S'$ with $card(S \cup S') = (1 + k)N_1$. Note that to fix the parameter k one can work on minimizing the probability of having anomalies, detected by AHCA, among the observations of the sample S'.

Hence, at the arrival of a new observation, and to compute the abnormality score of this new arrival, instead of applying the AHCA on all the observed instances, one can focus only on the ones belonging to $S \cup S'$. Thus, by applying this sampling method, the computational complexity of AHCA is significantly reduced to reach an order of less than $O\left(\left(\left(1 + k\right)N_1\right)^2 \times m\right)$. The application of this sampling permit to explore the AHCA has the advantages of working in a much faster way while preserving the important information in the dataset, which is linking anomaly detection to the extreme value theory.

As an application of this sampling technique, we performed an anomaly detection analysis based on three selected KPIs and the results are given in the appendix of this chapter. Notably, based on all the previous application subsections, one can confirm that AHCA verifies all the criteria listed in the Introduction to be considered an ML recognition model for AD. In fact, AHCA is fully automated with an online learning mode that can classify

observations into four different classes of abnormal behavior and optimized on computational complexity level. In addition, this clustering capacity of AHCA is complete with the ability to identify the importance of the different underlying variables contributing to the final clustering representation.

3.4.4 Advantages of the proposed algorithm

Compared to classical anomaly detection models, the AHCA algorithm has several advantages:

- No need to train a model on part of the data and test it on the remaining part. Instead, it is already an online learning process where each new arrival is tested for abnormal behavior without a risk of over-fitting and in the minimal time.

- AHCA gives the user the capability of measuring the contribution of each variable of the decision space toward making a given observation more or less abnormal. This is, to the best of the authors' knowledge, the first time that an algorithm has been able to prove the abnormality contribution of each variable on an observation level and not the overall feature importance, as for example, in the random forest approach. Quantifying these contributions can be seen as a starting point for a root cause analysis of an identified anomaly.

- AHCA is distribution-free in the sense that we do not have to make prior assumptions about the underlying probability distribution of the different variables. All the probabilities can be computed in an empirical data driven way.

- There are no assumptions about the interaction and the relationship between the variables of our decision space. So, the algorithm may fit any linear or non-linear decision space structure.

- AHCA has only one hyper-parameter to control, which is ε. Therefore, there is no need for complex hyper-parameter optimization techniques, meaning we are dealing with an algorithm with a low level of complexity.

- Unique among anomaly detection algorithms, AHCA helps the user discriminate between an abnormal behavior in a decision space and a probable outlier.

3.5 Conclusion

In this work, an anomaly detection algorithm, based on an approach coupling probabilistic and geometric principles, has been presented and shown to be more advantageous than other classical approaches on several levels. Without being exhaustive, the abnormality hyper-cubic approach is distribution-free and hyper-parameter-free; optimized in terms of time consumption; deals with data structures of high complexity (e.g., linear and non-linear correlations); well adapted to avoid the over-fitting phenomenon; has the ability to assess the abnormality contribution of each variable toward each observation and discriminate between abnormal behavior and outliers.

Moreover, we applied the algorithm to real data in the field of telecommunication and the results were compared to other classical anomaly detection approaches. As the approach is unsupervised, the results were validated by experts who confirmed that AHCA outperforms other approaches.

The authors intend to use the results of AHCA and try to develop a tool to help users with root cause analysis and even propose a solution to fix the anomaly. A second perspective is to study the exact probabilistic behavior of anomalies, based on extreme value theory-in particular, records theory-by drawing inspiration from the work of Hoayek and Ducharme [12]. A third perspective would involve investigating the option of graph representation and classification for AD as a potential generalization of AHCA so that it could achieve the following: 1) Consider data of different typologies (e.g., categorical or textual) that can be represented in a graphical format; 2) Use AHCA to classify a whole time series as abnormal or not, which is an important scientific gap in the field of existing AD models that may also help in the detection of abnormal elements (e.g., cells or stations) in the network instead of only detecting an abnormal observation defined at a small-time interval.

Appendix:

We also applied AHCA to the same dataset by considering three KPIs proposed by the SMEs, and we classified our detected anomalies-shown in red (with an $\varepsilon = 0.01$) in Figure VI-into four priority classes. The details of the different scenarios are shown in Figure VII. Notably, for Figures VI and VII, the $x-$ axis, $y-$ axis, and $z-$ axis designate the scaled values of the first, second, and third considered KPIs, respectively.

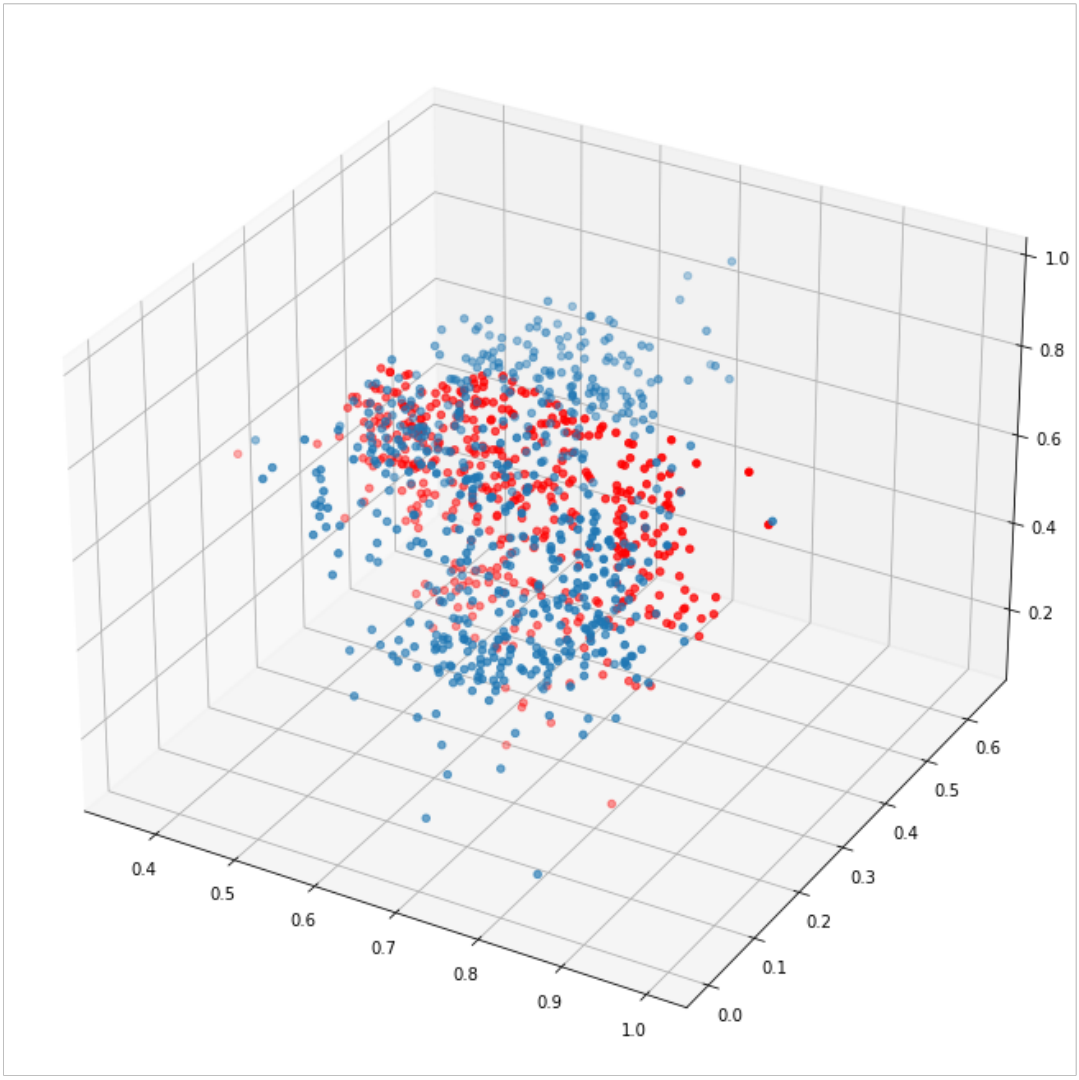
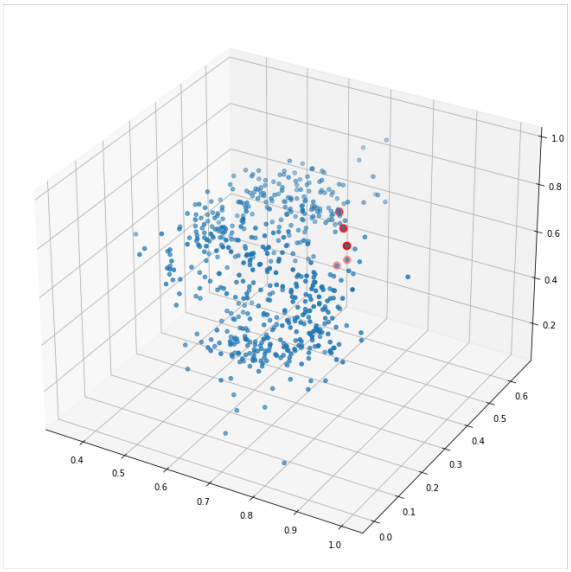
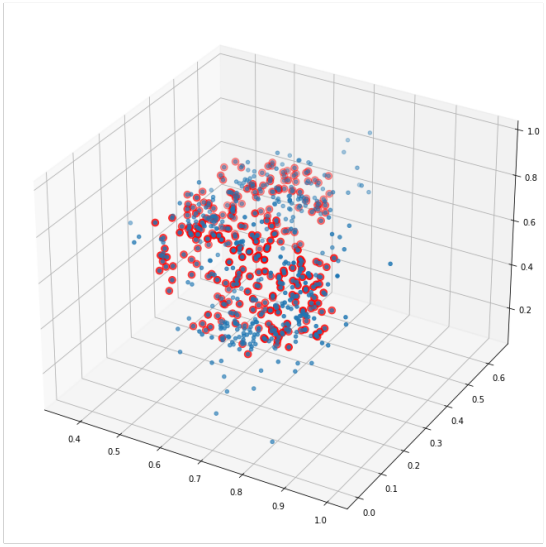


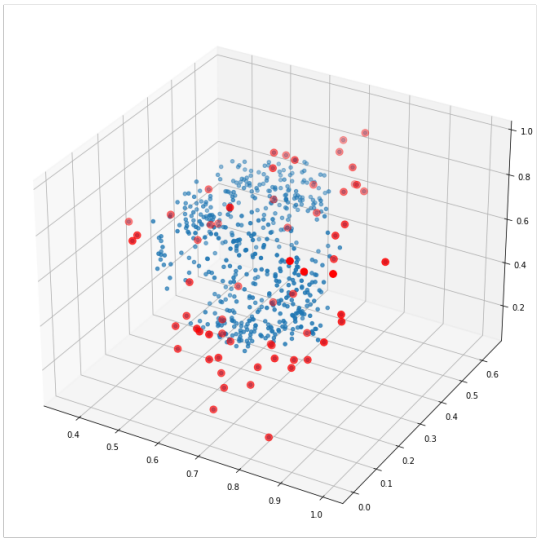
Figure VI: Anomalous observations based on AHCA algorithm



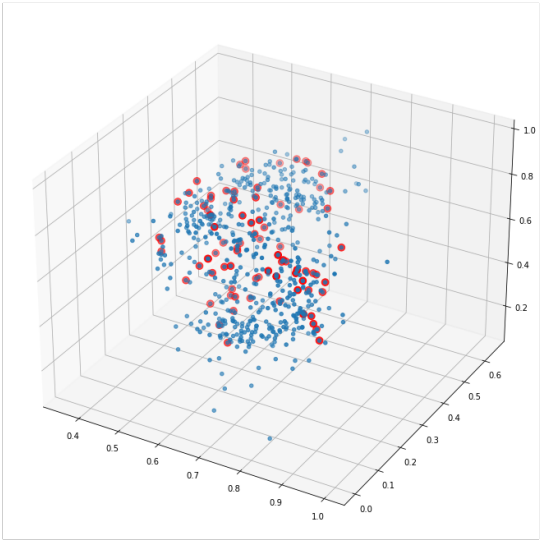
(a) Anomalies with High Priority



(b) Anomalies with Medium-I Priority



(a) Anomalies with Medium-II Priority



(b) Anomalies with Low Priority

Figure VII: Anomalies vs. outliers in different scenarios

References

[1] An, J., and Cho, S. Variational autoencoder based anomaly detection using reconstruction probability. Special Lecture on IE 2, 1 (2015), 1-18.

[2] Basu, S., and Meckesheimer, M. Automatic outlier detection for time series: an application to sensor data. Knowledge and Information Systems 11, 2 (2007), 137-154.

[3] Chakraborty, S., Tomsett, R., Raghavendra, R., Harborne, D., Alzantot, M., Cerutti, F., Srivastava, M., Preece, A., Julier, S., Rao, R. M., et al. Interpretability of deep learning models: A survey of results. In 2017 IEEE smartworld, ubiquitous intelligence & computing, advanced & trusted computed, scalable computing & communications, cloud & big data computing, Internet of people and smart city innovation (smartworld/ SCALCOM/UIC/ATC/CBDcom/IOP/SCI) (2017), IEEE, pp. 1-6.

[4] Chan, P. K., Mahoney, M. V., and Arshad, M. H. A machine learning approach to anomaly detection. Tech. rep., 2003.

[5] Fahrmann, D., Damer, N., Kirchbuchner, F., and Kuijper, A. Lightweight long short-term memory variational auto-encoder for multivariate time series anomaly detection in industrial control systems. Sensors 22, 8 (2022), 2886.

[6] Fawcett, T. E., and Provost, F. Fraud detection. In Handbook of data mining and knowledge discovery. 2002, pp. 726-731.

[7] Feng, Y., Cai, W., Yue, H., Xu, J., Lin, Y., Chen, J., and Hu, Z. An improved x-means and isolation forest based methodology for network traffic anomaly detection. Plos one 17, 1 (2022), e0263423.

[8] Fernandes, G., Rodrigues, J. J., Carvalho, L. F., Al-Muhtadi, J. F., and Proenc, a, M. L. A comprehensive survey on network anomaly detection. Telecommunication Systems 70 (2019), 447-489.

[9] Gaddam, A., Wilkin, T., Angelova, M., and Gaddam, J. Detecting sensor faults, anomalies and outliers in the internet of things: A survey on the challenges and solutions. Electronics 9, 3 (2020), 511.

[10] Habeeb, R. A. A., Nasaruddin, F., Gani, A., Hashem, I. A. T., Ahmed, E., and Imran, M. Real-time big data processing for anomaly detection: A survey. International Journal of Information Management 45 470 (2019), 289-307.

[11] Han, J., Liu, T., Ma, J., Zhou, Y., Zeng, X., and Xu, Y. Anomaly detection and early warning model for latency in private 5g networks. Applied Sciences 12, 23 (2022), 12472.

[12] Hoayek, A. S., Ducharme, G. R., and Khraibani, Z. Distribution free inference in record series. Extremes 20, 3 (2017), 585.

[13] Jain, P. K., Bajpai, M. S., and Pamula, R. A modified dbscan algorithm for anomaly detection in time-series data with seasonality. Int. Arab J. Inf. Technol. 19, 1 (2022), 23-28.

[14] Kingma, D. P., and Welling, M. Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013).

[15] Liu, F. T., Ting, K. M., and Zhou, Z.-H. Isolation forest. In 2008 eighth ieee international conference on data mining (2008), IEEE, pp. 413-422.

[16] Noble, C. C., and Cook, D. J. Graph-based anomaly detection. In Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining (2003), pp. 631-636.

[17] Pokrajac, D., Lazarevic, A., and Latecki, L. J. Incremental local outlier detection for data streams. In 2007 IEEE symposium on computational intelligence and data mining (2007), IEEE, pp. 504-515.

[18] Shannon, C. E. A mathematical theory of communication. The Bell system technical journal 27, 3 (1948), 379-423.

[19] Silva, N., Soares, J., Shah, V., Santos, M. Y., and Rodrigues, H. Anomaly detection in roads with a data mining approach. Procedia computer science 121 (2017), 415-422.

[20] Sun, J., Xie, Y., Zhang, H., and Faloutsos, C. Less is more: Compact matrix representation of large sparse graphs. In Proceedings of 7th SIAM International Conference on Data Mining (2007).

[21] Vikram, A., et al. Anomaly detection in network traffic using unsupervised machine learning approach. In 2020 5th International Conference on Communication and Electronics Systems (ICCES) (2020), IEEE, pp. 476-479.

[22] Wong, W.-K., Moore, A. W., Cooper, G. F., and Wagner, M. M. Bayesian network anomaly pattern detection for disease outbreaks. In Proceedings of the 20th International Conference on Machine Learning (ICML- 03) (2003), pp. 808-815.

[23] Xu, H., Pang, G., Wang, Y., and Wang, Y. Deep isolation forest for anomaly detection. arXiv preprint arXiv:2206.06602 (2022).

[24] Ye, N., and Chen, Q. An anomaly detection technique based on a chisquare statistic for detecting intrusions into information systems. Quality and reliability engineering international 17, 2 (2001), 105-112.

[25] Yeung, D.-Y., and Chow, C. Parzen-window network intrusion detectors. In Object recognition supported by user interaction for service robots (2002), vol. 4, IEEE, pp. 385-388.

Chapter 4: Combining Numeric KPI and Categorical Alarm Data

In the previous chapter, we have delved into the development of a novel algorithm for detecting anomalies based on numeric Key Performance Indicator (KPI) data, providing valuable insights into network performance and health. While this approach effectively utilizes numeric data, it does not consider the categorical data found in alarms, which also play a vital role in understanding network anomalies. In the following chapter, we will explore an anomaly detection approach that leverages alarm data. However, before we proceed, it is essential to establish a link between the two topics and explore the benefits of combining both numeric and categorical data for a more comprehensive analysis of telecommunication networks.

The Complementary Nature of Numeric KPIs and Categorical Alarms

Although numeric KPIs and categorical alarms represent different types of data, they can be seen as complementary in the context of network anomaly detection. Numeric KPIs provide quantitative information on network performance and help operators assess service quality and efficiency. Meanwhile, categorical alarms offer qualitative insights into the network's current state, highlighting potential issues or failures that could disrupt network operations.

Integrating these two data types allows operators to obtain a holistic view of network health, which can lead to improved decision-making and more effective anomaly detection. For instance, an unexpected spike in dropped call rate (a numeric KPI) might be better understood when correlated with a recent alarm indicating a faulty base station. By examining both data types simultaneously, network operators can identify the root cause of an issue more quickly and efficiently.

Challenges in Combining Numeric KPIs and Categorical Alarms

The integration of numeric KPIs and categorical alarms for anomaly detection comes with its own set of challenges. These challenges primarily stem from the inherent differences between the two data types and include:

A. Data Preprocessing: Numeric KPIs and categorical alarms require different preprocessing techniques. While KPI data often needs normalization or scaling, alarm data needs to be encoded to be effectively used in machine learning algorithms.
B. Feature Selection and Engineering: Combining both data types require careful consideration of relevant features to be used for anomaly detection. This involves selecting the most relevant KPIs and alarm attributes, as well as potentially creating new features that capture the relationships between the two types of data.
C. Model Selection and Evaluation: Different machine learning algorithms have varying levels of effectiveness when working with mixed numeric and categorical data. Identifying the appropriate algorithms and evaluation metrics is crucial for ensuring the success of the combined approach.

Conclusion

By linking numeric KPI-based anomaly detection with categorical alarm-based detection, network operators can gain a more comprehensive understanding of their networks, leading to improved decision-making and more effective anomaly detection. In the following chapter, we will explore the development and evaluation of an anomaly detection approach focused on leveraging alarm data.

Chapter 5: Anomaly Detection Based on Alarms/Events Data

Abstract

Alarms data is a very important source of information for network operation center (NOC) teams to aggregate and display alarming events occurring within a network element. However, on a large network, a long list of alarms is generated almost continuously. Intelligent analytical reporting of these alarms is needed to help the NOC team to eliminate noise and focus on primary events. Hence, there is a need for an anomaly detection model to learn from and use historical alarms data to achieve this. It is also important to indicate the root cause of anomalies so that immediate corrective action can be taken. In this chapter, we introduce a new algorithm to derive four features based on historical data and aggregate them to generate a final score that is optimized through supervised labels for greater accuracy. These four features reflect the likelihood of occurrence of events, the sequence of events and the importance of relatively new events not seen in the historical data. Certain assumptions are tested on the data using the relevant statistical tests. After validating these assumptions, we measure the accuracy on labelled data, revealing that the proposed algorithm performs with a high anomaly detection accuracy.

5.1 Introduction

Anomaly detection is an aspect of data mining that has been the subject of research in many fields, such as telecommunications, information technology and finance.

There are several definitions of anomaly in the literature. Hawkins [1] defines an anomaly/outlier as an observation, which deviates considerably from the remaining observations, as if generated by a different process. Dunning and Friedman [2] state that anomaly detection involves modelling what is normal in order to discover what is not. In general, anomalies are events with a special behaviour that is dissimilar to that of normal events, and it is expected that this behaviour would be detected by analysing underlying data. Therefore, there is an urgent need for intelligent algorithms to identify such abnormal behaviour.

Anomaly detection improves data quality by deleting or replacing abnormal data. However, in certain cases, anomalies reflect an extreme event and can provide useful new knowledge. For example, the detection of such anomalies can prevent material damage and encourage predictive maintenance in the industrial field. It also has applications in several other areas such as health [3], cybersecurity [4], finance [5], natural disaster [6], and telecommunication [7].

Several methods have been proposed for detecting anomalies, each of which has its own strengths and weaknesses. Patcha and Park [8] reviewed all the known methods used for anomaly detection. Additionally, an overview of existing techniques covering several approaches is presented in [9] and [10].

Despite the large volume of literature on anomaly detection for numeric data e.g., time series, there is limited knowledge on the problem of abnormal behaviour in the context of categorical and structured textual data.

In this chapter, we aim to design an anomaly detection algorithm in the context of alarms data (categorical data) in the field of telecommunication. In other words, in a given period of time, each network element of a telecommunication network generates a set of Key Performance Indicators (KPIs) and alarms that describe its behaviour. Alarms are typically categorical data with different characteristics (i.e., name, description,

severity of the event, start time, end time), triggered to indicate a certain event occurring on the network element. Based on this information, those intervals of time are detected that have a high probability/score of displaying abnormal behaviour. Alarms data is important in a real-world context when KPIs are unavailable and cannot be calculated or extracted. It should be noted that alarms are events that can start popping up on a certain network element at any time. Therefore, each alarm can be considered to be equivalent to a binary random variable that can appear at any time with a certain probability.

Here, we propose an approach that introduces two new, innovative aspects. First, four features are calculated and aggregated to define events data during a certain interval of time; this includes the number of alarms, occurrence time, inter arrival time, transition frequency (Markovian model) and historical frequency. By combining this information, we compute an abnormality score which is, to the best of our knowledge, the first time that an anomaly detection algorithm has incorporated all the attributes of an event. In fact, in the majority of prior influential studies only a few of the previously cited attributes were considered. [11] consider just the Markovian component; in [12], a feature selection step is proposed prior to anomaly detection, which is a process that is associated with a high risk of loss of key information and requires significant effort for data labelling; [13] consider categorical data to be textual and vectorize it before the anomaly detection phase which is also associated with a high risk of loss of information. Second, the proposed algorithm enables users to extract local and focused information about one of the previously discussed features which may provide greater insight into the root cause of the anomaly (also known as anomaly fingerprint).

In this chapter, we first describe the methodology used to build the abnormality score. We then present an application of the algorithm and analyze the results.

5.2 Methodology

We propose a semi-parametric scoring system that reflects the different behavioral aspects of a component of a network during a given interval of time using alarms data generated for that component. These aspects are (5.2.1) the number of alarms, (5.2.2) the inter-arrival time between alarms, (5.2.3) the transition probability of two consecutive alarms, and (5.2.4) the

historical frequency of an alarm. The calculation of the final score is demonstrated in Subsection 5.2.5 and the optimization of the model weights is shown in Subsection 5.2.6. Because alarms are generated from each network component, of which there are different types, we group these components by type when drawing inferences from the data to reduce volatility and heterogeneity in the calculated statistics.

5.2.1 Number of Alarms

It is a common practice in parametric statistics to assume a Poisson distribution while modelling the number of occurrences of a certain event during a fixed period. Therefore, under this assumption, we begin by estimating the rate parameter λ of the Poisson distribution by calculating the arithmetic average of the number of alarms across all the intervals for each different component type of the network. Therefore, if we have L different types of components in the network, L different rate parameters $\lambda_1, \dots, \lambda_L$ are estimated.

Now, let $N_l,\ l = 1, \dots, L$ denote the random variables (r.v.) indicating the number of alarms generated by a component of type $l \in \{1, \dots, L\}$ over an interval of time. Based on the previous assumption, N_l follows a Poisson distribution with rate parameter λ_l. Note that $\mathbb{E}(N_l) = \lambda_l$ and it can easily be shown that the proposed estimator is a minimum variance unbiased estimator (MVUE) of λ. Hence, if n denotes the observed number of alarms in a fixed interval for a component of type l, the associated probability can be computed as shown in Equation (1).

$$\mathcal{P}_1^l = \mathbb{P}[N_l = n] = \frac{e^{-\lambda_l}\lambda_l^n}{n!} \quad (1)$$

Hence, in order to standardize this probability and transform it into a score that reflects the number of alarms, and the fact that a higher-than-average score indicates a higher probability of abnormal behaviour, S_1^l can be defined as:

$$S_1^l = \begin{cases} \dfrac{\mathbb{P}[N_l = int(\lambda_l)] - \mathcal{P}_1^l}{\mathbb{P}[N_l = int(\lambda_l)] - \min\limits_{\text{over all intervals}} \left(\mathcal{P}_1^l\right)}, & \text{if } n \geq int(\lambda_l) \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

where $int(\cdot)$ denotes the integer part of a real number. Then, a value of S_1^l close to one implies that the number of alarms indicates abnormal

behaviour in the specified interval. Note that $int(\lambda_l)$ represents the mode of a Poisson distribution of parameter λ_l.

5.2.2 Inter-arrival time

Using the same logic, we consider the intervals of time during which at least two alarms were detected for each type of component L. We also define the r.v. Y_l representing the time between two consecutive alarms that occurred within the same interval of time. It is common to model such r.v. by an exponential distribution with rate parameter μ_l, which is estimated by calculating the inverse of the arithmetic average of the time between two consecutive alarms during all the intervals for each different component type of the network. Note that under the previous assumption, $\mathbb{E}[Y_l] = {}^{1}/_{\mu_l}$. Hence, if the number of alarms during an interval for a component of type $l \in \{1, \dots, L\}$ is $n \geq 2$, an associated probability can be computed as shown in Equation (3).

$$\mathcal{P}_2^l = \mathbb{P}\left[Y_l \leq \frac{\sum_{j=1}^{n-1} y_j}{n-1}\right] = 1 - e^{-\left({}^{1}/_{\mu_l}\right)\frac{\sum_{j=1}^{n-1} y_j}{n-1}} \quad (3)$$

Where $y_j, j = 1, \dots, n-1$ denotes the time between alarms j and $j+1$.

Similarly, this probability can be standardized and transformed into a score to reflect that alarms occurring consecutively within a very short span of time are more likely to indicate abnormal behaviour, as shown in Equation (4).

$$S_2^l = \frac{\max\limits_{\text{over all intervals}}\left(\mathcal{P}_2^l\right) - \mathcal{P}_2^l}{\max\limits_{\text{over all intervals}}\left(\mathcal{P}_2^l\right) - \min\limits_{\text{over all intervals}}\left(\mathcal{P}_2^l\right)} \quad (4)$$

Here, a value of S_2^l close to one implies that the time between consecutive alarms indicates abnormal behaviour in the specified interval.

5.2.3 Transition probability

In the same context as that of the inter-arrival time score, and based on all the observed alarms during all the intervals for a component of type l, we define the state space of alarms as $E^l = \{a_1, \dots, a_K\}$, where K denotes the number of unique observed alarms in component l. Subsequently, we empirically compute the transition probabilities, $\forall i, j \in \{1, \dots, K\}$, as shown in Equation (5).

$$p_{a_i a_j} = \text{probability of observing } a_j \text{ after } a_i \quad (5)$$

Hence, we obtain a transition matrix in a similar manner to a Markov chain, that summarizes all the historical transitions that have occurred for each type of component. Then, to highlight abnormal behavior during a given interval, we identify the occurrence of transitions that are historically uncommon. Practically, if the number of alarms during an interval for a component of type l is $n \geq 2$, where these alarms are elements of the state space E^l denoted by $x_1, \dots, x_n$, an associated probability can be computed as shown in Equation (6).

$$\mathcal{P}_3^l = \min_{k=1,\dots,n-1} p_{a_i=x_k, a_j=x_{k+1}} \quad (6)$$

As described previously, the probability is standardized and transformed into a score to reflect that the alarms that occur consecutively and that have not occurred one after the other frequently in the past are more likely to be displaying abnormal behavior. This score is obtained as shown in Equation (7).

$$S_3^l = \frac{\max_{\text{over all intervals}} \left(\mathcal{P}_3^l\right) - \mathcal{P}_3^l}{\max_{\text{over all intervals}} \left(\mathcal{P}_3^l\right) - \min_{\text{over all intervals}} \left(\mathcal{P}_3^l\right)} \quad (7)$$

Here, a value of S_3^l close to one implies that during this interval, a non-frequent transition is occurring, which is likely to be abnormal behaviour.

5.2.4 Historical frequency

Now, we consider the historical frequency of the alarms occurring during an interval. In other words, an alarm of a certain type that is historically infrequent is considered to be more critical and should be highlighted. In real world scenarios, given that access to big data can be limited, this attribute helps in identifying infrequent or non-occurring events in the network, especially high impact events that occur rarely. Then, for a component of type l we consider the state space of alarms $E^l = \{a_1, \dots, a_K\}$, and the historical frequency of each of these alarms is computed and denoted by $f_i, i \in \{1, \dots, K\}$.

Further, to highlight abnormal behaviour during a given interval, we focus on the alarm with the lowest historical frequency among those that occurred during this interval, which are denoted by $x_1, \dots, x_n \in E^l$, with $n \geq 1$. $\mathcal{P}_4^l$ is first defined as shown in equation (8).

$$\mathcal{P}_4^l = \max_{k=1,\dots,n} \frac{1}{f_k} \quad (8)$$

This is derived using all the available historical intervals data. This is followed by standardization, where $\mathcal{P}_4^l$ is transformed into a score quantity as shown in Equation (9).

$$S_4^l = \frac{\mathcal{P}_4^l - \min_{\text{over all intervals}} \left(\mathcal{P}_4^l\right)}{\max_{\text{over all intervals}} \left(\mathcal{P}_4^l\right) - \min_{\text{over all intervals}} \left(\mathcal{P}_4^l\right)} \quad (9)$$

Here, a value of S_4^l close to one implies that a non-frequent alarm occurs during this interval, which indicates abnormal behaviour.

5.2.5 Final score and individual contributions

To obtain a final abnormality score for a given interval of time and for a particular component of the network of type $l \in \{1, \dots, L\}$, the previously computed scores are aggregated as weighted average measures as shown in Equation (10).

$$S^l = \sum_{i=1}^{4} w_i S_i^l \quad (10)$$

With $0 < w_i < 1$ and $\sum_{i=1}^{4} w_i = 1$.

The values of different weights are determined based on interactions with subject matter experts (SMEs) and a supervised grid search approach that will be discussed later.

A value of S^l close to one indicates that abnormal behaviour is being displayed during the specified interval and addressing this should be considered to be a priority for the SMEs. In addition, diagnostic information can be extracted from the four individual scores which may provide a starting point for the SMEs to analyse the root cause of the detected anomalies. This additional information is used to explain the derived score by specifying which alarms occurred, which inter-arrivals are low, which transitions are rare and which alarms have the lowest occurrence historically.

5.2.6. Validation and optimization

After assigning a score to each of the intervals across all the components of the network, we validate the results by comparing our labels to the ones given by the SMEs (labels are determined by manual inspection of the data to identify occurrences of anomalies). From the scores obtained in Subsection 5.2.5, the labels are determined based on a predefined fixed threshold denoted by s, such that:

$$S^{l}_{labeled} = \begin{cases} 1 & if\ S^{l} > s \\ 0 & \text{Otherwise} \end{cases} \quad (11)$$

The values of the weights in Subsection 5.2.5 and the value of the threshold s are determined based on a grid search process [14], where several scenarios/combinations of the underlying parameters are considered. The selected combination is the one with the best performance based on the accuracy of the confusion matrix that shows the degree of similarity between our labels and the SMEs labels, and the value of the Area Under the ROC Curve (AUC) [15]. Such optimization makes the algorithm similar to supervised ML models with the aim of maximizing the correlation between labels and features. This is a unique supervision method to replicate and learn human decisions.

5.3 Application

The methodology described in Section 5.2 is applied on real world data obtained from a virtual telecommunication network to identify intervals with a high probability of displaying abnormal behaviour. Data description, results and analyses, and the advantages of the proposed algorithm will be presented in Subsections 5.3.1, 5.3.2 and 5.3.3 respectively.

5.3.1. Data description

From a virtual telecommunication network, and for a given period, we consider alarms occurring on the different network elements over 30-minute intervals with a sliding window step of 5 minutes. The concept of the sliding window is introduced to consider events (i.e., alarms) that overlap between two consecutive intervals. In addition, to assure that the different aspects of the methodology of Section 5.2 are applicable, we estimate the parameters of the underlying distributions—Poisson for counting alarms and exponential for inter-arrival time—separately on the

three types of components that are present in the network and are indexed as $l \in \{1,2,3\}$.

These alarms (categorical data) with their different levels of severity, e.g., critical, minor and major, occurring during a given interval indicate the occurrence of abnormal behaviour. Based on the abnormality score that is computed by the proposed algorithm, SMEs should prioritize intervention in such cases.

5.3.2 Results and analysis

We first estimate the parameters of the Poisson distribution $\lambda_l, l \in \{1,2,3\}$, and the exponential distribution $\mu_l, l \in \{1,2,3\}$, for each type of component by applying the methods described in Subsections 5.2.1 and 5.2.2. For each type of component, all the available alarm occurrences across all the intervals are used. Note that only those intervals with at least two alarms are considered for the estimation of μ_l because the exponential distribution models the time between two consecutive alarms (in minutes). Moreover, goodness-of-fit tests are conducted to test the feasibility of the assumption that the number of alarms and the time between two consecutive alarms are governed respectively by Poisson and exponential distributions. Pearson's chi-squared test [16] is used for the goodness-of-fit test. The estimation results and the p-values of the statistical tests are represented in Tables 1 and 2 respectively.

Table 1. Parameter estimation.

Network element components	λ_l	μ_l
i=1	0.312	0.166
i=2	0.158	0.201
i=3	7.055	0.191

Table 2. Goodness of fit tests.

	p-values	
Network element components	**Poisson**	**Exponential**
i=1	0.98	0.84
i=2	0.231	0.279
i=3	0.785	0.871

Table 2 shows that when the different types of components are considered separately, the assumption about the underlying distribution appears to be reasonable. Therefore, the parametric approach defined in Subsections 5.2.1 and 5.2.2 can be relied upon to compute scores S_1^l and S_2^l for each interval for the different network elements.

The third type of score is based on a transition matrix of the probabilities of different descriptions of alarms for a given type of component. To compute such a matrix, the empirical approach described in Subsection 5.2.3 is applied. Table 3 shows the transition matrix of alarm descriptions for the component of type $l = 3$.

Table 3. Transition matrix.

Alarm Description	3(a)	3(b)	3(c)	3(d)	3(e)	3(f)	3(g)	3(h)
3(a)	1	0	0	0	0	0	0	0
3(b)	0	0.2	0	0	0	0	0	0.8
3(c)	0	0	1	0	0	0	0	0
3(d)	0	0	0	1	0	0	0	0
3(e)	0	0	0	0	0	0	1	0
3(f)	0	0	0	0	0	1	0	0
3(g)	0	0	0	0	0	0	1	0
3(h)	0	0.3	0	0	0	0	0	0.7

As an example of how to read Table 3, we can say that for network element of type $l = 3$, when at least two alarms occur during an interval, alarms of description 3(b) are followed by alarms of the same description in 20% of cases and alarms of description 3(h) in 80% of cases.

Using these matrices and applying the method described in Subsection 5.2.3, one can obtain the third score S_3^l for each interval for the different network elements.

The next step is to compute the historical frequency of each alarm description for a given type of component and to use these frequencies, as described in Subsection 5.2.4, to calculate the fourth score S_4^l for each interval for different network elements. An example of these frequencies is shown in Table 4 for the network element of type $l = 1$. Then, when an alarm of description 1(b) occurs during an interval and is observed to have a low historical frequency, then the interval is suspected to be displaying abnormal behavior.

Table 4. Frequency table.

Alarm Description	Frequency
1(a)	2248
1(b)	10
1(c)	8608
1(d)	2324
1(e)	862
1(f)	17684
1(g)	29
1(h)	253
1(i)	441
1(j)	1348

Now, for each component type $l \in \{1,2,3\}$ and for all the intervals, the final abnormality score S^l can be computed by applying Equation (10) and by setting the initial values for the weights $w_i, i = 1,2,3,4$ (e.g., $w_i = \frac{1}{4} \forall i$). In addition, to apply Equation (11), a threshold s needs to be determined to label all the intervals with 1 if abnormal behaviour is taking place and 0 otherwise.

To optimize the choice of the underlying weights and threshold, the SMEs label a parallel and independent abnormal behaviour based on the same data for the same intervals. Then, based on a random grid search process, the parameters of the algorithm are optimized, for each component type, on two levels. First, among all the tested combinations of weights w_i verifying $0 < w_i < 1$ and $\sum_{i=1}^{4} w_i = 1$ we select the one with the highest AUC. Second, among all the threshold used to draw the optimal ROC curve, we select the one with the highest accuracy in terms of true positives and true negatives, i.e., we maximize the sum of the diagonal terms of the confusion matrix. Hence, we begin the grid search process by considering the following equation:

$$w^* = \underset{w \in \mathfrak{D}}{\operatorname{argmax}}\, AUC(w) \quad (12)$$

where w is a vector of weights $w_i, i = 1,2,3,4$ and $\mathfrak{D}$ is the set of all the considered combinations of weights during the first level of the grid search process. Once the optimal combination of weights w^* is selected, we select optimal threshold by applying the following equation:

$$s^* = \underset{s \in \mathcal{T}}{\operatorname{argmax}}\big(TP(s) + TN(s)\big) \quad (13)$$

where $TP(s)$ and $TN(s)$ denote the true positive and true negative labels respectively when the threshold s is fixed. $\mathcal{T}$ represents the set of all the considered thresholds during the second level of the grid search process. The construction of the sets $\mathfrak{D}$ and $\mathcal{T}$ is done with collaboration and validation by the SMEs. Furthermore, we are not concerned by the phenomenon of overfitting because we are using the latter grid search process merely to optimize the selection of the underlying weights of different scores and the threshold based on a matching method with labels fixed by the SMEs.

Considering network element of type $l = 1$, Fig. 1 shows the optimal ROC curve with a maximum AUC of 0.975 corresponding to the vector of weights $w^* = (w_1^* = 0.41, w_2^* = 0.29, w_3^* = 0.2, w_4^* = 0.1\,)$ for intervals with at least two alarms (i.e., S_2^l and S_3^l are computable) and a vector $w^* = (w_1^* = 0.8, w_2^* = 0, w_3^* = 0, w_4^* = 0.2\,)$ for intervals with less than two alarms.

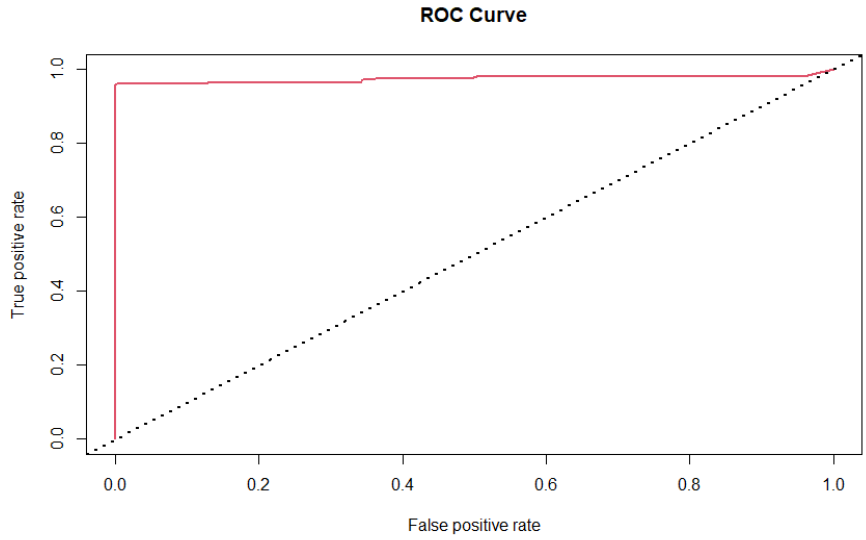


Figure 1. Optimal ROC curve

Table 5 shows the optimal confusion matrix corresponding to a threshold of 0.58. In other words, for components of type $l = 1$, we will apply the following rule to label anomalous behaviour across all the intervals:

$$S^{l}_{labeled} = \begin{cases} 1 & if\ S^{l} > 0.58 \\ 0 & \text{Otherwise} \end{cases}.$$

In the following certain interpretations and metrics are discussed based on the confusion matrix:

- True positive: 2521 intervals; False positive: 531 intervals.
- True negative: 199513 intervals; False negative: 103 intervals.
- Sensitivity: proportion of true positive among SMEs abnormal intervals: $^{2521}/_{(2521+103)} = 0.961$.
- Specificity: proportion of true negative among SMEs non abnormal intervals: $^{199513}/_{(199513+531)} = 0.997$.
- Accuracy: $^{(2521+199513)}/_{(2521+531+103+199513)} = 0.997$.

Table 5. Confusion matrix.

Predicted Labels \ SMEs Labels	1	0
1	2521	531
0	103	199513

Based on all the previous metrics computed after interactions with the SMEs, it is evident that the proposed algorithm is performing well with a high accuracy, and that we can rely on it to detect abnormal behaviour in future intervals. Furthermore, the algorithm is applied on online arriving

alarms data and intervals that have been declared as anomalous and validated by experts. Here, approximately 1% of the intervals under control were behaving in an abnormal way, which is very reasonable in practice and is commonly encountered by SMEs. In addition, the algorithm presented in this chapter has several advantages when compared to the classical anomaly detection approach. Most of these advantages will be enumerated in the next subsection.

5.3.3 Advantages of the proposed algorithm

Compared to popular anomaly detection models, the proposed algorithm has four main advantages:

- Our algorithm is already adapted to be an online anomaly detection model applied directly to new arrivals for the purpose of highlighting abnormal behaviour. Hence, there is no need to train such a model on a sample and to test it on another because such a model has no risk of overfitting.

- To the best of our knowledge, this is the first time that an anomaly detection algorithm, based solely on alarms categorical data, has successfully extracted diagnostic information from the four different components of the global score (i.e., S_1^l, S_2^l, S_3^l and S_4^l) to help SMEs initiate root cause analysis of the detected anomalies.

- The proposed algorithm has the ability to generate abnormality scores based solely on alarms data without any additional information about numeric KPIs, which is uncommon in the field of anomaly detection for telecommunication networks.

- The interpretability of this model adds great value and is important for both developers and users.

5.3.4 Test of independence between different type of alarms

The algorithm proposed in this chapter assumes the existence of one family of alarms. In fact, if other families of alarms are available, our model can easily be generalized by proposing a weighted anomaly score for the different families of alarms. Additionally, we can consider the same

optimization process proposed in Subsections 5.2.6 and 5.3.2 to determine the values of the different weights.

Further, to ensure the statistical independence between different families of alarms in terms of occurrence time we suggest an independence test. This is essentially a uniform distribution goodness-of-fit test using classical chi-squared test. Therefore, for alarms of family $\mathcal{A}$, we test whether the occurrence times of such alarms, between two alarms of another family $\mathcal{B}$, are uniformly distributed. It is important to note that in order to apply such an approach, the intervals of time separating the occurrence of two alarms of the same family need to be normalized.

5.4 Conclusion and Perspectives

In this chapter, an innovative anomaly detection algorithm that solely uses structured alarms (categorical data) has been presented. The proposed model takes into consideration four different attributes extracted from alarms occurrence data to compute a global anomaly score. This can then be used to extract diagnostic information that helps SMEs in performing root cause analysis. Our algorithm is shown to be more advantageous than other existing anomaly detection models, when applied in the same context.

Moreover, we applied the algorithm to real data in the field of telecommunication. The results were then validated by SMEs who provided positive feedback and found that our algorithm outperforms the previously used classical approaches. Users of such a model are also convinced by its output because it relies on the behaviour of historical data and generates real-time ranking of events occurring on a network component in terms of abnormality.

A first perspective of this work is to mathematically formalize a model/algorithm using the extracted information from different sub-scores in order to enhance existing root cause analysis methods based solely on alarms data. A second perspective is to combine alarms data with other type of non-numeric features, e.g., textual data, to build a more complete anomaly detection approach that covers novel aspects that have not been addressed before. Such pioneering work can be initiated by drawing inspiration from [17].

References

[1] D. M. Hawkins, Identification of outliers, vol. 11. Springer, 1980.

[2] T. Dunning and E. Friedman, Practical machine learning: a new look at anomaly detection. " O'Reilly Media, Inc.", 2014.

[3] A. Ukil, S. Bandyoapdhyay, C. Puri, and A. Pal, "IoT healthcare analytics: The importance of anomaly detection", In 2016 IEEE 30th international Conference on advanced information networking and applications (AINA), pp. 994-997, IEEE, 2016.

[4] D. A. Bierbrauer, A. Chang, W. Kritzer, and N. D. Bastian, "Anomaly detection in cybersecurity: Unsupervised, graph-based and supervised learning methods in adversarial environments," arXiv preprint arXiv:2105.06742, 2021.

[5] M. Sekar, "Fraud and anomaly detection," in Machine Learning for Auditors, pp. 193–202, Springer, 2022.

[6] S. Miau and W.-H. Hung, "River flooding forecasting and anomaly detection based on deep learning," IEEE Access, vol. 8, pp. 198384–198402, 2020.

[7] M. Kamel, A. Hoayek and M. B. Hubert "Probabilistic approach for anomaly detection with geometric dynamics," unpublished.

[8] A. Patcha and J.-M. Park, "An overview of anomaly detection techniques: Existing solutions and latest technological trends," Computer networks, vol. 51, no. 12, pp. 3448–3470, 2007.

[9] C. C. Aggarwal, "An introduction to outlier analysis," in Outlier analysis, pp. 1–34, Springer, 2017.

[10] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," ACM computing surveys (CSUR), vol. 41, no. 3, pp. 1–58, 2009.

[11] H. Ren, Z. Ye, and Z. Li, "Anomaly detection based on a dynamic Markov model," Information Sciences, vol. 411, pp. 52–65, 2017.

[12] Y. Liu, H. Xu, H. Yi, Z. Lin, J. Kang, W. Xia, Q. Shi, Y. Liao, and Y. Ying, "Network anomaly detection based on dynamic hierarchical clustering of cross domain data," in 2017 IEEE International

Conference on Software Quality, Reliability and Security Companion (QRS-C), pp. 200–204, IEEE, 2017

[13] B. Nie, J. Xu, J. Alter, H. Chen, and E. Smirni, "Mining multivariate discrete event sequences for knowledge discovery and anomaly detection," in 2020 50th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), pp. 552–563, IEEE, 2020.

[14] M. Claesen and B. De Moor, "Hyperparameter search in machine learning," arXiv preprint arXiv:1502.02127, 2015.

[15] T. Fawcett, "An introduction to roc analysis," Pattern recognition letters, vol. 27, no. 8, pp. 861–874, 2006.

[16] K. Pearson, "X. on the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling," The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, vol. 50, no. 302, pp. 157–175, 1900.

[17] K. Ezukwoke, H. Toubakh, A. Hoayek, M. Batton-Hubert, X. Boucher, and P. Gounet, "Intelligent fault analysis decision flow in semiconductor industry 4.0 using natural language processing with deep clustering," in 2021 IEEE 17th International Conference on Automation Science and Engineering (CASE), pp. 429–436, IEEE, 2021.

Chapter 6: Expanding the Anomaly Detection Horizon to Syslogs

In the previous chapters, we have examined anomaly detection in telecommunication networks through the lens of numeric Key Performance Indicator (KPI) data and categorical alarm data. Both approaches have proven valuable in their respective domains; however, there is another crucial source of information that can further enhance our understanding of network anomalies: syslog data. Before diving into a new chapter dedicated to anomaly detection based on syslog data, we must establish a connection between the alarm-based approach and the upcoming syslog-based approach, demonstrating the benefits of incorporating this additional data source in our analysis.

The Role of Syslogs in Telecommunication Networks

Syslogs are event logs generated by network elements, providing detailed information about operational events and activities occurring within the network. Syslog data is vital for understanding both normal and anomalous network behavior, as it contains granular information about network events that might not be captured in KPIs or alarm data.

Incorporating syslog data into our anomaly detection framework can complement the insights derived from KPI and alarm data, offering a more comprehensive perspective on network health and performance. For instance, a series of syslogs indicating repeated login attempts on a network device might suggest an unauthorized access attempt not reflected in KPI or alarm data.

Challenges in Integrating Syslog Data

Integrating syslog data with KPI and alarm data presents several challenges due to the unique characteristics of this data source:

a. Data Preprocessing: Syslogs are typically unstructured, consisting of text messages and event codes. Transforming this data into a structured format suitable for machine learning algorithms requires sophisticated preprocessing techniques, such as text mining and natural language processing.

b. Feature Selection and Engineering: Syslog data may contain a large number of distinct events and attributes, necessitating careful feature selection and engineering to identify the most relevant and informative features for anomaly detection.

c. Model Selection and Evaluation: Identifying machine learning algorithms capable of effectively handling the combination of numeric, categorical, and text data requires thorough experimentation and evaluation to ensure the best possible performance.

Conclusion

By incorporating syslog data into our anomaly detection framework, we can achieve a more comprehensive understanding of network anomalies, leading to more accurate and timely detection. In the next chapter, we will explore the development and evaluation of a novel approach to anomaly detection that leverages syslog data, ultimately enhancing our overall framework by considering all three data sources – numeric KPIs, categorical alarms, and text-based syslogs.

Chapter 7: Anomaly detection based on SysLogs textual data

Abstract

With the increase of network virtualization and the disparity of vendors, the continuous monitoring and detection of anomalies cannot rely on static rules. An advanced analytical methodology is needed to discriminate between normal events and unusual anomalies. In this chapter, we focus on logs data (textual data), which is a crucial source of information for network performance. Then, we introduce an algorithm, used as a pipeline to help with the pretreatment of such data, group it in patterns, and dynamically label each pattern into anomaly or not. Such tools will provide users and experts with continuous real-time logs monitoring capability, to detect anomalies and failures in the underlying system that can affect performance. The algorithm is illustrated by an application on real-world data.

7.1 INTRODUCTION

System log is a hub containing all the information about the events taking place during system operation. In general, one talks about textual log messages when system log data has a structured textual format [1]. In the telecommunication field, the deployed logging infrastructure helps to store and aggregate logs produced by all the components of the network—e.g., cells, core, transport—in a continuous way. Having access to all these historic logs triggers a series of analyses and studies, improving the performance of the network and optimizing the whole industrial process, e.g., analysis of customer fulfillment [2].

In addition, a popular field in which textual logs are used is anomaly detection (AD), followed by a root cause analysis (RCA) of abnormal behavior that may occur in a system. Several machine learning and deep learning methods are used to this end (for a survey on these techniques, see [3]). In general, AD is a field of data science encountered in various areas of research, such as health [4], cybersecurity [5], finance [6], natural disaster [7], and telecommunication [8]. In fact, in the literature, one may find several definitions of an anomaly in a set of data. For example, Pang et al. [9] speak about anomalies as a novelty in a set of data or time series, and define it as an instance that significantly deviates from the majority of data instances. Ruff et al. [10] start by defining concepts of normality, and then explain that an anomaly is an observation not respecting some of these concepts. The most important point to keep in mind is that an anomaly, when it occurs, has a significant impact on the underlying process. Hence, it is crucial to develop machine learning (ML) and artificial intelligence (AI) algorithms to detect such outliers.

Without being exhaustive, there are several algorithms and models for anomaly detection in the literature. A survey of existing methods that may be adapted to different contexts and problems, with various levels of complexity reflected by underlying data of diversified typologies (e.g., numeric, categorical), can be found in [11], [12] and [13]. However, to the best of authors' knowledge, there are few research works dealing with detecting abnormal behavior in a system based solely on structured textual data.

In this chapter, the aim is to build an AD algorithm based solely on system logs historical data for an application in the field of telecommunication. In

fact, for a fixed time window, different elements of a telecommunication network provide different types of indicators that help experts assess the system's performance and describe the behavior of each element. These indicators include, among others, textual log messages collected and aggregated to keep track of all the events taking place with the corresponding characteristics (i.e., description, occurrence time, severity of the log). Using all these sources of information, the main goal is to detect an interval of time showing a high probability of abnormal behavior. Note that our work becomes of high interest when the only available data in a system is the logs, which is the case in many real-world applications.

In this work, we start by a pretreatment of the textual logs data based on selected natural language processing techniques (NLP). In the second step, we cluster textual data based on a novel developed unsupervised clustering algorithm that considers the specificities of our data, without requiring an intermediate vectorization step. The aim of this second step is to extract homogeneous subfamilies of observations—an indicator that will be used in the final phase of assigning anomaly labels to each of the intervals. The main three contributions of the method that we are proposing are: (1) the new clustering algorithm is designed to be adapted to the structure of the textual data that we are dealing with, and to decrease, by as much as possible, the loss of information by avoiding transforming the raw textual data into numeric data through a vectorization process; (2) before blindly applying an anomaly detection model using vectorized data, an optimal structure of the decision space is extracted through an unsupervised clustering technique, which will lead to a more accurate and homogeneous abnormality scoring system; (3) an optimization of the underlying hyperparameters of the proposed algorithm is conducted based on an innovative supervised approach, taking into consideration the interaction with the experts of the domain, in order to improve the performance of the proposed AD method.

This chapter is organized as follows. Section 5.2 describes the whole methodology used to get abnormality labels for each network element on different time intervals. An application of the method on real world data, followed by a discussion of the results, are then presented in Section 5.3. A conclusion closes the chapter.

7.2 Methodology

The algorithm we are proposing uses a non-parametric approach, generating abnormality labels for each component of a telecommunication network during a fixed time interval using only system logs as underlying data. The different steps of the method are: (5.2.1) preprocess textual data based on common NLP rules; and (5.2.2) cluster using a novel unsupervised algorithm, then assign abnormality labels to selected clusters. As logs are generated from all network components, of which there are different types, we decided to deal with each type separately, while drawing inferences in order to reduce volatility and increase the model's accuracy.

7.2.1 Preprocessing textual data

As the system logs we are dealing with use a structured textual data format, the first step is preprocessing the data based on certain NLP rules (for a review of NLP rules, see [14]). The idea behind preprocessing textual data is to reduce noise and artifacts that may have a negative influence on the quality of the algorithms and machine learning models that will be used in the next steps. In addition, the selection of rules that should be applied is fully adapted to the structure of our underlying data, and it was chosen based on discussions with subject matter experts (SMEs). The list of the rules that we are applying is as follows:

- Remove timestamps and all time-related words (e.g., Jan, Feb, Mon)
- Remove links, path strings, host names, and IP addresses (e.g., python-requests/2.6.0 CPython/2.7.5 Linux/3.10.0-1062.1.2.el7.x86_64)
- Remove symbols (e.g., @, [], /)
- Make all characters lowercase (e.g., change ERROR to error)
- Remove 'words' comprising only numbers
- Remove two-characters 'words' (e.g., f1, c2, aa)
- Combine 'not_word' and 'no_such_word' by "_"

Table I shows three examples of logs from the data, before and after applying the preprocessing rules.

Original log message	Cleaned log message
sshd[36261]:%AUTH-2: rad_send_request: Invalid RADIUS response received	Sshd auth rad_send_request invalid radius response received
: %PFE-3: fpc0 daemon.err sensord: Error updating RRD file:/var/run/sensord.rrd: /var/run/sensord.rrd: /var/run/sensord.rrd: /var/run/sensord.rrd: /var/run/sensord.rrd: ….	pfe sensord error updating rrd file
Server contrail_webui/overcloudcfn-cc-0.internalapi.nbg991 is DOWN, reason: Layer6 timeout, check duration: 2000ms. 2 active and 0 backup servers left. 0 sessions active, 0 requeued, 0 remaining in queue.	server down reason timeout check duration active and backup servers sessions active requeued remaining

TABLE I Logs preprocessing

7.2.2 Clustering and structure extraction algorithm

Due to the large number of logs continuously generated, it is nearly impossible to analyze the log messages as they are. Thus, after cleaning these messages, as described in the previous section, it is useful to reduce their complexity, using a clustering approach to highlight a hidden structure of the underlying space, by dividing it into subfamilies.

There are many clustering algorithms adapted for system logs data proposed in the literature, with many concepts and specifications, depending on the practitioners' objectives (for more details, see [15], [16] and [17]). Most of these methods start by converting text to numeric data (the 'vectorization' step), and then apply a classic clustering algorithm, such as K-means [18]. This may not fit our problem, however, as the volume of our data is huge, and its streaming behavior is highly complex. In addition, the algorithm that we are proposing bypasses vectorization step, because

such a step always generates a loss of information, which, in some cases, may be significant for the decision maker. In other words, contrary to the approach commonly found in the literature, we designed our own clustering and family extraction strategy, to avoid both simple frequency-based vectorization methods (e.g., TF-IDF [19]) and deep black box vectorization methods (e.g., word2vev [20]).

Once we have the preprocessed textual data, the next step is to inject the cleaned logs data as an input into the algorithm, which we designed in a customized way, to fit our objectives, to be adapted to the structure of the underlying decision space, and to be validated by SMEs.

Therefore, our novel algorithm consists of the following steps:

Step 1: Log lines are read line by line from a log file or log stream. The strings are then preprocessed (as described in Subsection 5.2.1) to remove certain artifacts, which have a negative influence on the quality of the clustering.

Step 2: The first log line that is being read, denoted by l_0, always forms a new cluster with itself, and is considered the representative of the cluster. In general, a representative of a cluster C in the set of clusters $\mathcal{C}$ is denoted by c.

Step 3: For each other log line l, a set of cluster candidates $\mathcal{C}_l \subseteq \mathcal{C}$ is selected, based on the lengths of c and l (note that l denotes the most recent log line). A cluster C is added to $\mathcal{C}_l$ if:

$$(|\,l\,| - s|\,l\,|) \leq |\,c\,| \leq (|\,l\,| + s|\,l\,|) \qquad (1)$$

where $|\,l\,|$ and $|c|$ represent the number of tokens in l and c respectively, and s is a rate showing the maximum accepted length difference between logs. If Equation (1) is not verified by any of the cluster representatives, the log line l forms a new cluster with itself as representative, and we repeat the algorithm for the next log starting from Step 3.

Step 4: The most similar of the cluster candidates $\mathcal{C}_l$ is determined using a string metric distance based on tokens. For this, the cluster C in $\mathcal{C}_l$ that minimizes the distance $d(c,l)$ is selected, where:

$$d(c,l) = 1 - \frac{number\ of\ common\ tokens}{\sqrt{|c| \times |l|}} \qquad (2)$$

Step 5: If this distance to the most similar representative c is not larger than a predefined threshold t, then l is allocated to C. Otherwise, the log line forms a new cluster with itself as representative, and we repeat the algorithm for the next log starting from Step 3.

Note that the proposed algorithm makes the clustering by focusing on both the length of the logs (first criteria—see Step 3) and token similarity (second criteria—see Step 4). After finishing the training process and the construction of the different clusters based on a given historical set of data, isolated clusters of logs, with a number of observations less than a threshold $\mathcal{N}$, are considered abnormal clusters. On the other hand, in practice the algorithm will be applied in an online learning way, assigning each new arriving log to one of the predefined clusters (in the training phase) and

deciding about its abnormality. Finally, in the case that the algorithm fails to assign a new arriving log to the existing clusters, this log will be directly considered abnormal. In fact, the training phase and the definition of the underlying clusters is updated on a weekly basis, to take into consideration system updates of all types.

In addition, the distance (normalized distance measure) proposed in Step 4 may be replaced by any other distance, and can be considered as a hyperparameter of the method that needs to be optimized based on the global performance of the corresponding algorithm. Finally, the values of the thresholds s, t and $\mathcal{N}$ will be also optimized, using a grid search process, where the aim is to select the combination with the best performance, based on the accuracy of a confusion matrix showing how the algorithm's labels fit those of the SMEs. So, for the optimization phase we are proposing a unique supervised method, learning human decisions and taking into consideration the interaction between the machine and the SMEs, in order to maximize the correlation between labels and logs. This whole optimization process is detailed in Subsection 5.3.2.

7.3 Application

Now, we will be applying the method described in Section 7.2 on real-world data extracted by a virtual telecommunication network, with the aim of detecting time intervals during which there is a high probability of an abnormal behavior occurring. Data description will be the subject of Subsection 7.3.1, followed by the results and analysis in Subsection 7.3.2. Finally, advantages of the proposed algorithm will be presented in Subsection 7.3.3.

7.3.1 Data description

A set of the historical logs generated by the system (the virtual telecommunication network) is used to define the different clusters described in Subsection 7.2.2. Afterwards, these logs are aggregated to one minute as time dimension, and then a count of logs by cluster is completed every minute (this can be parametrized based on model usage requirements).

On the other hand, to get better performance, and to ensure that the algorithm is applied on homogeneous data and the source of noise is reduced as much as possible, we optimize the hyperparameters of the method estimate separately, using the four component types present in the considered virtual network, which are indexed as $k \in \{1,2,3,4\}$.

These logs, with their different characteristics and time intervals, will detect moments and intervals that should be prioritized by the SMEs because of their high abnormal behavior probability.

7.3.2 Results and discussions

For each type of component, after getting the different clusters and identifying abnormal ones based on the algorithm described in Subsection 7.2.2, we move on to showing the distribution of logs over the different clusters for every minute, using the per-minute aggregated data. The number of log occurrences for each of the different cluster IDs is then represented in a table, using the format of Table II, which shows an example of the adopted representation for the component of type $k = 1$. Note that, based on the proposed algorithm, we get nine different log clusters for the type 1 component, and clusters 2, 5 and 7 are considered abnormal, because at the end of the training process the number of observations in these clusters is less than the threshold $\mathcal{N}$.

Intervals Aggregated by one minute	Cluster 1	Cluster 2	…	Cluster 9
1	0	0	…	1
2	0	2	…	0
3	0	0	…	3
4	3	0	…	1
5	2	0	…	5
6	0	0	…	0
7	3	0	…	0
⋮	⋮	⋮	…	⋮

TABLE II Logs distribution over clusters

Then, based on Table II, one can say that, for a component of type 1, interval 2 is showing an abnormal behavior, given that logs of cluster 2 are detected in this interval. Hence, such an interval can be labeled abnormal based on the proposed algorithm.

Now, we move to work on the optimization of the different threshold of the algorithm. To do this, SMEs start labeling abnormal intervals independently, by looking to the same dataset that the algorithm is using. Next, using a grid search technique [21], the optimal thresholds are selected, by applying the following optimization:

$$(s^*, t^*, \mathcal{N}^*) = \underset{(s,t,\mathcal{N})\in\mathfrak{D}}{\operatorname{argmax}} \left(TP(s,t,\mathcal{N}) + TN(s,t,\mathcal{N})\right) \qquad (3)$$

$TP(s,t,\mathcal{N})$ and $TN(s,t,\mathcal{N})$ denote the true positive and true negative labels, respectively, for a given combination of thresholds $(s,t,\mathcal{N})$. $\mathfrak{D}$ represents the set of all the considered combination thresholds that will be used for the grid search process. The structure of the set $\mathfrak{D}$ is constructed following collaboration with and validation from the SMEs.

Table III shows the confusion matrix corresponding to the vector of optimal thresholds $(s^* = 0.1, t^* = 0.25, \mathcal{N}^* = 85)$, retrieved by a grid search process.

Predicted Labels \ SMEs Labels	1	0
1	322	45
0	52	5,387

TABLE III Confusion matrix

Key metrics that help assess the quality of the proposed algorithm are then extracted from the confusion matrix table. These are shown below.

- True positive: 322 observations (i.e., intervals)
- False positive: 45 observations
- True negative: 5,387 observations
- False negative: 52 observations
- Sensitivity: proportion of true positives among SMEs' abnormal intervals: $^{322}/_{(322+52)} = 86.1\%$
- Specificity: proportion of true negatives among SMEs' non abnormal intervals: $^{5,387}/_{(5,387+45)} = 99.2\%$
- Accuracy: proportion of well detected intervals

$$^{(322+5,387)}/_{(322+45+52+5,387)} = 98.3\%$$

Thus, the validation of the algorithm's results with the SMEs' confirms that the proposed methodology is reliable and is detecting anomalies with a high credibility. In the next subsection, we further discuss the specificities of our algorithm.

7.3.3 Particularities of the methodology

The algorithm that we are proposing in this chapter, to detect abnormal behavior in a telecommunication network based on system logs, has several particularities not found in similar, state-of-the-art methods:

- To the best of our knowledge, this is the first time an anomaly detection model has been developed based solely on textual log data, without passing through a vectorization phase and so avoiding loss of information.

- The method is adapted to online learning, where abnormal behavior is easily and quickly detected among new arrivals.

- Our algorithm is nonparametric and distribution-free, in the sense that the user is not obliged to make assumptions about the underlying probabilistic distribution of the decision space and the related parameters that need to be estimated.

- The proposed algorithm can generate abnormality labels based solely on log data, without any additional information about numeric KPIs, which is uncommon in the field of anomaly detection for telecommunication networks.

- As the data is not vectorized, the model has high level of interpretability and helps both developers and users to use it, with the aim of extracting information about the root cause of a detected anomaly.

- The proposed algorithm takes into consideration both the length of logs and token similarity, as criteria from which to extract the structure of the underlying space.

7.4 Conclusion and perspectives

In this work, a new anomaly detection algorithm, using solely textual log data provided by a telecommunication network, has been introduced. The algorithm comprises several phases: (1) preprocessing textual data based on an adapted pipeline; (2) unsupervised similarity and clustering analysis, made directly on textual data, without the need of vectorization phase; (3) definition of abnormal clusters based on three key thresholds, each optimized using a specific supervised approach that takes into consideration an interaction with SMEs.

An application of the algorithm on real-world data has been presented. The results show a good performance, based on several metrics calculated from a confusion matrix. In addition, we have received positive feedback from the SMEs regarding several aspects of the algorithm, particularly the simplicity of its usage and the interpretability of its results.

Finally, we suggest two main avenues of further research, based on this work: (1) The clusters we are generating may be used as input features for another machine learning algorithm, in order to calculate an abnormality score, instead of having a binary decision about anomalous intervals; (2) Propose an aggregation method to combine the decision of the proposed algorithm with others developed using numeric data [22] and categorical/events data [23], in such a way as to produce one global

abnormality score for each time interval in a telecommunication network.

References

[1] L. Tang and T. Li, "Logtree: A framework for generating system events from raw textual logs," in 2010 IEEE International Conference on Data Mining, pp. 491–500, IEEE, 2010.

[2] E. Mahendrawathi, H. M. Astuti, and A. Nastiti, "Analysis of customer fulfilment with process mining: A case study in a telecommunication company," Procedia Computer Science, vol. 72, pp. 588–596, 2015.

[3] R. B. Yadav, P. S. Kumar, and S. V. Dhavale, "A survey on log anomaly detection using deep learning," in 2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO), pp. 1215–1220, IEEE, 2020.

[4] A. Ukil, S. Bandyoapdhyay, C. Puri, and A. Pal, "Iot healthcare analytics: The importance of anomaly detection," in 2016 IEEE 30th international conference on advanced information networking and applications (AINA), pp. 994–997, IEEE, 2016.

[5] D. A. Bierbrauer, A. Chang, W. Kritzer, and N. D. Bastian, "Anomaly detection in cybersecurity: Unsupervised, graph-based and supervised learning methods in adversarial environments," arXiv preprint arXiv:2105.06742, 2021.

[6] M. Sekar, "Fraud and anomaly detection," in Machine Learning for Auditors, pp. 193–202, Springer, 2022.

[7] S. Miau and W.-H. Hung, "River flooding forecasting and anomaly detection based on deep learning," IEEE Access, vol. 8, pp. 198384–198402, 2020.

[8] A. Bouillard, A. Junier, and B. Ronot, "Hidden anomaly detection in telecommunication networks," in 2012 8th international conference on network and service management (cnsm) and 2012 workshop on systems virtualiztion management (svm), pp. 82–90, IEEE, 2012.

[9] G. Pang, C. Shen, L. Cao, and A. V. D. Hengel, "Deep learning for anomaly detection: A review," ACM Computing Surveys (CSUR), vol. 54, no. 2, pp. 1–38, 2021.

[10] L. Ruff, J. R. Kauffmann, R. A. Vandermeulen, G. Montavon, W. Samek, M. Kloft, T. G. Dietterich, and K.-R. Müller, "A unifying review of deep and shallow anomaly detection," Proceedings of the IEEE, vol. 109, no. 5, pp. 756–795, 2021.

[11] A. Patcha and J.-M. Park, "An overview of anomaly detection techniques: Existing solutions and latest technological trends," Computer networks, vol. 51, no. 12, pp. 3448–3470, 2007.

[12] C. C. Aggarwal, "An introduction to outlier analysis," in Outlier analysis, pp. 1–34, Springer, 2017.

[13] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," ACM computing surveys (CSUR), vol. 41, no. 3, pp. 1–58, 2009.

[14] L. Hickman, S. Thapa, L. Tay, M. Cao, and P. Srinivasan, "Text preprocessing for text mining in organizational research: Review and recommendations," Organizational Research Methods, vol. 25, no. 1, pp. 114–146, 2022.

[15] Z. Yang, S. Ying, B. Wang, Y. Li, B. Dong, J. Geng, and T. Zhang, "A system fault diagnosis method with a reclustering algorithm," Scientific Programming, vol. 2021, 2021.

[16] R. Yang, D. Qu, Y. Qian, Y. Dai, and S. Zhu, "An online log template extraction method based on hierarchical clustering," EURASIP Journal on Wireless Communications and Networking, vol. 2019, no. 1, pp. 1–12, 2019. Z. Xiao, L. Kong, B. Zhang, J. Qian, and F. Jin, "Information entropy based density clustering algorithm of database log," in 2021 IEEE 24th International Conference on Computer Supported Cooperative Work in Design (CSCWD), pp. 102–108, IEEE, 2021.

[17] Z. Xiao, L. Kong, B. Zhang, J. Qian, and F. Jin, "Information entropy based density clustering algorithm of database log," in 2021 IEEE 24th International Conference on Computer Supported Cooperative Work in Design (CSCWD), pp. 102–108, IEEE, 2021.

[18] J. MacQueen, "Classification and analysis of multivariate observations," in 5th Berkeley Symp. Math. Statist. Probability, pp. 281–297, 1967.

[19] K. S. Jones, "A statistical interpretation of term specificity and its application in retrieval," Journal of documentation, 1972.

[20] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," arXiv preprint arXiv:1301.3781 , 2013.

[21] M. Claesen and B. De Moor, "Hyperparameter search in machine learning," arXiv preprint arXiv:1502.02127, 2015.

[22] M. Kamel, A. Hoayek and M. B. Hubert "Probabilistic approach for anomaly detection with geometric dynamics," unpublished.

[23] M. Kamel, A. Hoayek and M. B. Hubert "Anomaly detection based on alarms data," International conference on AI, Machine Learning in Communications and Networks, 2022, in press.

Chapter 8: From whole distribution approach to a tail distribution approach

The previous three chapters highlight various challenges in existing anomaly detection models and propose novel algorithms to address them. In this chapter, we will discuss the importance of these proposed approaches and their linkage to the next chapter, which proposes a new method based on record and extreme value theory for anomaly detection.

Chapter three proposed a new geometrical multidimensional probabilistic model that addresses the challenges of long training times and the inability to quantify the drivers of abnormal behavior when mining in a multidimensional space. The proposed model searches for abnormal behavior in the data space, generates anomaly scores, and quantifies anomaly drivers. Additionally, the chapter proposed a data-driven definition of an outlier score to prioritize devices anomalies and a sampling approach to speed up scoring newcomer observations.

Chapter four proposed an algorithm that derives four features based on historical alarms data to address the challenge of a long list of alarms generated almost continuously. These four features reflect the likelihood of occurrence of events, the sequence of events, and the importance of relatively new events not seen in the historical data. The proposed algorithm is optimized through supervised labels for greater accuracy.

Chapter five introduced an algorithm that addresses the challenge of continuous monitoring and detection of anomalies using logs data. The proposed algorithm helps with the pretreatment of logs data, groups it in patterns, and dynamically labels each pattern into anomaly or not. This approach provides continuous real-time logs monitoring capability to detect anomalies and failures in the underlying system that can affect performance.

The proposed approaches in these three chapters address the challenges in existing anomaly detection models used for telecommunication networks. They improve the efficiency and accuracy of anomaly detection, help network engineers prioritize network maintenance tasks, and assist in detecting anomalies and identifying their drivers. The proposed methods can be applied to different types of data and can aid network engineers in maintaining networks efficiently.

The following chapter will propose a new approach based on record and extreme value theory for anomaly detection. This approach focuses on the behavior of the tails of underlying variables, rather than the entire distribution, and introduces a novel anomaly scoring system that can distinguish between rare and common events. This approach has several advantages over current state-of-the-art models and is demonstrated by implementing it on a real-world dataset.

Chapter 9: Record theory for Anomaly detection & information selection

The proliferation of interconnected devices is rapidly expanding globally, and, as a result, telecommunication operators are responsible for managing intricate and expansive networks. Consequently, there is a need for advanced and efficient systems to aid network engineers in maintaining these networks. Devices, which can also be referred to as network elements, continuously transmit essential performance data known as key performance indicators. By utilizing data derived from these metrics and implementing intelligent anomaly detection models, the devices can assist in determining the optimal production maintenance schedule for the network. As anomaly detection models deal with extreme events, this study proposes a method of reducing dimensions by focusing on the behavior of the tails of underlying variables, rather than the entire distribution. In addition to that, an anomaly scoring system, also based on records theory, is proposed, which has several advantages over current state-of-the-art models. The effectiveness of this approach is demonstrated by implementing it on a real-world dataset.

9.1 Introduction

Identifying anomalies in a time series refers to identifying observations that differ from the typical pattern of the other observations. Such anomalies are uncommon and significant as they can have an impact on the underlying system that generates the time series. It is crucial to quickly and accurately detect these abnormal behaviors to ensure the proper functioning of the upstream system (Chandola et al., 2009). Anomaly detection is a research topic that is encountered in various fields such as industry (Zhou et al., 2020), cybersecurity (Rashid et al., 2022), healthcare (Sabic et al., 2021), environment (Vangipuram et al., 2020), and telecommunication (Kamel et al., 2023; Ali et al., 2020).

Various research areas, including machine learning (ML) (Alvi et al., 2022), statistical learning (Sha et al., 2015), game theory (Huang et al., 2019), and graph theory (Akoglu et al., 2015) have contributed to the development of current state-of-the-art anomaly detection algorithms and models.

The literature has addressed several questions and challenges related to anomaly detection. One such challenge is the ability to generate abnormality scores in an unsupervised context, where auto-encoder (AE) neural network models are the most widely used approach. Another challenge is detecting abnormal behavior without making any assumptions about the probabilistic distribution of the underlying random variables, which is addressed by using random-forest–based models. However, there are still many challenges that cannot be tackled by traditional anomaly detection models and require innovative approaches.

This chapter aims to address several research gaps in anomaly detection. First, it proposes a dimension reduction method that is adapted to abnormal and extreme events for dealing with large datasets. This method uses records theory (see Sections 6.2 and 6.3 for details) which is a branch of extreme value theory to develop a variable selection methodology that focuses on the behavior of the tails of the underlying variables rather than the whole distribution, as in classical dimension reduction methods like principal component analysis and AE. Second, this work uses records theory to propose an abnormality scoring system that can be used in one or multidimensional datasets. This system generates a density distribution of the scores and uses a grid search process to minimize classification errors and set a threshold value for the underlying variables above which an observation is considered abnormal. This threshold can be communicated

to subject matter experts (SMEs), and, to the best of the authors' knowledge, this is the first algorithm to propose an anomaly threshold for each considered variable. Third, the proposed anomaly detection algorithm is adapted to online learning modes and is optimized in terms of computational complexity, despite being coupled with a variable selection method. Finally, the proposed approach provides initial information about the root cause of a detected anomaly.

In summary, the main objective of this chapter is to develop a comprehensive methodology that enables users to detect anomalies in time series data while addressing the challenges associated with this task.

The proposed method is primarily designed to detect abnormal behavior in different elements of a telecommunication network. These network elements generate numerous key performance indicators (KPIs), and the sheer number of features that describe the performance of the different services provided is enormous, making manual analysis of these observations challenging, if not impossible. In addition, the ability to identify irregularities in real time with minimal delays requires the utilization of sophisticated correlation analysis and extensive data mining techniques to reveal concealed patterns and associations within the generated data.

The rest of this chapter is structured as follows. Section 9.2 provides an introduction to records theory. Section 9.3 presents the mathematical formalization of the most popular models in records theory and how they are adapted to the current context. Section 9.4 describes the use of records theory for variable selection. Section 9.5 shows how records theory is used to generate abnormality scores in one and multidimensional datasets. Section 9.6 demonstrates some real-world applications. Finally, Section 9.7 concludes the chapter.

9.2 Records, an Introduction

The study of records in a time series as a field of extreme value theory can be traced back to Chandler's work in 1952. Since then, there have been numerous developments in this field, including the works of Arnold, Nevzorov, and their collaborators during the 1980s and 1990s. Initially, researchers focused on the classic case in records theory, which assumes that the random variables (RV) are independent and identically distributed (IID). However, this case did not fully capture the complexity of multiple

datasets, so researchers began to explore cases where the observations are independent but not identically distributed. Eventually, they even considered the most general case where neither the independence assumption nor the assumption of identical distribution holds.

Data are found in the form of records across various fields that use statistics, such as sports (Yang, 1975), climate change (Wergen and Krug, 2010; Wergen, 2013), risk assessment of diseases (Khraibani et al., 2015), financial markets (Hoayek et al., 2018), and satellite imagery (Jabbour et al., 2021).

It is worth noting that there is a greater interest in records when they are the only available values in a particular time series. Since records are a part of popular culture, they are usually kept in easily accessible places, such as the Guinness World Records.

In simpler terms, a record is a result in a given series of events that exceeds anything seen before. Therefore, a new record is always something remarkable and attracts attention, whether it is associated with positive or negative news.

Our research applies records theory to solve two challenges related to anomaly detection in an industrial context. The first challenge is to reduce the dimensionality when dealing with a large number of time series to detect abnormal behavior. The second challenge is to develop an innovative ML model that efficiently and accurately detects anomalies using the principles of records theory, which aims to model extreme values.

9.3 Mathematical Formalization

We start by considering the probability space $(\Omega, \mathcal{F}, \mathbb{P})$. Here, X denotes a real RV with a cumulative distribution function (CDF) $F(\cdot)$ and a density function $f(\cdot)$. We assume that the space $(\Omega, \mathcal{F}, \mathbb{P})$ has good properties to define an infinite sequence $\{X_t, t \geq 1\}$ of IID RVs, which are independent copies of X. When the index t represents time, then we are dealing with a time series having an IID underlying distribution. An observation X_t is considered an upper record at time t if it is higher than all previous observations, that is, $X_t > \max(X_1, \dots, X_{t-1})$. In this chapter, we focus on upper records, but lower records can be defined similarly, by multiplying the time series by "−1". As time progresses, another important sequence of RVs can be defined: the sequence of record values $\{R_n, n \geq 1\}$ and the sequence of

occurrence time of records $\{L_n, n \geq 1\}$. In other words, L_n is the occurrence time of the n^{th} record, which is $R_n = X_{L_n}$.

In most applications of records theory, the available data consists of a sequence of pairs $\{(R_n, L_n), n = 1, \dots, N_T\}$, where T represents the current time (i.e., length of the time series) and N_T is the total number of records in $\{X_t, t = 1, \dots, T\}$.

In addition to the previously defined sequences, one can also define the sequence of record indicators $\{\delta_t, t = 1, \dots, T\}$, where:

$$\delta_t = \begin{cases} 1 \text{ if } X_t \text{ is a record} \\ 0 \quad \text{otherwise} \end{cases}. \qquad (1)$$

Note that $\delta_1 = 1$ because the first observation is always a record, which is called a trivial record.

We will demonstrate later, in this section, that based solely on the sequence of record indicators, we can extract significant information about the overall behavior of the records in a time series. Note that it is straightforward to remark that:

$$N_T = \sum_{t=1}^{T} \delta_t.$$

The stochastic properties of sequences of record values have been widely studied in the case where X_t are IID RVs (Arnold et al., 2011; Nevzorov, 2001). Many of these properties are distribution-free, meaning that they do not depend on the choice of the underlying distribution of the observations. The most important results in the IID context are:

First, $\forall t \geq 1$, δ_t are mutually independent and follow a Bernoulli distribution with parameter $P_t = \frac{1}{t}$, which is called the record rate at time t. In other words, $\mathbb{P}[\delta_t = 1] = \frac{1}{t}$, which is the probability of observing a record at time t, and $\mathbb{E}[\delta_t] = \frac{1}{t}$. It is worth noting that:

$$\lim_{t \to +\infty} P_t = 0. \qquad (2)$$

Therefore, one can conclude that records are more likely to appear among the first observations. In addition, the expected number of records until time T is given by:

$$\mathbb{E}[N_T] = \sum_{t=1}^{T} \mathbb{E}[\delta_t],$$

$$= \sum_{t=1}^{T} \frac{1}{t}. \qquad (3)$$

Moreover, Arnold et al. (2011) found that records tend to become more spread out over time a t or n increases. However, this was not always the case in practice. For example, advancements in technology are causing sports records to occur more frequently than what is expected under the IID assumption. As a result, more complex models have been developed to better predict records beyond the classical IID case. These models can be grouped into two families based on their level of complexity, which we will discuss in the next two subsections.

9.3.1 Independent but Not Identically Distributed Observations

First, consider the case where underlying observations are independent but not identically distributed. In this context, two common models are used:

- Linear drift model (LDM), introduced by Ballerini and Resnick in 1985, formalized by:

  $$X_t = Y_t + \theta t, \qquad (4)$$

  where $Y_t, t \geq 1$ is a sequence of IID RVs and $\theta > 0$ is a parameter that needs to be estimated.

- Yang record model, initially introduced by Yang (1975) and later developed by Nevzorov (1988). This model is considered more suitable for the independent but not identically distributed context and in most cases, it is more generalized than the LDM. The Yang model can be represented by the following formula:

  $$X_t \sim F(\cdot)^{\rho_t}, \qquad (5)$$

  where $\rho_t (t \geq 1)$ are real constants ≥ 1 and $F(\cdot)$ is a CDF of a particular underlying distribution. In this chapter, we will focus on a specific parametrization of the Yang model, in which $\rho_t = \gamma^t$, with γ being a parameter that needs to be estimated and is ≥ 1. This formalization is interesting because it has the structure of a proportional hazard model, which is commonly used in survival analysis to model various datasets (Hoayek et al., 2017). In addition, each X_t represents the maximum value obtained from ρ_t observations that are generated simultaneously and independently at time t from the same underlying RV Y of CDF $F(\cdot)$. Then,

  $$X_t = \max\left(y_1, y_2, \dots, y_{\rho_t}\right). \qquad (6)$$

  Based on the fact that the underlying RV Y is independent, it can be demonstrated that the record rate at time t is expressed as follows:

$$P_t = \mathbb{P}[\delta_t = 1] = \frac{\rho_t}{\sum_{k=1}^{t} \rho_k},$$
$$P_t = \frac{\gamma^t}{\sum_{k=1}^{t} \gamma^k} = \frac{\gamma^t(\gamma - 1)}{\gamma(\gamma^t - 1)}. \qquad (7)$$

In this case P_t will be denoted as $P_t(\gamma)$.
Thus,

$$\lim_{t \to +\infty} P_t(\gamma) = \lim_{t \to +\infty} \frac{(\gamma - 1)}{\gamma\left(1 - {}^{1}/_{\gamma^t}\right)} = \frac{\gamma - 1}{\gamma}. \qquad (8)$$

Therefore, in the Yang model, the probability of having new records in the long term does not decrease. As a result, a time series exhibiting this type of behavior can be considered more volatile and unstable compared to the classical IID case.

Despite its usefulness in various applications, the Yang model cannot be utilized in practice without first estimating the parameter γ. To do this, Hoayek et al. (2017) proposed an estimation method based on maximizing the following Log-Likelihood function that was constructed using solely the observed sequence of indicators:

$$\mathrm{Log}\, L(\gamma) = \mathrm{Log}\, \mathbb{P}[\delta_1, \dots, \delta_T; \gamma]. \qquad (10)$$

Then, by solving,

$$\frac{\mathrm{d}\, \mathrm{Log}\, L(\gamma)}{\mathrm{d}\gamma} = 0, \qquad (11)$$

we get our estimator which is denoted by $\hat{\gamma}$. In addition, also based on the work of Hoayek et al. (2017), one can show the asymptotic behavior of $\hat{\gamma}$ which is also distribution-free:

$$\frac{(\hat{\gamma} - \gamma)}{\sqrt{I_T^{-1}(\gamma)}} \longrightarrow N(0,1), \qquad (12)$$

Here, $I_T^{-1}(\gamma)$ represents the Fisher information associated with the previous likelihood. Therefore, by understanding the asymptotic behavior of our estimator, we can conduct further inferential analysis such as constructing confidence intervals for a given asymptotic risk of error level α.

Additionally, in the same context, Nevzorov (1988) demonstrated that record indicators are mutually independent, regardless of the choice of the underlying distribution Y. Thus, it can be concluded that the stochastic process $\{\delta_t\}_{t \geq 1}$ is a Bernoulli process with parameter P_t.

Using this property, we can obtain the expression of the expected value and variance of the number of records:

$$\mathbb{E}[N_T] = \sum_{t=1}^{T} \mathbb{E}[\delta_t] = \sum_{t=1}^{T} P_t, \qquad (13)$$

$$\mathbb{V}[N_T] = \sum_{t=1}^{T} \mathbb{V}[\delta_t] = \sum_{t=1}^{T} P_t\,(1 - P_t) = \mathbb{E}[N_T] - \sum_{t=1}^{T} {P_t}^2. \qquad (14)$$

9.3.2 Dependent and Not Identically Distributed Observations

Another level of complexity arises when we consider the scenario where underlying observations are dependent and not identically distributed. In this context, the most prevalent record model is the discrete-time random walk model (DTRW) introduced by Majumdar and Ziff (2008). The underlying observations in this model can be formalized as follows:

$$X_t = X_{t-1} + \eta_t, \qquad (15)$$

where the increments η_t are drawn from a continuous distribution in an IID way.

In the context of DTRW, the record rate at time t can be expressed as:

$$P_t = \mathbb{P}[\delta_t = 1] = \binom{2t}{t} 2^{-2t}. \qquad (16)$$

Majumdar and Ziff (2008) demonstrated that P_t asymptotically approaches zero in the DTRW model, though at a slower rate than it does for the IID case. Therefore, it can be concluded that in terms of long-term record probability, the DTRW model lies somewhere between the classical IID model and the Yang model. Additionally, it is worth noting that the majority of the results on DTRW are distribution-free.

Figure I provides a comprehensive overview of the behavior of record rates for different record models. Note that in Figure I, without any loss of generality, the parameter γ of the Yang model is assumed to be equal to 1.2.

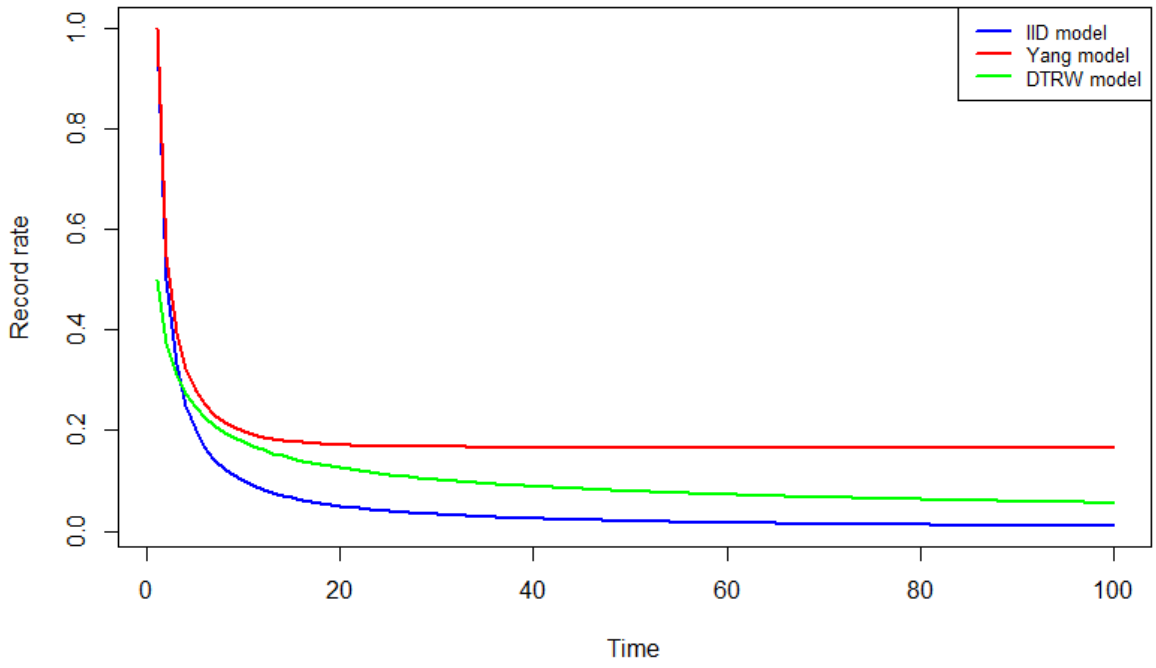


Figure I: Record rates for different record models for a time series of length 100.

9.3.3 Record Model Selection

Selecting the appropriate model to explain the record behavior of a time series involves performing a sequence of statistical tests.

Before considering non-IID models, we begin by testing whether the underlying observations of the time series in question are generated from an IID sequence. We do this by considering the null hypothesis:
H_0: Records are generated from an IID underlying sequence of observations.

To perform this first test, we use the fact that under H_0, Arnold et al. (2011) showed that:

$$\mathcal{N}_T = \frac{N_T - \log T}{\sqrt{\log T}} \longrightarrow \text{Standard Gaussian Distribution } N(0,1). \qquad (17)$$

Therefore, $\mathcal{N}_T$ can be viewed as the statistic used in the test. In practice, if $\mathcal{N}_T > q_{1-\alpha}$, where $q_{1-\alpha}$ is the $(1-\alpha)^{th}$ quantile of $N(0,1)$, then the IID model is rejected, and we should consider one of the models outside the classical case.

The next step is to establish a statistical test for the Yang model. Assuming the Yang model hypothesis, we create an RV referred to as the inter-record time (i.e., the time between two consecutive records), which is defined as:

$$\Delta_{L_n} = L_{n+1} - L_n, n \geq 1. \qquad (18)$$

Hoayek et al. (2017) showed that in a Yang model Δ_{L_n} follow a geometric distribution asymptotically. Therefore, we can use this result to construct a goodness of fit test for the Yang model. The null hypothesis for this test is that the inter-record time observations fit a geometric distribution. To conduct this test, we can adapt Pearson's chi-square test to the context of record models (for details see Hoayek et al., 2017). If Pearson's test rejects the geometric distribution, it is not appropriate to use the Yang model, and we should consider moving to a higher level of complexity where observations are dependent and not identically distributed. A common method to test the dependency between underlying observations is to use the Ljung–Box test (1978).

9.4 Variable Selection Based on Record Behavior

We will apply the methodology outlined in the previous section to create a variable selection tool for detecting anomalies. Most anomaly detection algorithms are designed to identify abnormal behavior, and users often aim to avoid the problem of dimensionality that can lead to increased computational costs, especially during the training phase.

To address this issue, various classical solutions have been proposed, such as linear and non-linear dimension reduction methods like principal component analysis and AEs, as well as variable selection methods like genetic algorithms. However, all of these methods consider the entire multidimensional distribution behavior of the underlying variables to determine how dimension reduction should be performed. This approach may not be suitable for certain application contexts, particularly when the focus is on the tails distribution behavior of the variables, as in the fields of anomaly detection and extreme event detection. Therefore, the proposed method is innovative and specifically adapted for the anomaly detection case. The method primarily focuses on the behavior of extreme events, especially upper records, to determine which variables should be selected. By prioritizing the features that are critical during anomaly detection, the dimension of the decision space is reduced, and the application of any algorithm becomes faster and less computationally complex. This is particularly important for online detection purposes.

In practice, we will use a collection of time series, known as KPIs, to evaluate the quality of services provided by a virtual telecommunication network. These KPIs will be used to identify abnormal behavior for each element of the network.

However, before applying any of the anomaly detection algorithms, we will assess each KPI separately and assign a priority level to each of them based on the following rules:

a. KPIs with high priority: when record behavior related to the underlying distribution of the KPI follows the Yang model. In this case, the probability of observing a new record on a long-term basis is constant, and extreme abnormal behavior is always likely to occur at any time. Therefore, such KPIs are considered risky in the context of anomaly detection.

b. KPIs with medium priority: when the DTRW model is accepted as a description of record behavior. In this case, record rates converge to zero in the long term, but at a slower rate than in the classical record IID model (see Figure I). Such records are considered to have medium risk and may have a significant impact on abnormal behavior.

c. KPIs with low priority: when records fit the classical IDD case. In such cases, record rates converge rapidly to zero, and abnormal behavior is observed very rarely on a long-term basis. Such KPIs are considered to have low risk and can be removed from later global abnormal behavior analysis.

To track the changes in the behavior of KPIs over time, we will apply the priority classification rules on sliding windows of fixed length $'k'$ with a step size $'s'$. This will allow us to monitor the KPIs' behavior over time and make any necessary adjustments to their priority levels. After assessing each window, a final decision on the priority level of the KPI will be made by aggregating the results of all the windows, using a rule determined by SMEs. In practical terms, the entire time series is examined for each KPI and then broken down into sliding windows. Using the information gathered from each window, we carry out the following steps:

- **Step 0:** If the KPI shows concerning high values, do not change the time series observations $X_t^w, t = 1, \dots, N = k$, where X_t^w denotes the observation of the considered KPI at time t in window w. Otherwise, transform the time series by considering $-X_t^w$ instead of X_t^w. In both cases, focus on the upper records for analysis.

- **Step 1:** From the time series obtained at the end of Step 0, extract record observations (R_n, L_n) and calculate the values of record indicators RV δ_t and the number of records N_T.

- **Step 2:** Test for an IID behavior based on the statistic of Eq. (17). If the behavior is classical, assign a low priority to the window. Otherwise, proceed to Step 3.

- **Step 3:** Calculate the values of the inter-record times RV Δ_{L_n} and use them to perform the goodness of fit test for the Yang model. If the Yang model is reasonable, assign a high priority to the considered window. Otherwise, consider that we are in the context of the DTRW model and assign a medium priority to the considered window.

- **Step 5:** Repeat Steps 1 to 3 for all sliding windows and assign priority decisions for each window.

- **Step 6:** Aggregate the results for all windows using a rule established by SMEs and assign the resultant priority to the corresponding KPI. For example, consider the highest priority assigned across all windows as the KPI priority.

9.5 Anomaly Scoring System Based on Records Distribution

9.5.1 One Dimensional Abnormality Score

It is crucial to create a scoring system based on records to detect anomalies in a random variable that exhibits extreme behavior and abnormal events. Suppose $\{X_t, t \geq 1\}$ is a time series that represents the behavior of a specific KPI over time with real values. $\forall t \geq 1$, we denote:

$$\Lambda_t = \left\{R_n, n \geq 1 \text{ such that } R_n \text{ is the } n^{th} \text{ record of the series } \{X_i, 1 \leq i < t\}\right\},$$
$$\Lambda_t^* = \{R_n \in \Lambda_t \text{ such that } R_n \geq X_t\},$$
$$\mathcal{D}_t = \{R_n - X_t, \text{such that } R_n \in \Lambda_t^*\},$$
$\overline{\mathcal{D}_t} = \text{Arithmetic average of the elements of } \mathcal{D}_t.$

Now for each observation $X_t, t \geq 1$ the corresponding abnormality score is given by:

$$\mathcal{S}_t = \begin{cases} 1, & \text{if } X_t \text{ is a record} \\ \frac{1}{\left(1 + \frac{\text{Card}\,\Lambda_t^*}{\text{Card}\,\Lambda_t}\right)} \times \left(\frac{1}{1 + \overline{\mathcal{D}_t}}\right), & \text{Otherwise} \end{cases}.$$

Where $\text{Card}(\cdot)$ gives the number of elements in a given set.

Assuming that the time series $\{X_t, t \geq 1\}$ has been standardized to have values between 0 and 1, and transformed so that high values indicate abnormal behavior, the $\mathcal{S}_t$ will fall between $\frac{1}{4}$ and 1. This score is calculated based on upper records only. Whenever X_t reaches its maximum (i.e, $X_t = 1$), it is considered a new record.

On the other hand, when $\mathcal{S}_t$ is closer to 1, it indicates a higher risk of abnormal behavior. Each component of $\mathcal{S}_t$ focuses on an aspect of abnormality in the underlying time series based on records:

- $\frac{1}{\left(1+\frac{\text{Card}\,\Lambda_t^*}{\text{Card}\,\Lambda_t}\right)}$: This component is closer to 1 when almost all the records taking place before t are lower than X_t. Therefore, in this case, even if X_t is not a record, it has an impact that is comparable to the majority of the previously detected records and should be highlighted as a potential anomaly.

- $\left(\frac{1}{1+\overline{\mathcal{D}_t}}\right)$: This component has a complementary role to the previous one. Here, we are computing the average distance between the observation X_t and all previously detected records with a value higher than X_t (elements of Λ_t^*). Thus, for this component, we obtain a value close to 1 when the value of X_t is close to the records of the set Λ_t^* which is also a scenario that should be highlighted in the process of detecting potential abnormal behavior.

While not an exhaustive list, the proposed record-based scoring system offers several advantages over classical anomaly detection models:

1. Unlike popular anomaly detection ML models, there is no risk of overfitting because there is no classical training/testing phase in the proposed algorithm. Additionally, the algorithm is designed to function as an online anomaly score system, generating a score for each new arrival.

2. The algorithm is distribution-free, meaning there is no need to make assumptions about the probability distribution of the underlying random variables in each time series.

3. The algorithm is parameter-free, requiring no statistical estimation or numerical optimization.

4. The approach has low computational complexity, allowing for fast generation of scores, giving SMEs the necessary time to intervene and address any detected anomalies.

5. Unlike most ML anomaly detection models, the threshold scores and values used to classify observations as anomalies are automatically fixed, minimizing the risk of confusion and ensuring optimal algorithm performance. This approach also allows for proposing optimal threshold values for each KPI, above which the KPI becomes alarming (further clarification is provided in the application section). This is the first anomaly score system to generate scores and assist with setting optimal scoring thresholds with minimal intervention from SMEs.

Note that, to address the risk of the first records in a time series being declared as anomalies, even if their values are not high enough, a practical solution is to run the algorithm on a warm-up period before initiating the extraction and detection of anomalies.

9.5.2 Multidimensional Abnormality Score

To obtain a more comprehensive understanding of abnormal behaviors, it is preferable to develop a scoring system that takes into account all available features at a given point in time and generates an abnormality score reflecting the interaction between all variables (i.e., KPIs). Suppose that we have l variables characterizing the status of a system over time, denoted by $\{X_t^i, t \geq 1 \text{ and i} = 1, \dots, \text{l}\}$. As a first step, we define upper records in a multidimensional context using the following two definitions:

1. $\forall t \geq 1$, an observation $X_t = (X_t^1, \dots, X_t^l)$ is considered to be an upper record if it is a record on at least one of the underlying dimensions. In other words, if there exists an $\exists i \in \{1, \dots, l\}$ such that $X_t^i > \max_{j<t} X_j^i$.

   This definition is referred to as the "At Least One-Based Multidimensional Record" (ALO) in the rest of this chapter. It is worth noting that this definition of records in a multidimensional context is introduced in Arnold et al. (2011; page 266).

2. $\forall t \geq 1$, the first step is to compute the Euclidian distance from the origin to the observation $X_t = (X_t^1, \dots, X_t^l)$:

$$d_t = \sqrt{\sum_{i=1}^{l} (X_t^i)^2}$$

   Then, based on the time series $\{d_t, t \geq 1\}$ instead of $\{X_t, t \geq 1\}$, the abnormality score at time t is computed in the same manner as in Subsection 6.5.1. This approach will be called the "Distance-Based Multidimensional Record". However, this approach has a weakness in that it transforms the multidimensional data into one distance series, losing information about the impact of each underlying variable on the final abnormality score. Consequently, this approach cannot interpret the scores on a variable (KPI) level or determine the root cause of the anomaly. Since SMEs prefer models that can be used for both anomaly scoring and root cause analysis, the ALO approach will be the sole focus of the chapter going forward.

Once the record series of the underlying multidimensional time series dataset has been collected using the ALO approach, the next step is to modify the abnormality score formula proposed in Subsection 6.5.1 to suit

the multinational context. Let $\{X_t, t \geq 1\}$ be the multidimensional time series that displays the behavior of l KPIs over time. $\forall t \geq 1$, we denote:

$$\Lambda_t = \left\{R_n = (R_n^1, \dots, R_n^l), n \geq 1 \text{ such that } R_n \text{ is the } n^{th} \text{ record of the series } \{X_i, 1 \leq i < t\}\right\},$$

$$\Lambda_t^* = \left\{R_n \in \Lambda_t \text{ such that } \exists j \in \{1, \dots, l\} \text{ with } R_n^j \geq X_t^j\right\},$$

$$\Lambda_{t,j}^* = \left\{R_n \in \Lambda_t^* \text{ such that } R_n^j \geq X_t^j\right\} \text{ with } j \in \{1, \dots, l\},$$

$$\mathcal{D}_{t,j} = \left\{R_n^j - X_t^j, \text{such that } R_n \in \Lambda_{t,j}^*\right\} \text{ with } j \in \{1, \dots, l\},$$

$$\overline{\mathcal{D}}_{t,j} = \begin{cases} 0, & \text{if } \mathcal{D}_{t,j} = \emptyset \\ \text{Arithmetic average of the elements of } \mathcal{D}_{t,j}, & \text{Otherwise,} \end{cases} \quad \text{with } j \in \{1, \dots, l\}.$$

Then, for each observation $X_t, t \geq 1$ the corresponding abnormality score is given by:

$$\mathcal{S}_t = \begin{cases} 1, & \text{if } X_t \text{ is a record} \\ \dfrac{1}{\left(1 + \dfrac{\text{Card}\,\Lambda_t^*}{\text{Card}\,\Lambda_t}\right)}\left(\dfrac{1}{1 + \sum_{j=1}^{l} \overline{\mathcal{D}}_{t,j}}\right), & \text{Otherwise} \end{cases}.$$

9.6 Real-World Data Application

The data analyzed in this research consists of 18 primary metrics (KPIs) that assess the quality of service provided by a virtual telecommunications network cell. These KPIs are consolidated hourly, resulting in 955 observations in total, where each observation represents 1 hour of data.

The KPIs include metrics such as Downlink and Uplink volume of data, Downlink and Uplink throughput, network availability, call setup success rate, and dropped call rate. These metrics play a vital role in measuring the efficiency and effectiveness of data transmission over the network, as well as the overall performance of the cell.

Analyzing the dataset provides valuable insights into the virtual telecommunication network cell's performance and helps identify areas for improvement. For example, a high dropped call rate could indicate network congestion or other issues that need to be addressed by implementing corrective measures to enhance the quality of service offered to customers. In summary, the dataset used in this study presents a comprehensive view of the virtual telecommunication network cell's performance, empowering network operators to make informed decisions about resource allocation and optimize network performance to enhance the user experience.

To account for the specificities of telecom time series data, the KPI values have been standardized to fit within the interval [0,1] and transformed to give higher values a more alarming indication of abnormal behavior. Therefore, we are working within a space of dimensions $[0,1]^{955\times18}$, with a focus on the upper records for each of the underlying variables. It should be noted that when a KPI reaches the upper bound (e.g., KPI = 1), this observation is regarded as a new record.

Before starting anomaly scoring, a feature selection process is undertaken using the methodology described in Section 6.4, where only the high and medium risk KPIs are considered, following the Yang and DTRW record models, respectively.

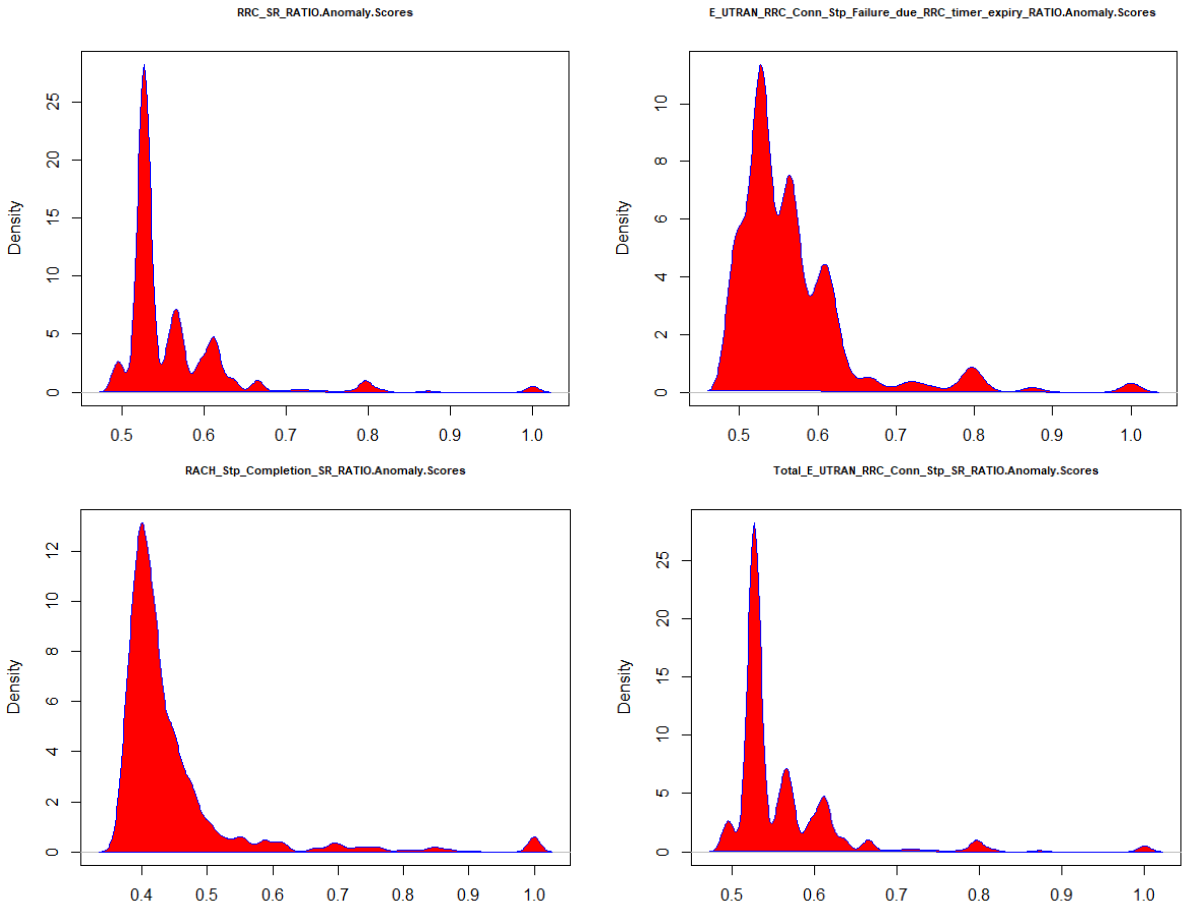
Table I shows the selected KPIs and their corresponding risk levels in terms of anomaly detection.

KPI	Risk Level
RRC_SR_RATIO	High
E_UTRAN_RRC_Conn_Stp_Failure_due_RRC_timer_expiry_RATIO	High
RACH_Stp_Completion_SR_RATIO	Medium
Total_E_UTRAN_RRC_Conn_Stp_SR_RATIO	High
E_RAB_QCI1_DR_RATIO	Medium
DCR_LTE_RATIO	Medium
LTE_INTER_ENODEB_HOSR_RATIO	Medium
E_UTRAN_tot_HO_SR_inter_eNB_X2_RATIO	High
DL_THROUGHPUT_RATIO	High
E_RAB_DR_RATIO	Medium

Table I: Risk level of the selected KPIs

For each of the chosen KPIs, the one-dimensional abnormality score, developed in Subsection 9.5.1, is calculated and the kernel density function of the scores is plotted in Figure II. It is evident that the probability density functions are multimodal, and that the abnormality scores associated with each of the KPIs can effectively discriminate between observations classified

as normal and abnormal, using a threshold that can be determined by a simple descriptive analysis of the various distributions. Therefore, by establishing these score thresholds, an optimal corresponding KPI threshold can be recommended to SMEs to minimize classification errors. For example, consider LTE_INTER_ENODEB_HOSR_RATIO. Based on Figure II, the recommended abnormality score threshold is 0.7 (i.e., an observation with a score above 0.7 is deemed anomalous). A grid search technique is then applied to determine the optimal KPI value threshold, which is found to be 0.1131 (i.e., an observation with a KPI value above 0.1131 is considered anomalous), with a classification error rate of 5.23%. This is the first time that an anomaly detection algorithm has been able to propose a threshold for SMEs to consider, rather than the opposite. Results for all KPIs are presented in Table II.



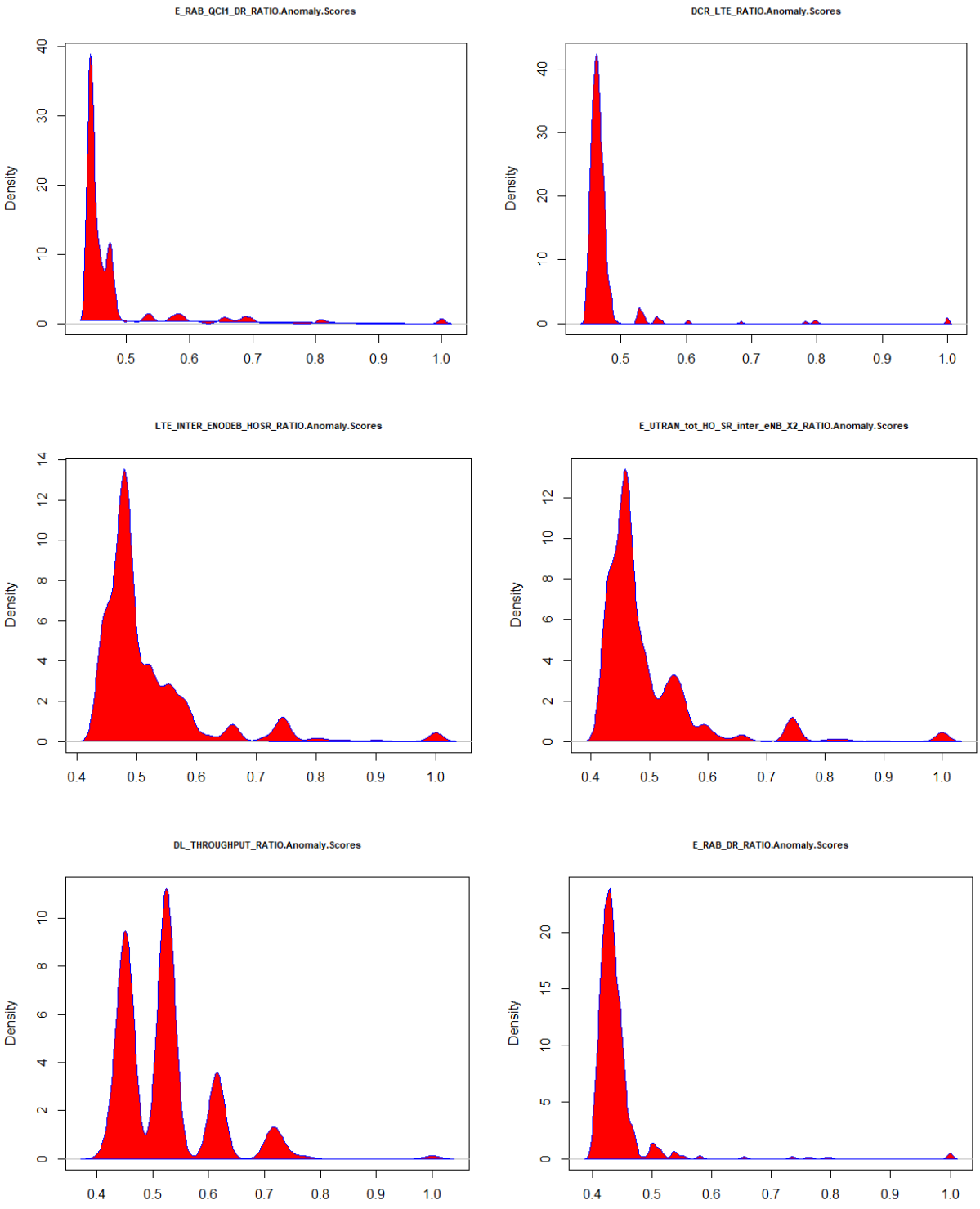


Figure II: Kernel density functions of the one-dimensional abnormality scores of each KPI.

KPI	Score Threshold	KPI Value Threshold	Error %
RRC_SR_RATIO	0.69	0.0103	1.47%
E_UTRAN_RRC_Conn_Stp_Failure_due_ RRC_timer_expiry_RATIO	0.85	0.0163	1.26%
RACH_Stp_Completion_SR_RATIO	0.65	0.2693	2.83%
Total_E_UTRAN_RRC_Conn_Stp_SR_RATIO	0.69	0.0103	1.47%
E_RAB_QCI1_DR_RATIO	0.71	0.1492	2.30%
DCR_LTE_RATIO	0.51	0.0854	0%
LTE_INTER_ENODEB_HOSR_RATIO	0.7	0.1131	5.23%
E_UTRAN_tot_HO_SR_inter_eNB_X2_RATIO	0	0.4835	5.13%
DL_THROUGHPUT_RATIO	0.8	0.4414	0.31%
E_RAB_DR_RATIO	0.6	0.241	0.21%

Table II: One-dimensional anomaly score analysis

To assess the relationship between all the KPIs and generate a single anomaly score that represents the behavior of all the underlying variables, we will use the ALO method discussed in Subsection 9.5.2. The graph in Figure III shows the kernel density function of the abnormality scores that were calculated. This distribution is bimodal, making it easy to distinguish between anomalies and non-anomalies without the need to estimate a threshold, unlike traditional anomaly detection models. Using this method, we identified 4.4% of observations as abnormal.

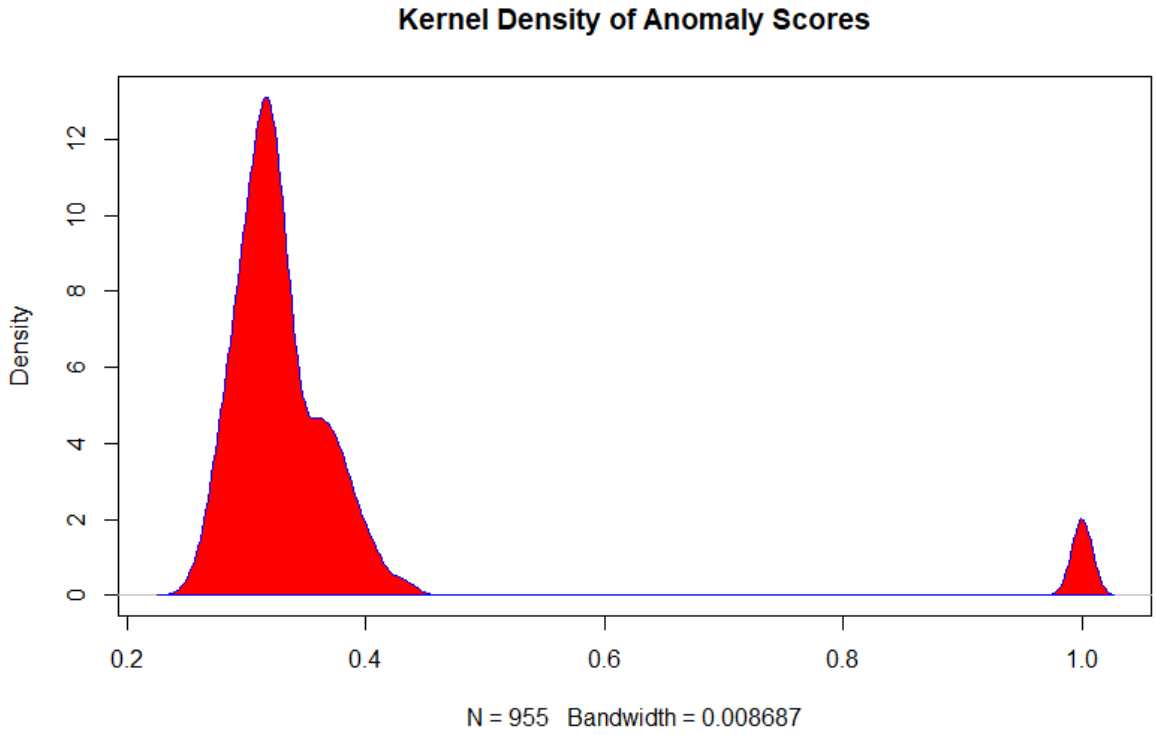


Figure III: Kernel density functions of the ALO approach abnormality scores of each KPI.

9.7 Conclusion

This chapter describes the use of records theory to create two methods. The first method reduces the number of variables in a time series to focus on those that have a significant impact on abnormal behavior. The second method proposes a scoring system for anomaly detection that can be applied in one or multiple dimensions. This system can objectively detect anomalies and suggest threshold values for KPIs without the need for expert input.

The suggested anomaly detection scoring system is a simple algorithm that does not rely on any specific distribution or parameters. It is designed to be used as an online system for detecting anomalies with minimal computational complexity, and it eliminates the risk of overfitting. Additionally, the system can automatically estimate the threshold value needed to classify observations as anomalies, ensuring optimal performance of the algorithm. Furthermore, the algorithm was tested on real-world telecommunications data, and it demonstrated excellent performance in detecting anomalies with very low error rates.

One possible application of this work is to conduct a more in-depth analysis of the anomaly scores in order to extract information about the underlying causes of the anomalies. Another potential direction is to explore the probabilistic properties of the different anomaly scores generated by the system, using records theory as a basis for analysis. This approach could be informed by the research conducted by Hoayek and Ducharme in 2017.

References

[1] AKOGLU, L., TONG, H., AND KOUTRA, D. Graph based anomaly detection and description: a survey. Data Mining and Knowledge Discovery 29 (2015), 626–688.

[2] ALI, W. A., MANASA, K., BENDECHACHE, M., FADHEL ALJUNAID, M., AND SANDHYA, P. A review of current machine learning approaches for anomaly detection in network traffic. Journal of Telecommunications and the Digital Economy 8, 4 (2020), 64–95. [3] ALVI, A. M., SIULY, S., AND WANG, H. Developing a deep learning-based approach for anomalies detection from EEG data. In Web Information Systems Engineering–WISE 2021: 22nd International Conference on Web Information Systems Engineering, WISE 2021, Melbourne, VIC, Australia, October 26–29, 2021, Proceedings, Part I (2022), Springer, pp. 591–602.

[4] ARNOLD, B. C., BALAKRISHNAN, N., AND NAGARAJA, H. N. Records. John Wiley & Sons, 2011.

[5] BALLERINI, R., AND RESNICK, S. Records from improving populations. Journal of Applied Probability 22, 3 (1985), 487–502.

[6] CHANDLER, K. The distribution and frequency of record values. Journal of the Royal Statistical Society: Series B (Methodological) 14, 2 (1952), 220–228.

[7] CHANDOLA, V., BANERJEE, A., AND KUMAR, V. Anomaly detection: A survey. ACM computing surveys (CSUR) 41, 3 (2009), 1–58.

[8] HAMIE, H., HOAYEK, A., AND AUER, H. Modeling the price dynamics of three different gas markets-records theory. Energy Strategy Reviews 21 (2018), 121–129.

[9] HOAYEK, A. S., DUCHARME, G. R., AND KHRAIBANI, Z. Distribution-free inference in record series. Extremes 20, 3 (2017), 585– 603.

[10] HUANG, Z., KANG, X., LI, S., AND HAO, Q. Game theory-based hyperspectral anomaly detection. IEEE Transactions on Geoscience and Remote Sensing 58, 4 (2019), 2965–2976. [11] JABBOUR, C., HOAYEK, A., MAUREL, P., KHRAIBANI, Z., AND GHALAYINI, L. Examining satellite images market stability using the records theory: Evidence from French spatial data infrastructures. Journal of Spatial Information Science, 22 (2021), 61–82.

[12] KHRAIBANI, Z., JACOB, C., DUCROT, C., CHARRAS-GARRIDO, M., AND SALA, C. A non-parametric exact test based on the number of records for an early detection of emerging events: illustration in epidemiology. Communications in Statistics-Theory and Methods 44, 4 (2015), 726–749.

[13] LJUNG, G. M., AND BOX, G. E. On a measure of lack of fit in time series models. Biometrika 65, 2 (1978), 297–303.

[14] MAJUMDAR, S. N., AND ZIFF, R. M. Universal record statistics of random walks and lévy flights. Physical Review Letters 101, 5 (2008), 050601.

[15] NEVZOROV, V. B. Records. Theory of Probability & Its Applications 32, 2 (1988), 201–228.

[16] NEVZOROV, V. B. Records: mathematical theory. American Mathematical Soc., 2001. [17] RASHID, A. B., AHMED, M., SIKOS, L. F., AND HASKELLDOWLAND, P. Anomaly detection in cybersecurity datasets via cooperative co-evolution-based feature selection. ACM Transactions on Management Information Systems (TMIS) 13, 3 (2022), 1–39.

[18] ŠABIC´, E., KEELEY, D., HENDERSON, B., AND NANNEMANN, S. Healthcare and anomaly detection: using machine learning to predict anomalies in heart rate data. AI & SOCIETY 36, 1 (2021), 149–158.

[19] SALTON, G., WONG, A., AND YANG, C.-S. A vector space model for automatic indexing. Communications of the ACM 18, 11 (1975), 613–620.

[20] SHA, W., ZHU, Y., CHEN, M., AND HUANG, T. Statistical learning for anomaly detection in cloud server systems: A multi-order Markov chain framework. IEEE Transactions on Cloud Computing 6, 2 (2015), 401–413.

[21] VANGIPURAM, R., GUNUPUDI, R. K., PULIGADDA, V. K., AND VINJAMURI, J. A machine learning approach for imputation and anomaly detection in IoT environment. Expert Systems 37, 5 (2020), e12556.

[22] WERGEN, G. Records in stochastic processes theory and applications. Journal of Physics A: Mathematical and Theoretical 46, 22 (2013), 223001.

[23] WERGEN, G., AND KRUG, J. Record-breaking temperatures reveal a warming climate. EPL (Europhysics Letters) 92, 3 (2010), 30008.

[24] ZHOU, X., HU, Y., LIANG, W., MA, J., AND JIN, Q. Variational LSTM enhanced anomaly detection for industrial big data. IEE

Chapter 10: Conclusion and research perspective

In this thesis, we have investigated the existing anomaly detection algorithms and how we can contribute by developing new methodologies to improve performance and accuracy of detecting anomalies applied to time series data generated from telco networks. Through a series of deep mathematical and analytical research followed by multiple experiments on the data, we have made the following contributions to the field:

1. Our research introduces a novel geometrical multidimensional probabilistic model for detecting abnormal behavior in data, generating anomaly scores, and quantifying anomaly drivers. This innovative approach enables more effective determination of the root causes of detected anomalies. We also propose a data-driven outlier scoring system to prioritize device anomalies and tackle data observations with a higher likelihood of being outliers. To enhance real-time performance, our model incorporates a sampling approach that accelerates scoring of new observations. When tested on real-world datasets and compared to conventional AD methods, our new model demonstrates improved efficiency in detecting anomalies and identifying their drivers. Telecommunication network experts have validated the effectiveness of our approach, highlighting its potential for significantly enhancing network maintenance and management.

2. We present a novel algorithm that extracts four features from historical alarms data and aggregates them to generate an optimized final score, informed by supervised labels for enhanced accuracy. These features capture the likelihood of event occurrence, event sequences, and the significance of relatively novel events not previously observed in the data. Prior to implementation, we test certain assumptions on the data using appropriate statistical tests to ensure validity. Our evaluation on labeled data demonstrates the high anomaly detection accuracy of our proposed algorithm. This innovative approach offers a valuable tool for NOC teams to efficiently process and analyze alarms data, enabling them to prioritize critical events and take timely action to address network anomalies.

3. We present a novel algorithm that serves as a pipeline for preprocessing textual logs data, grouping it into patterns, and dynamically labeling each pattern as anomalous or non-anomalous. This innovative approach offers users and experts continuous, real-time logs monitoring capabilities, facilitating the detection of anomalies and system failures that could impact network

performance. By applying our algorithm to real-world data, we demonstrate its effectiveness in processing and analyzing logs data for timely anomaly detection, ultimately providing a more robust and adaptable solution for network performance monitoring in an evolving technological landscape.

4. We proposed a novel anomaly scoring system based on records theory, offering multiple advantages over current state-of-the-art models. Our research makes a significant contribution by presenting an innovative approach to anomaly detection and scoring that focuses on tail behavior, leading to more efficient and accurate network maintenance decision-making. The effectiveness of this approach is showcased through its application to a real-world dataset, highlighting its practical implications and potential for enhancing network performance management.

Research Perspectives and Future Work

This research has successfully demonstrated the effectiveness of various advanced anomaly detection models for efficient network performance management in telecommunication networks. While these models have shown promise in addressing the challenges of network maintenance and prioritization, there remain several potential areas for future exploration and development:

- Integration of multiple data sources: Investigate the potential benefits of combining alarms data, logs data, and KPIs to develop a more comprehensive and robust anomaly detection model, aggregating the individual scores obtained in the different chapters and that leverages the strengths of each data type.

- Adaptation to emerging technologies: As new communication technologies and network architectures continue to evolve, it will be essential to adapt and refine the proposed models to accommodate the unique challenges posed by these innovations.

- Real-time processing and scalability: Explore strategies for optimizing the proposed algorithms' real-time processing capabilities to ensure their performance scales effectively with the ever-growing size and complexity of telecommunication networks.

- Automated root cause analysis: Develop methods for automatically identifying the root causes of detected anomalies to further streamline network maintenance processes and accelerate response times.

- Advanced visualization techniques: Implement advanced visualization tools and techniques to improve the interpretability of the results generated by the proposed models, enabling network engineers to identify and prioritize issues requiring attention more easily.

By addressing these challenges and building upon the existing research, future work in this area can further advance the field of network performance management and support the ongoing development and maintenance of increasingly complex telecommunication networks.

In conclusion, this thesis has made significant contributions to the understanding of anomaly detection algorithms, while also highlighting areas for future research.

École des Mines de Saint Etienne

Thèse préparé par **Michel Kamel** 2021 – 2023

Mots clefs: Détection d'anomalies, modèle géométrique multidimensionnel, théorie des records.

Résumé:

La croissance exponentielle des réseaux de dispositifs connectés dans le monde entier signifie que les opérateurs de télécommunications ont besoin de systèmes intelligents et performants pour aider à maintenir leurs réseaux vastes et complexes. Pour répondre aux limites des modèles de détection d'anomalies (DA) les plus populaires, les auteurs proposent un nouveau modèle géométrique multidimensionnel probabiliste pour rechercher les comportements anormaux dans l'espace de données, générer des scores d'anomalie et quantifier les facteurs d'anomalie. Ils introduisent également un algorithme pour générer un score final basé sur quatre caractéristiques dérivées des données historiques pour les données d'alarme. En outre, ils présentent un algorithme pour aider à prétraiter les données textuelles, les regrouper en classes et étiqueter dynamiquement chaque classe comme une anomalie ou non. Enfin, ils proposent une méthode qui réduit la dimensionnalité et propose un système de score d'anomalies basé sur la théorie des records. Dans l'ensemble, leurs recherches fournissent des méthodes innovantes pour détecter et prioriser les anomalies dans les réseaux de télécommunications et fournir des outils puissants pour l'analyse de données et la maintenance du réseau.