Czech Technical University in Prague

Faculty of Electrical Engineering

Doctoral Thesis

February 2024 *Ing. Zagroz Aziz*

Czech Technical University in Prague

Faculty of Electrical Engineering
Department Telecommunication Engineering

Machine Learning Models for Pattern Identification and Predictive Analytics

Doctoral Thesis

Ing. Zagroz Aziz

Prague, *(February 2024)*

Ph.D. Programme: P2612 Electrical Engineering and Information Technology
Branch of study: 2601V013 Telecommunication Engineering

Supervisor: ***Ing. Robert Bestak, Ph.D.***

Declaration

I hereby declare that this doctoral thesis has been composed and accomplished solely and independently by myself in accordance with the Guideline on ethical preparation of university final theses. This work has not been submitted, in whole or in part, to any previous application for any other degree. Except where stated otherwise by reference or acknowledgment, the presented work is entirely my own.

February 2024 ....
Prague, Czechia Signature

Acknowledgements

Everything is possible when you have the right people around to lean on. This work is outcome of the support and endeavor of many people, to whom I'm eternally grateful. First and foremost, it feels like words aren't enough to acknowledge and express my gratitude to my supervisor Ing. Robert Bestak, Ph.D. for his significant support, contributions and insights throughout the years, without him this work would not have been possible.

To my father, my mentor, and my hero.. he will always be in my heart.. I am who I am now because of him, and I am proud of what he has built, and left behind...

To my mother, the beauty and peace of my life, and like they say ”Life doesn't come with a manual, it comes with a mother.”

I appreciate all the support as well received from our beloved university, faculty and department. I am so grateful to be part of this community.

Last, but not least, I would like also to thank my brother, sisters and friends who were always there for me with their unwavering support, motivation and encouragement.

Abstract

In the last decades, the amount of data generated has seen exponential growth. The rapid growth of data led the data mining techniques to play a significant role in establishing baselines, benchmarks, objectives, analyzing and better understanding concepts. However, mining useful information in data requires relevant techniques and procedures. In contrast, the process of pattern identification in the raw data entities becomes more challenging. Patterns are the key to understanding, analyzing, predicting and decision making. On the other hand, finding patterns needs first data analysis, machine learning, artificial intelligence, and statistical models. This process is called predictive analytics, which is the process of employing data to forecast future outcomes. In this thesis, different novel case studies have been introduced using different datasets. Each dataset has gone through preprocessing. Data preprocessing includes data extraction, collection, profiling, reduction, wrangling, enrichment and validation before being studied. Every case study introduces certain problems and provides solutions, brings in better analytical understanding, system enhancement, outcomes and accuracy improvement. In addition, several statistical and machine learning algorithms and models have been employed to address the trends and patterns in the data. Furthermore, we provide a statistical analysis using different methodologies to identify traffic patterns, indicate the network performance and quality of service. Moreover, we projected anomaly behavior, detection and prediction that helps network operators to understand and forecast such network behaviors. All these studies have been evaluated and assessed using several performance metrics and parameters.

Keywords: data mining, machine learning, Call Detail Record, Rainfall forecasting, mobile network, statistical models, pattern identification, predictive analytics.

Abstrakt

V posledních desetiletích zaznamenalo množství generovaných dat exponenciální růst. Rychlý růst dat vedl techniky dolování dat k tomu, aby hrály významnou roli při stanovování základních linií, měřítek, cílů, analýzy a lepšího porozumění konceptům. Získávání užitečných informací v datech však vyžaduje příslušné techniky a postupy. Naproti tomu proces identifikace vzoru v entitách nezpracovaných dat se stává náročnější. Vzorce jsou klíčem k porozumění, analýze, předpovídání a rozhodování. Na druhou stranu, hledání vzorců vyžaduje nejprve analýzu dat, strojové učení, umělou inteligenci a statistické modely. Tento proces se nazývá prediktivní analytika, což je proces využívání dat k předpovídání budoucích výsledků. V této práci byly představeny různé nové případové studie s použitím různých datových sad. Každá datová sada prošla předzpracováním. Předzpracování dat zahrnuje extrakci dat, sběr, profilování, redukci, hádky, obohacení a validaci před tím, než jsou studována. Každá případová studie představuje určité problémy a poskytuje řešení, přináší lepší analytické porozumění, vylepšení systému, zlepšení výsledků a přesnosti. Kromě toho bylo k řešení trendů a vzorců v datech použito několik statistických algoritmů a modelů a algoritmů strojového učení. Dále poskytujeme statistickou analýzu pomocí různých metodologií k identifikaci vzorců provozu, indikaci výkonu sítě a kvality služeb. Kromě toho jsme navrhli chování, detekci a predikci anomálií, které pomáhají provozovatelům sítí porozumět a předvídat takové chování sítě. Všechny tyto studie byly vyhodnoceny a posouzeny pomocí několika výkonnostních metrik a parametrů.

Klíčová slova: dolování dat, strojové učení, záznam podrobností o hovoru, předpověd̆ srážek, mobilní sít̆, statistické modely, identifikace vzorů, prediktivní analytika.

Acronyms

2G Second generation
3G Third generation
4G Fourth generation
5G Fifth generation
ACF Autocorrelation Function
ADSL Asymmetric Digital Subscriber Line
AI Artificial Inteligence
AIC Akaike Information Criterion
ANN Artificial neural network
AR Auto-Regressive
ARIMA Autoregressive Integrated Moving Average
ARMA Auto-Regressive Moving Average
AUC Area Under ROC Curve
BTQ Bits through Queues
CDF Cumulative Distribution Function
CDR Call Detail Record
CM Climate Model
CNN Convolutional Neural Network
CSMA Carrier Sensing Multiple Access
DBMS Database Management System
DBSCAN Density-Based Spatial Clustering of Applications with Noise
DT Decision Tree
EM Expectation Maximization
ENN Evolving Neural Network
FE Forecast Error
FN False Negative
FP False Positive
FPR False-Positive Rate

GA Genetic Algorithm
GAN Generative Adversarial Networks
GMM Gaussian Mixture Models
GPS Global Positioning System
GSM Global System for Mobile communication
HCLM Hybrid Climate Learning Model
HMM Hidden Markov Models
HTTP Hypertext Transfer Protocol
ICI Inter-Cell Interference
IEEE Institute of Electric and Electronics Engineers
IMS IP Multimedia Subsystem
IP Internet Protocol
ITU International Telecommunication Union
KDE Kernel Density Estimation
LDA Latent Dirichlet Allocation
LSTM Long Short-Term Memory
LTE Long Term Evolution
MA Moving Average
MAC Media Access Control
MAE Mean Absolute Error
MAPE Mean Absolute Percent Error
MFE Mean Forecast Error
MGW Media Gateway
MLE Maximum Likelihood Estimation
MMD Maximum Mean Discrepancy
MMPP Markov Modulated Poisson Process
MS Mean Shift
MSE Mean Square Error
NLP Natural Language Processing
PACF Partial Autocorrelation Function
PMLP Probabilistic Multilayer Perceptron
RBFNN Radial Basis Function Neural Network
RCS Rich Communication Services
RFA Random Forest Algorithm
RMSE Root Mean Square Error
ROC Receiver Operating Characteristic curve
SARIMA Seasonal Autoregressive Integrated Moving Average
SIP Session Initiation Protocol

SMS Short Message Service
SSW Soft Switch
SVR Support Vector Regression
TDM Time-division multiplexing
TIM Time Interpolation Method
TN True Negative
TP True Positives
TPR True-Positive Rate
TSDB Time Serious Database
UE User Equipment
VoIP Voice over IP

List of Figures

List of Tables

Contents

Chapter 1

Introduction

1.1 Thesis overview

The interaction between data and Artificial Intelligence (AI) provides the infrastructure of the most sophisticated systems nowadays. AI builds models through machine learning using the generated data from the systems. Machine learning is considered a subset of AI that requires often large amounts of data to accurately identify patterns, make predictions, and continuously refine the models. The quality and quantity of the data directly impact the effectiveness of machine learning applications. This symbiotic relationship is often referred to as the "training" phase, where a model learns from historical data to make predictions on new, unseen data. The combination of data and machine learning has transformative implications across various industries. From predictive analytics in finance, mobile networks or meteorological data to image recognition in healthcare, these technologies are revolutionizing how organizations extract insights, automate decision-making processes, and enhance overall efficiency. In a nutshell, data are the lifeblood that fuel machine learning algorithms, and machine learning is the engine that extracts meaningful insights and knowledge from the vast sea of data. Together, they form the backbone of contemporary artificial intelligence applications, driving innovation and advancement in numerous fields. The major contribution to my data comes from mobile networks along with the other types of datasets such as precipitation, population, distance, etc. I introduce and explain briefly the environment/structure of each contribution in my study in the followings:

1.1.1 Mobile Networks

Mobile networks, also known as cellular networks, are a crucial part of modern telecommunications systems that allow mobile devices, such as smartphones and tablets, to communicate wirelessly with one another and access various services. These networks provide the infrastructure for voice and data communication over long distances, through base stations. Nevertheless, communication can be held internationally either using satellite systems or optical fiber (transoceanic or underground cabling).

Mobile networks consist of various elements, including switching centers denoted as core network, air interface or access system, and data centers (Fig. 1.1 shows a simple mobile network architecture) [1]. These elements work together to route calls and data between users and services. These calls are recorded and stored as CDR in the CDR database that is located at the core network. For any call establishment, management, and clearing, a signaling is exchanged among the network elements. The signaling is responsible for controlling calls, managing the network, monitoring the performance and network status, network services, etc. [2].

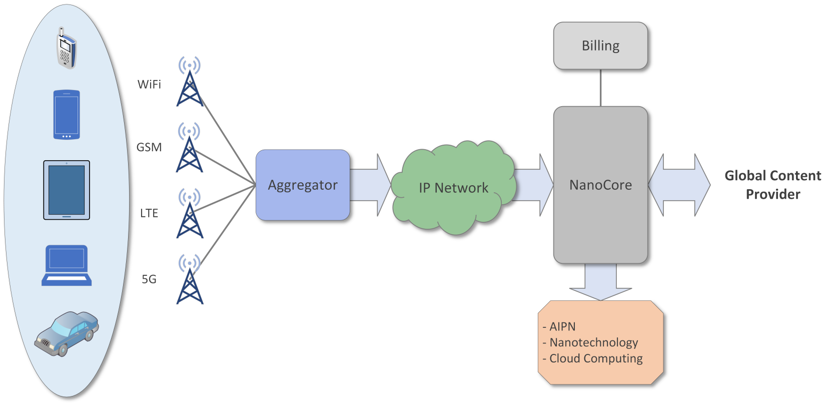


Figure 1.1: International voice call flow through a carrier network.

Mobile networks use specific radio frequency bands allocated by regulatory authorities. Different frequency bands are used for various purposes, such as voice communication (1G, 2G, 3G), and data in case of Long-Term Evolution (LTE), 4G, 5G, and more. This technology has evolved through generations offering improvements in speed, capacity, capabilities and features. Common mobile network generations include 2G (second generation), 3G (third generation), 4G (fourth generation), and 5G (fifth generation) [3]. Each generation has different features and specifications and 5G is the latest generation deployed with higher capabilities in terms of bandwidth, user mobility, data transfer (upload, download), better performance, etc. (Fig. 1.2 shows the evolution of mobile networks).

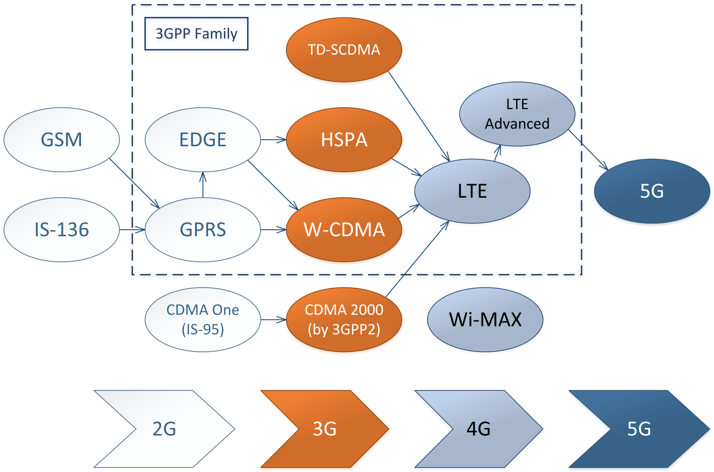


Figure 1.2: Evolution of mobile networks.

1.1.2 Machine Learning

Machine learning is a subset of artificial intelligence that focuses on developing algorithms and models that enable computers to learn from data and improve their performance on a specific task over time. One of the major advantages of machine learning is that it grows better and improves with experience [4]. Machine learning is generally a common tool to use to conduct data mining, pattern identification and predictive analytics.

Data mining is introduced as a result of the natural evolution of information technology. It is the process of discovering and extracting knowledge out of large amounts of data [5]. The data source is usually one of the most important parts in data mining as it provides a full understanding of the data. In general, we might be in need to combine multiple data sources to fortify a goal. First, the data should be cleaned from irrelevant and inconsistent arrays as it is illustrated in Fig. 1.3 that shows the process of mining useful information from mobile networks data along with other sources of data.

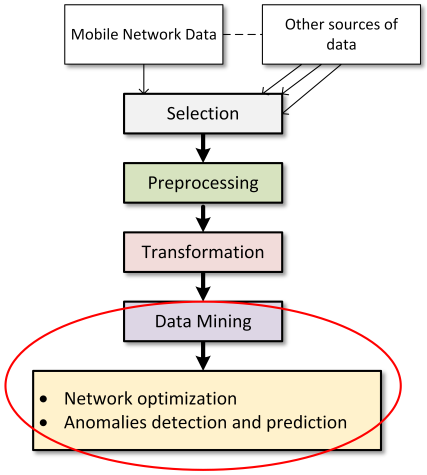


Figure 1.3: Data Mining Process.

The selection of data is made based on the objectives of mining. Then the data are transformed and merged into appropriate formulae for mining, which is a crucial step for retrieving the required data patterns. Such patterns in my study are later used to optimize the system and other data sources. Once the data optimization is done to find anomalies and ready for prediction and forecasting, further work is done like patterns identification, predictive analytics or predictions from the analyzed data.

We often talk about patterns when it comes to mining data or employing machine learning, but we probably know a simple definition of it. Patterns can be found everywhere and in every aspect around the globe. Beginning with the patterns on our clothes, through maps, streets, buildings, products to technologies. However, the usual question that comes up is, what does pattern mean? Almost comprises everything or anything, and how the technology is implementing it for everyday use? The answer is quite simple for those who think of a long and complicated answer. Somehow back when we were kids used to receive tasks with identifying a sequence of numbers or to join dots to complete a figure. The expected figure or sequence of numbers to complete the task is the pattern identification in machine learning.

The process or method in identifying a trend or regularities in data using machine learning algorithms is called pattern identification or recognition [6]. These patterns can be recognized mathematically or empirically using the proper tools like algorithms. The major three components of identifying any pattern come to feature extraction, classification, and clustering [7]. Feature extraction is about extracting useful information out of raw data like specific attributes or characters in the raw data for certain tasks or objectives. Classification or supervised learning with labeled dataset is the process to classify elements into various categories. It divides the data into training set and test set, using the training set to teach the model and the test set to examine the model. In return, clustering is a method to group items in a dataset with similar traits. On the other hand, predictive analytics is the process of using past and present data to forecast futuristic values and outcomes. That includes data analysis, statistical models, artificial intelligence and machine learning to explore patterns and identify correlations for future behavior predictions.

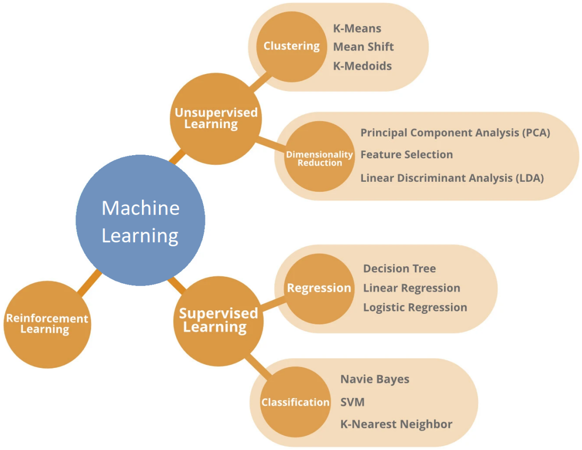


Figure 1.4: Machine Learning types.

Machine learning applications are categorized into four major groups (Fig. 1.4 shows examples for each category) [8]:

1. Supervised learning - it is a task of learning, which maps an input to an output with data being labeled and expected output. Classification and regression are considered supervised learning problems.

2. Unsupervised learning – includes algorithms that come with finding the relationship between the input and output from unlabeled datasets and no supervision. Clustering is one of the examples of unsupervised learning.

3. Semi-supervised learning – it is a combination of supervised and unsupervised learning. This technique uses a small amount of labeled data and plenty of unlabeled data for predictive model training. It is about merging clustering and classification algorithms.

4. Reinforcement Learning – it is about finding the optimal behavior in an environment to achieve the maximum reward. It is like having an agent trained based on reward and punishment strategies.

1.1.3 Data Structure

Data refers to a collection of individual facts, statistics, or information that can be in various forms, such as numbers, text, images, sounds, or any other representations of facts or ideas. The singular form of data is called datum. In general, data are raw and unprocessed and typically lacks context or meaning on its own. It becomes valuable when it is collected, organized, and analyzed to derive insights and make informed decisions. Data are generated every day from various sources and systems such as mobile networks, sensors, meteorological systems, satellites, finance, etc. (see Fig. 1.5).

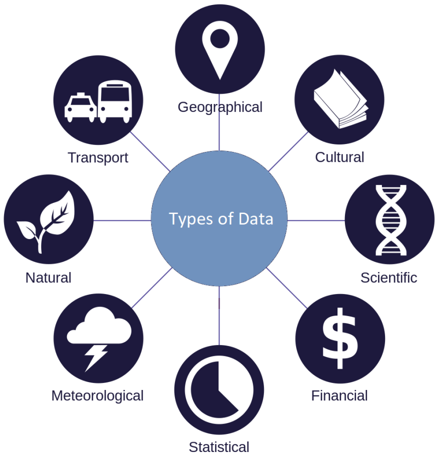


Figure 1.5: Types of data.

There are two main types of data:

1. Structured Data: This type of data is highly organized and typically fits into predefined formats [9]. Structured data is often found in databases,

spreadsheets, and tables. Examples include numerical values, dates, and categorical data. It's easy to process and analyze, making it suitable for many data science and analysis tasks.

2. Unstructured Data: Unstructured data lacks a specific format or structure [10]. It can be textual, such as emails, social media posts, or documents, as well as non-textual, including images, videos, and audio recordings. Unstructured data requires more advanced techniques, like natural language processing (NLP) and computer vision, to extract meaningful information from it. In my thesis, the majority of the research is using unstructured data or about 90%.

Data are the foundation of many fields, including data science, analytics, and business intelligence. They play a crucial role in decision-making, research, and problem-solving across various industries. The process of collecting, cleaning, analyzing, and interpreting data is essential to turn raw data into actionable insights and knowledge.

In this thesis, there are several types and sources of datasets being used. However, they needed to be collected, prepared and preprocessed. Beside the mobile networks data (both traffic and signaling), there are other data types used in my study such as population of countries, distance among countries, great circle distance, rainfall, areas and centroids of countries:

- Population data: They refer to information and statistics related to a specific group of people living in a particular geographic area or country. The data that have been collected per each country are back to 2016. Population data is vital for various fields, including demography, sociology, economics, public policy, healthcare, and urban planning, as it helps researchers, policymakers, and businesses make informed decisions and understand the dynamics of a population.

- Great-circle distance: It is also known as the orthodromic distance, is the shortest distance between two points on the surface of a sphere [11]. It is most commonly used to measure distances on the Earth's surface, as the Earth is approximately spherical in shape (see Fig. 1.6). The great-circle distance is determined along with the arc of the great circle that connects the two points, rather than along with a straight line, because the Earth's surface is curved. I use the haversine formula that takes into account the latitude and longitude coordinates of the two points and calculates the arc length of the circle that connects them on the sphere's surface [12]. I use great-circle distance to measure the distance between one reference point/country to the rest of countries around the globe.

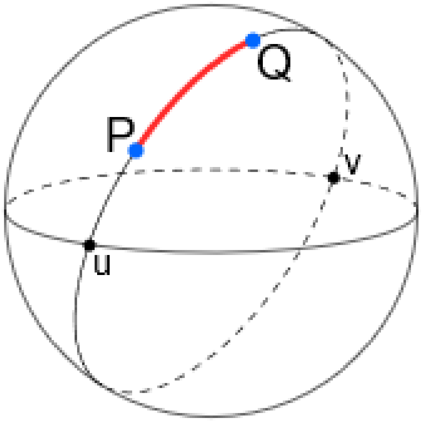


Figure 1.6: Great-circle distance.

- Centroid of countries: A centroid is a geometric concept that represents the center or the mean of a set of points, whether in a two-dimensional or three-dimensional space. The countries around the globe are geographically and geometrically in irregular shapes. This leads to a challenge when it comes to calculating the distance between two countries since any point taken on the area of a country would make a difference in distance calculation. However, I use centroids of countries to overcome this obstacle (see Fig. 1.7). The centroid is calculated using First Moment Integral, which is a mathematical approach to calculate the coordinates of the centroid or center of mass of that shape [13].

- Rainfall data: It is the amount of precipitation, specifically rain, that falls over a particular area during a specified period of time measured in (mm). It is an essential component of weather and climate data that plays a crucial role in various fields, such as meteorology, hydrology, agriculture, environmental science, and urban planning. Rainfall data provide insights into precipitation patterns, helps in monitoring and

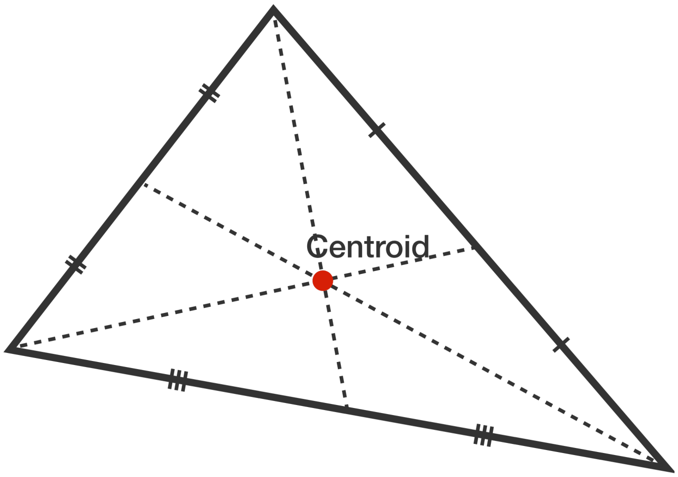


Figure 1.7: Centroid.

predicting weather conditions, and supports in decision-making in a wide range of applications.

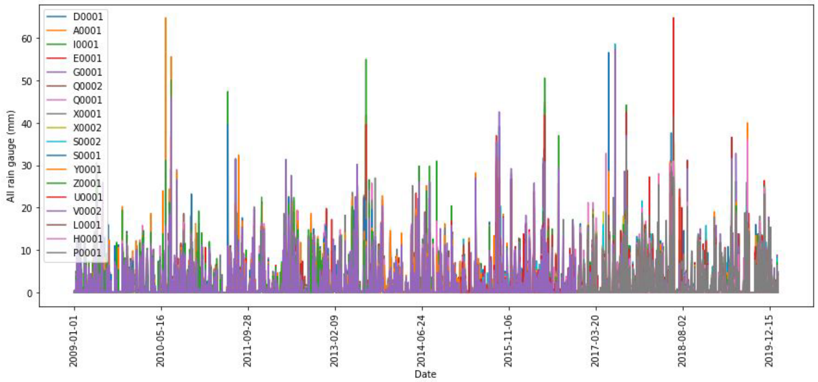


Figure 1.8: Rainfall data from the rain gauges.

Rainfall data are collected using instruments called rain gauges, which can be of various types, including standard rain gauges, tipping bucket rain gauges, and radar systems. There are many research centers that distribute rain gauges over a specific region or location to measure the amount of rain and use the accumulated data to forecast the weather as shown in Fig. 3.9 from one of my studies on rainfall forecasting using pluviometer/rain gauges. However, there are additional parameters along with the rain gauges that can be utilized to make the prediction more accurate such as clouds using image recognition, temperature, wind speed and direction, humidity, etc.

1.2 Motivation

Over the last few decades, data have gradually increased in volume, value, variety, velocity, and veracity. In return, management and preprocessing of big data became more challenging and tough to deal with. Besides, it is quite crucial and vital to provide complex and comprehensive analytics in this complicated world. Nonetheless, data analytics is often a complex process to uncover knowledge like hidden patterns, correlations, market trends and customer preferences. More importantly, studying the environment of the data source may help in providing results with higher accuracy and efficiency.

Uncovering hidden patterns often necessitates an understanding of the object behavior. However, meticulously selecting the appropriate statistical algorithms and predictive models for a specific study case demands significant effort and time.

The process of achieving accurate predictive analytics to forecast future outcomes needs reliable and trustworthy data sources along with statistical algorithms and machine learning techniques. In addition, it is essential to choose the algorithms and models based on several performance and evaluation parameters. Every significant study initially provides a concrete analysis of the available datasets before employing the models. However, many times authors ignore theoretical approaches and deep analysis, which makes huge difference in understanding data structure and thus affecting the final conclusions.

There are three major difficulties/problems that can affect any study. Firstly, the availability of the data since many data sources nowadays tend to be inaccessible for privacy reasons that firms and companies hide from the public. The second problem is the size of the available data as less data mean less accurate studies, and finally the number of attributes that the data provide sometimes lead to poor analysis, prediction and modeling.

Data can consist of a lot of outliers and trivial information in any system that might mislead the researchers. In contrast, anomaly detection can be part of the objective since it is about certain conditions that occur due to specific events and generally do not correspond to a well-defined notion of normal behavior. Therefore, it is important to differentiate between a trivial outlier and an anomaly that occurs due to an event. It is quite crucial to detect such anomalies in a trice since they have negative impacts on the system. There are several available techniques to handle such anomalies in terms of detection and prediction, but it is necessary to understand and study first the normal profile and image of the system.

1.3 Aim of the thesis

My thesis deals with pattern identification from several sources of data and providing predictive analytics. The main part of the thesis focuses on mobile networks data with the latest network generation deployment. The aim is to identify the user mobility and behavior in the network to detect and predict the anomalies. With the population growth followed by network expansion and development to contain the needs of users, more data are generated at enormous speed. That makes understanding and identifying the normal pattern in any data hard and challenging, more importantly, to detect and predict anomalies, especially with several data sources combined. Thus, one of the aims is to build and develop models to extinguish a normal behavior first and then be able to detect and predict abnormal events. My thesis can be summarized but not limited to the following contributions:

1. Statistical analysis and deep understanding of voice traffic profiles in mobile networks.

2. Dependency between distance and voice traffic in international voice traffic.

3. Influence of neighboring countries on wholesale voice termination.

4. Impacts of other related data sources combined with mobile networks data.

5. Queuing theory approach on call arrivals and voice traffic distribution.

6. Empirical and mathematical study of interarrival, service and waiting time.

7. Modeling voice traffic profiles to identify user patterns in mobile networks.

8. Building models for peak hours and call duration profiles identification.

9. Building four models for anomaly detection and prediction.

10. Time series models used for weather forecasting and influence of historical data.

1.4 Organization of the thesis

This section describes the organization of the thesis.

Chapter 2 – State of the art provides an insight into the recent studies that have been published in the area of the thesis. I deliver a comprehensive overview of the challenges and limitations along the methodologies and approaches being used. In section 2.1, I discuss the aims of exploitation of mobile network data. Section 2.2 describes the available techniques for data analytics and the present frameworks for data preprocessing. Then, I depict the data characteristics and their applications in section 2.3. I present in section 2.4 the current status of the use of machine learning algorithms and models. The chapter concludes the discussion with time series data forecasting in section 2.5.

Chapter 3 – The data are presented in this chapter including the structure and preprocessing. I start with CDR data and the call flow description, followed by signaling data and the use of population data of the globe for each country. Then, I present distance and centroid data along with the Great-Circle Distance formulas. The last section describes the rainfall data and the structure of the sensors and their locations.

Chapter 4 – Consists of the CDR data analysis of local and international voice traffic, the influence of the neighboring countries, voice traffic patterns per date and time, the changes during the weekends, weekdays and holidays that can have impact on overall traffic. Moreover, the CDR data are used proportionately with their corresponding country population around the globe, and the distance between the reference country and the rest of the world based on centroid of countries utilizing great-circle distance.

Chapter 5 – Proposes a theoretical and practical framework of queueing theory based on the signaling and voice traffic data. The influence of Poisson and Exponential distributions on the traffic characteristics. In addition, the characteristics of interarrival, waiting and service time are explained theoretically as well as empirically.

Chapter 6 – Focuses on using machine learning algorithms and models for pattern identification, anomaly detection and prediction. I use linear regression function, statistical visualization, Gaussian Mixture Models (GMM) and Mean Shift (MS) clustering to understand behavior of users in the mobile network, then I employ Z-score, Isolation Forest, K-means and Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithms. I evaluate each of these algorithms via several performance metrics.

Chapter 7 – Presents a multi-sensory precipitation forecast study through three time series forecasting algorithms. The research studies the rainfall prediction based on historical data from 2009-2021. However, I propose also that 67 percent of the sensors deployed can still deliver similar accuracy in results and performance by relocating and redistributing.

Chapter 8 – Concludes and summarizes the thesis along the research contribution and future works.

Chapter 2

The state of the art

This chapter depicts the current and latest works being presented in this field as well as the limits and challenges. I go through the existing and relevant research topics published. Furthermore, I discuss their structures, the architecture, and the conceptual framework such as the methodology that has been used along with their objectives and capabilities. I aim here to synthesize the research, not summarize it. In general, finding useful information and trends in the data needs analysis and modeling according to the area of study. Thus, the majority of the works have data involvement to serve a certain aim or goal using machine learning algorithms and models.

2.1 Mobile Networks

Recently, there have been done a lot of works in the area of exploitation of mobile networks data for different purposes. That may include network optimization or network statistics, whether the data generated via the core network or the air interface/radio access network. However, there is a good use of user data through the generated data from the smartphones for measurements to allocate radio resources and to adapt parameters of radio link [14], to estimate the location of users using specific tools [15] or certain investigation analysis like crime records investigation [16].

Each network component is responsible of certain tasks, and in return, huge amounts of data generated whether concerning data/voice traffic data such as CDR, real-time statistical reports, monitoring systems or signaling information for various services to have the network works as one entity

(Fig. 2.1 illustrates the components of the mobile networks with the latest network generations deployment) [17].

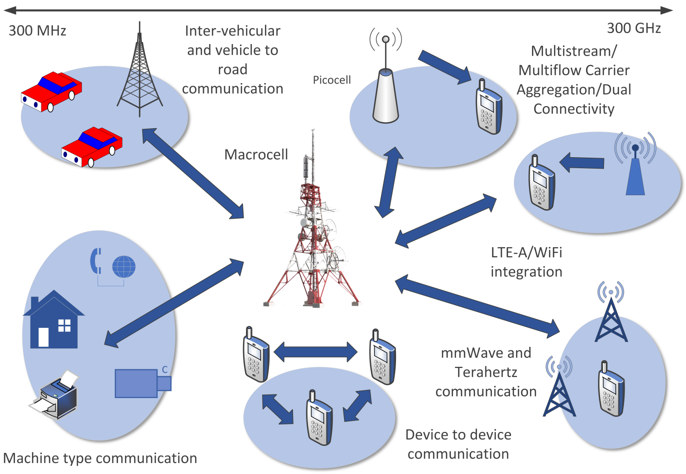

300 MHz
300 GHz
Inter-vehicular and vehicle to road communication
Picocell
Multistream/ Multiflow Carrier Aggregation/Dual Connectivity
Macrocell
LTE-A/WiFi integration
mmWave and Terahertz communication
Device to device communication
Machine type communication

Figure 2.1: Mobile network services.

Analyzing such data either through using various tools or machine learning models [18][19], is crucial to define many profile categories such as user behaviors through voice calls and data usage for different date and time profiles.

In contrast, the system needs to be evaluated regarding reliability, survivability and quality of service, thus it is essential to classify the network usability rate profiles [20] and introduce characteristics of mobile network behavior. The longer the period is covered in the data, the more accurate results are achieved. Though, there are usually big chunks of useless lines coming with every collected data. Thus, there are new techniques and strategies introduced every year to reduce the amount of data through filtering out trivial information and cleaning the data from unwanted/undesired data [21]. These techniques and strategies are based on either software tools, machine learning models, or network built-in applications.

2.2 Data analytics and insights

In the last decade, the role of data for different sorts of purposes is gradually increasing. They are mainly implemented to analyze and optimize the network performance (internet data, user data, radio access, core networks, charging and billing systems, etc.), or to evaluate the impacts of implemented company policies. Nevertheless, authors often present a framework first to process large and complex raw CDR datasets [22]. The suggested framework is then applied to efficiently manage raw data to help improve communication service provider data management and processing.

There are systems, which are built to analyze mobile network metadata. For instance, exerting CDR data and Internet Protocol Detail Records to detect potential criminal activities over end-to-end encryption messaging such as WhatsApp, Facebook messenger, etc. [23]. In return, the data cleaning process would be one of the essential steps for this sort of analysis, especially for large datasets. One of the works includes 2.5 million CDRs to filter anomalies is proposed in [24]. The proposal uses social network analysis to analyze behaviors and relationships between customers through their calling profiles. It is also possible to utilize Hadoop based mobile big data [25] processing platform to find out user mobility behaviors.

The analyzed data usually can go up to hundreds of gigabytes in size. There are times a network can be partially analyzed including only a specific region or location, especially in large, populated areas. This can help to improve, evaluate or analyze the network performance or diagnose for a particular issue only affecting certain areas, like a study was recently presented [26] based on CDR dataset of 27604 mobile network base stations from 75 zones in Beijing. The research proposes the use of CDR data for traffic distribution and location update analysis. Likewise, there is also the possibility to concentrate only on a particular group or class of subscribers to study their profiles, behaviors and activities in the network [27]. Furthermore, there are developed schemes to store CDRs in a data warehouse and process them to categorize user profiles to understand customers' behaviors according to marketing offers (i.e.: offering special rates, deals, bonus, etc.) per a region or location. That can include also time patterns such as daily, weekly [28] or monthly to identify human mobility based on classifications of user profiles.

2.3 Data characteristics and applications

Data characteristics and their applications are main topics of interest nowadays in the data community. New technologies always lead to exponential growth of the amount of carried out data. There are many case studies that have been recently presented such as queueing theory applications, data characteristics and distribution including traffic load characteristics. In recent years, there is use of the lognormal distribution of interarrival time and service time with exponential distribution in the queueing system [29]. The analysis shows the influence of the keep-alive mechanism on the queue process. However, it is declared that the waiting time and the queue length are shorter than what is proposed by 3rd Generation Partnership Project organization (3GPP). In contrast, the discrete time Markovian chain model based has shown promising results [30] to analyze Carrier Sensing Multiple Access (CSMA) mechanism of MAC layer of IEEE 802.15.4 and investigate the interarrival time distribution for network queue model and yet, to detect anomalies in the network. On the other hand, traffic congestion model is one of the case scenarios for computer network environment that can be classified as one of the major queuing issues. There are researches being published [31] using the continuous-time finite-state homogeneous bivariate Markov chain to evaluate the performance of protocols and applications in a network. In general, traffic congestion can occur due to random path delays and packet losses. Similarly, the performance of mobile networks is also possible to be analyzed and evaluated mathematically and numerically using Markov Modulated Poisson Process call arrivals (MMPP) [32]. In return, exponential process model shows significant results for the traffic variance [33] based on analytical expression to estimate the byte loss probability for a single server queue system alongside multifractal traffic arrival.

The problem of modelling voice traffic in mobile networks, considering inter-arrival time and holding time characteristics is usually addressed [34] through defining a mathematical model that complied with experimental data provided by a mobile operator. In contrast, there are cases of using the queuing theory to design a model simulating the Internet Protocol (IP) Multimedia Subsystem (IMS) service, including the Voice over IP (VoIP) [35]. In general, the developed model includes a single server following the Poisson distribution of arrivals and the exponential distribution of service time. Conversely, it is possible to provide a scheme to transmit a message by means of a queue timing channel [36]. The message is encoded in a sequence of additional delays in service time. However, Poisson distribution shows better QoS when it comes to base stations in mobile networks [37]. This is normally done through evaluating a natural class of typical-user service characteristics (including path-loss, interference, signal-to-interference ratio, spectral efficiency). The challenges that researchers usually face is when it comes to traffic characteristics of mobile networks with big data. There are several applications and models for that purpose to help overcome such difficulties. Of course, there are also improved models to handle large data in mobile networks, such as Zipf-like model for large data use [38] to characterize the distributions of traffic, subscribers, and requests among service providers.

2.4 Use of machine learning models

The significance of data and exploitation of inside parameters lies in its role as a key to better optimizing and dimensioning systems. In recent years, there have been many articles published on the use of data to resolve many problems and enhance performance as well as the quality of service. In mobile networks, the need to understand the user behavior in the network is essential for better network accessibility and usability. Many algorithms and models have been offered to improve the network functionality and plans for future expansions and challenges. However, big data in mobile networks is still one of the challenges that need powerful tools and models for analysis and extracting meaningful insights.

I have seen previously gradual increase in employing K-means hierarchical clustering for anomaly detection in the mobile network [39]. The proposed idea is to train the data with and without anomalies in the network. However, it is a hard and long procedure to isolate anomalies, especially in the case of large data. Furthermore, K-means is not among the most efficient algorithms to handle outliers in big data as the algorithm needs a lot of enhancement before being utilized. Similarly, studies [40][41] aim generative adversarial networks (GAN) with long short-term memory (LSTM) neural networks to shape an anomaly detection framework and K-means to authenticate and verify the anomalies with Autoregressive integrated moving average (ARIMA) respectively. The framework utilizes the original dataset without changing the distribution of the data while correlating the user activities and data expansion.

The ability to process big data in real-time is a challenge in terms of volume and velocity, which makes it problematic for many algorithms to handle the characteristics of the data properly. Though, LSTM shows in many cases study promising results. Furthermore, Convolutional Neural Network (CNN) model has shown promising results to outline the anomalous data, which is also an encouraging model for outliers detection and prediction. There is a study [42] lately published based on 100 base stations in the mobile network to provide a framework of a multi-label vector identifying anomalous cell output. The experiment is regionally based to focus on users that might lead to a loss of revenue. One of my studies also deals with anomalies but on a larger scale and for a longer period of time. I similarly suggest that most of the anomalies in the network have reasons. Hence, they might be rather periodic than a one-time occurrence.

Many researchers in recent years have used CDR data to analyze and detect fraudulence in mobile networks. For instance, there is an ideal solution to use Latent Dirichlet Allocation (LDA) to outline users and Maximum Mean Discrepancy (MMD) to assess the distribution of samples to fit roaming fraudsters [43]. According to the results, the proposal appears encouraging, would've been interesting to see the results on a larger scale over a longer period of time though. Nevertheless, validation of the accuracy with different time periods is important since it deals with a very sensitive topic.

In general, mobile networks data can be categorized into user and network-oriented. There are several ways to do so, such as employing Erlang measurement with CDR data to analyze the behavior of base stations during a certain time [44]. One of studies proposes the K-means algorithm to separate daily patterns. This study shows impressive results to understand the influence of nighttime traffic. However, the extension of this study to use other network generations would be remarkable since it is done using only Global System for Mobile Communications (GSM) networks.

Lately, with the growth of population and advances in mobile networks technology, enormous amount of data are being generated on a daily basis, especially in large cities that can reach several million CDRs within a couple of weeks. For example, an enormous amount of data consisting of 800 million CDR records have been studied in [45] to identify 12 weeks profile. The study aims to analyze the geographical profiles of territories. The idea is important for public transportation in addition to urban planning. However, this might be good for short-term mobility rather than long-term. The result can drastically vary for the long-term period as this is a case of user mobility, which I see higher accuracy over a long period of time. With the latest network generation of 5G, authors in [46] introduce the Pseudo Code of Orthogonal Subspace Projection algorithm based on CDR data to minimize the complexity of the classification algorithm, which is needed to obtain key information about network use analysis such as detecting network anomalies, human mobility, or network activities.

The CDR record generally offers several attributes, but only certain attributes might be used more frequently based on the needs of the study. In addition, certain models and algorithms can be seen more frequently utilized. Sometimes, this leads to similar results that might not necessarily be required as it is already considered a duplicate. In [47] input-output hidden Markov models (HMM) to understand travelers' activity patterns were developed from CDR data. The research is done in the San Francisco Bay area. According to the study, the model is applied to the data collected from a network carrier that serves millions of users. The results come from different locational data sources. The advantage of using HMM is to resolve several internal states that are hard to observe, which makes it an optimal approach to use. Yet, validating with three different locational data sources can lead to high accuracy results, but may not be quite practical as all those sources are not always available. In contrast, there are studies applying CDR data for minimizing energy consumption and increased inter-cell interference (ICI) caused by densification in ultra-dense 5G networks [48]. The notion is to develop a scheme to proactively schedule radio resources and small cell sleep cycles producing substantial energy savings and thus reduced ICI. This is achieved without conceding the quality of service. Although the scheme needs regular supervision and updates, the results can be promising and provide outstanding performance in terms of saving energy.

In recent years, there has been growth in the usage of mobile data, especially user-oriented data like CDR for various usages [49][50]. Each study has enhanced the network performance, availability, and quality of experience at certain locations and times or within the entire network for the long term. The available Machine Learning models and algorithms nowadays with access to thousands of sources online made it easier for researchers and enthusiasts to achieve their objectives. In return, I introduce a new approach to detect network congestion and anomalies. A multi-algorithm approach where each algorithm targets a specific area of the data and groups the outcome of the algorithm as a model. I aim to use GMM and Mean Shift clustering to study voice traffic patterns and user behavior in the network based on CDR data, and then use Z-score, Isolation Forest and DBSCAN to detect and predict anomalies in the network using 37 million of one-year CDR data.

2.5 Time series data forecasting

Time series data study provides crucial insights about economy, marketing, traffic pattern analysis, stock market, weather forecast such as rainfall. I focus in one of my studies on rainfall forecasting since it is essential to study the flood and drought conditions monitoring, landslide activities, soil moisture conditions, and freshwater availability, especially with the latest trends in global warming and climate shifts. Yet, rainfall has a huge impact on water supply reservoirs, underground water, crops, and farms. Therefore, there have been always many articles in the past years about rainfall data analysis and forecasting using different models and techniques. In return, machine learning models made handling the forecasting of rainfall data easier. However, accuracy has always been one of the hardest targets to achieve. Thus, understanding and analyzing the quality of these models are very important to ensure they fit into the objectives with least errors, higher accuracy, and best performance. Another important factor, or might be the most important, is the quality and the quantity of the data as the model relies strongly on the data to provide forecasting.

The rainfall data depends often on the type of the rain gauge that has been deployed. In addition, the amount of rainfall collected by the rain gauge influences the measurement that has been done. Tipping bucket rain gauges that include radar stations have been indicated as one of the most used rain gauges around the globe [51] since it is very straight forward device, quite practical and easy to use.

Various studies engage different models. As stated, it is always better to have more than one model handy for more accurate results achievement, either for comparison or to build a robust model. In the last few decades, many models have been proposed and built to provide more robust and reliable results. For rainfall data forecasting, there are many models that have proven to provide consistency over longer period of time forecasting with low error values and high direction accuracy and correlation such as M5P [52]. In contrast, a shorter period of time rainfall forecast can be achieved using several available models like using Moving Average (MA) utilizing Evolving Neural Network (ENN), support vector machine, Fuzzy system based on Genetic Algorithm (GA) and hybrid climate learning model (HCLM) [53][54][55]. Nevertheless, Artificial Neural Networks (ANN) such as Support Vector Regression (SVR), Decision Tree (DT), Random Forest Algorithm (RFA), and LSTM models have also shown impressive results for rainfall data forecasting as shown in Fig. 2.2 illustrating how specific algorithms work [56][57][58].

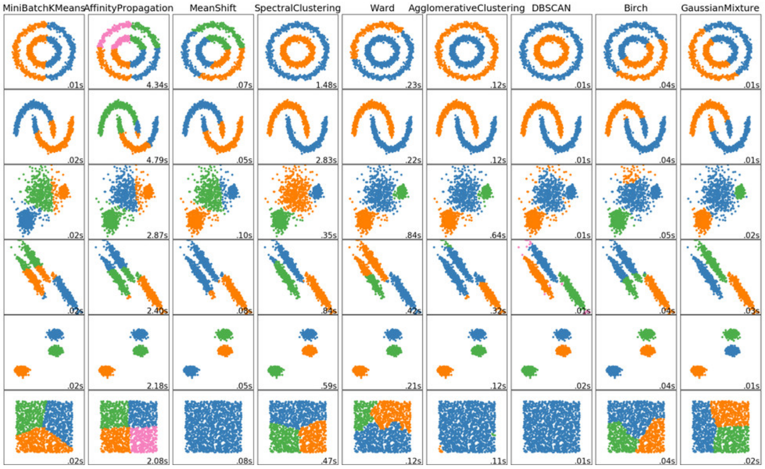


Figure 2.2: Different algorithms comparison.

In recent years, the complexity of the data and extension in the number of parameters included in building the models have led to model improvements. It is the same model but modified in order to fit into the study or research like an improved Radial Basis Function Neural Network (RBFNN) [59] model to realize the match of the weather radar data and the rain gauge data in time.

On the other hand, there is increase of open-source algorithms in the last decade since they are freely available for use and show quite substantial performance in modeling time series data including univariate data such as the Prophet model by Facebook [60]. Numerous studies nowadays propose the Prophet model due to its high reliability, efficiency, and accuracy.

The Prophet model shows considerable results in terms of low Mean Forward Error (MFE) and shorter time needed to be executed. It can handle missing values and outliers as well as seasonality and holiday effects since it is easily tunable. In contrast, Seasonal Auto-Regressive Integrated Moving Average (SARIMA) model is still one of the oldest and most reliable time series forecasting models. It is one of the major and well-known models that has been used for almost a century to forecast the amount of rainfall. However, the data must be stationary, and the model is selected based on the Autocorrelation function (ACF) and Partial Autocorrelation function (PACF). This model does a great job in time series forecasting and with high accuracy outcomes including the research which developed the model before implementing [61][62].

The time series models usually go through certain metric performance evaluation based on their needs. There are plenty of metrics available to value the efficiency of the model and provide crucial results to approve the validity of the model whether it is good to employ or not. Many studies have used root-mean-square error (RMSE), lower Mean Absolute Percent Error (MAPE), Mean Forecast Error (MFE) and Mean Absolute Error (MAE) to enhance the accuracy of the results in their models [63][64].

Chapter 3

Data description and preprocessing

3.1 Call Detail Record

3.1.1 Introduction

A CDR is a type of data record that is generated by the mobile networks to file call scenario details [65]. These records are parsed and stored in the core network of mobile networks, mainly in the CDR database server (see the architecture of mobile networks and CDR database store in Fig. 3.1).

The metadata in CDRs include various attributes such as call date/time, call direction, duration, location, calling and called party numbers, etc.

There are many other attributes that can be perceived in a single call CDR. Each one of them may contribute to a specific study or analysis. These attributes are crucial for network evaluation, efficiency, performance studies, and providing robust statistical analysis. CDR data can also be a strong indicator in defining the dimensions and service availability in mobile networks.

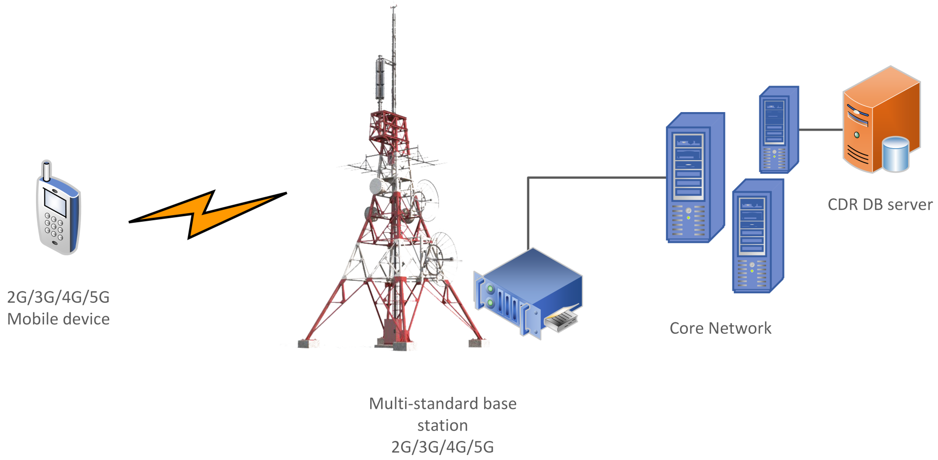


Figure 3.1: Mobile network architecture.

CDR record is mainly written to a flat file (text file) in the mobile core network for every call scenario before being stored in the database. These specialized databases usually store enormous amounts of CDR data every day. In general, there are two different forms of storing such data for each voice call established as well as attempts in the database:

1. Text-based data, which consist of several lines at the time of the voice call being recorded. Every line specifies the call status, including relevant attributes and information. Additionally, the line states when a call starts with call setup timestamp until the call is released in case of a successful call attempt.

2. CDR file that holds hundreds of entire voice call scenario records (text-based data). This file is commonly created every 15 minutes by default and stored in the database or depends on the system configuration.

CDR data are typically represented through attributes and terminologies (i.e. Session Initiation Protocol (SIP), Time Division Multiplexing (TDM), etc.) that detail call scenario steps and their status. TDM has been the traditional and the most common type of voice infrastructure for a very long time. However, SIP has taken over this classic voice calls establishment nowadays, and it is used by the latest technologies in mobile networks such as VoLTE (voice over 4G/LTE), Vo5G (voice over 5G), and Rich Communication Services (RCS). SIP is a signaling protocol for controlling communication sessions (initiating, maintaining, and terminating) as in VoIP voice calls including voice, video and messaging applications. SIP is based on request/response transactions, in a similar manner to the Hypertext Transfer Protocol (HTTP)[66]. Each transaction consists of a SIP request (one of several request methods), and with at least one response. These request/response codes are identifying the status of the call, and they are specified as follows [67]:

- 1xx - Provisional of information, such as status 100 Trying, 180 Ringing or 181 Call is Being Forwarded.
- 2xx – Success, the most common response is 200 OK that indicates the request was successful.
- 3xx – Redirection, it informs the requesting client that further measures are demanded.
- 4xx - Client error, which means the request contains bad syntax or cannot be fulfilled.
- 5xx - Server error, it states that the server failed to fulfill an apparently valid request.
- 6xx - Global failure, it indicates the destination does not wish to participate in the call.

3.1.2 Call Flow Process

The scenario is generally similar for an international as for a local phone call. Fig. 3.2 illustrates a SIP phone call scenario. A completed process of a call, starting from the Connecting status to the End of Call status, is declared as a successful call. A call is established through a SoftSwitch (SSW) in my study. The SSW is basically a computer specialized software, representing the IMS part in the core network, i.e., the switching system element in the network. The SSW typically handles the call processing control and call routing using SIP protocol [68]. Among others, it also converts a voice bit stream into packets and recto-verso.

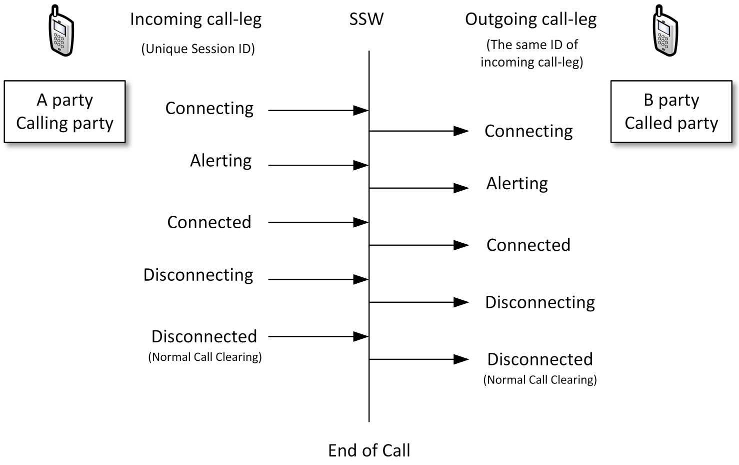


Figure 3.2: A successful voice call flow.

At the initial stage, a Calling party (*A* party) sends an INVITE message to a Called party (*B* party) through the network. The INVITE message initializes the Connecting status of the call, which means that the *A* party is trying to setup a connection and reaching the *B* party.

In the next phase, the SSW creates an incoming call-leg (in other words an incoming channel) for the *A* party and tries to find out a route that would interconnect the *A* and *B* parties. Afterwards, Media Gateway (MGW) inside the SSW is activated and it begins to handle the SIP call and creates an outgoing call-leg for the *B* party.

A channel for both *A* and *B* parties is assigned, and both incoming call-leg and outgoing call-leg are specified, and identified in the system via a unique *ID* [69].

As soon as the INVITE message reaches the *B* party, 18x ringing response message (Alerting) is sent back to the SSW and to the *A* party. The SIP 18x message is usually sent to indicate the status of *B* party, either ringing, additional information needed, or call is being forwarded. If the *B* party is ready to accept the call, it sends back to the *A* party the 200 OK message to establish the call [69]. Once the call connection is established, a Connected status is assigned to both sides of the call. The created CDR file details the whole call scenario (e.g., the time and date of call, the connection type: SIP or TDM, the channel *ID*, etc.).

The call is terminated once the SSW receives another 200 OK message.

As next, the BYE message and Disconnecting request status are issued to the A and B parties by network. The call session is ended, and the Disconnected status is regularly sent by network till the MGW is deactivated [70]. The call session profile is successfully released if all aspects of the call are correctly recorded in the CDR.

3.1.3 Data preprocessing

The exploitation of CDR data is done in the following steps until it is ready for use to be implemented in the algorithms and then modeling (see Fig. 3.3):

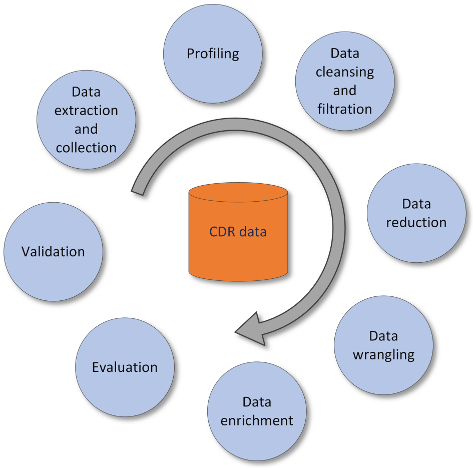


Figure 3.3: Data preprocessing.

1. Data extraction and collection: I first extract the data and collect them from the CDR database.

2. Profiling: I examine and review the quality of the data. I study the content to identify the major attributes of my study.

3. Scrubbing and filtration: indicating and eradicating irrelevant, inconsistent, duplicate, and missing data.

4. Data reduction: this involves data cube aggregation, attributes subset selection, numerosity, and dimensionality reduction.

5. Data wrangling: at this point, the data is still unstructured, I need to transform and convert it into a structured and valuable format.

6. Enrichment: I enhance the quality of my data with knowledge from external sources. This also includes a preliminary understanding of the geographical location of the study.

7. Validation: I divide the data into three sets; training data that are used to train my models, validating set and testing data that I use to help us find any flaws or weaknesses in the pool of my hypothesis and assess the performance of the models.

8. Evaluation: this is to evaluate the models used to avoid overfitting and achieve an unbiased estimation.

3.1.4 The CDR dataset in this thesis

In my research I use CDR datasets. The collected CDR data are generated by the mobile network from July 2016 to June 2017. The CDR data are stored in the CDR database server located at the core of the mobile network as shown in Fig. 3.4.

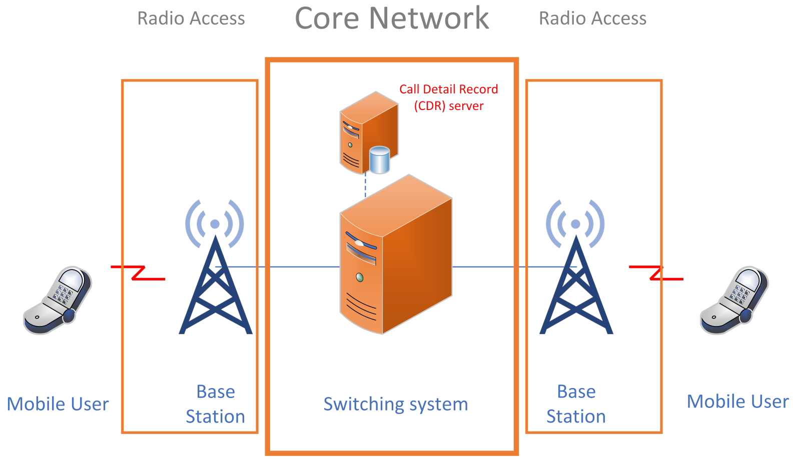
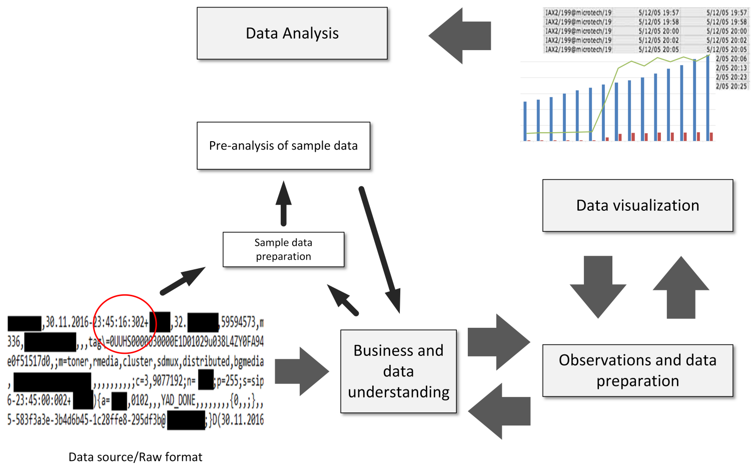


Figure 3.4: CDR database server.

The total amount of investigated CDRs corresponds to over 39 million phone calls of SIP and TDM voice calls. The file size stored in the database varies based on the number of calls recorded within 15 minutes that can go up to a few hundred megabytes.

At the start, I extracted the information by transforming the raw data into a dataset shown in Fig. 3.5. Next, data attributes are filtered out that are appropriate for the study. Finally, the datasets are rearranged by adding missing entries and dropping the unnecessary data. Many statistical analysis approaches can be delivered such as (number of calls during working days and weekends, morning, noon and evening comparison, mean and variance of the calls on hourly, daily, monthly and yearly basis).


Data Analysis
IAX2/199@microtech/19 5/12/05 19:57 5/12/05 19:57
IAX2/199@microtech/19 5/12/05 19:58 5/12/05 19:58
IAX2/199@microtech/19 5/12/05 20:00 5/12/05 20:00
IAX2/199@microtech/19 5/12/05 20:02 5/12/05 20:02
IAX2/199@microtech/19 5/12/05 20:05 5/12/05 20:05
2/05 20:06
2/05 20:13
2/05 20:23
2/05 20:25
Pre-analysis of sample data
Data visualization
Sample data preparation
Business and data understanding
Observations and data preparation
Data source/Raw format

```
[redacted],30.11.2016-23:45:16:302+[redacted],32.[redacted],59594573,m
336,[redacted],,,tag\=0UUHS0000030000E1D01029u038L4ZY0FA94
e0f51517d0,;m=toner,rmedia,cluster,sdmux,distributed,bgmedia
,[redacted],,,,,,,,,;c=3,9077192;n=[redacted];p=255;s=sip
6-23:45:00:002+[redacted]){a=[redacted],0102,,,YAD_DONE,,,,,,,,{0,,;},,
5-583f3a3e-3b4d6b45-1c28ffe8-295df3b@[redacted];}D(30.11.2016
```

Figure 3.5: Research CDR data preprocessing.

The data processing is done by understanding the environment of the business environment and users' behaviors from the data that are given (see Fig. 3.5, it shows how raw CDR data look like in the early stage). In the CDR files, different parameters are given to provide the status of the phone call scenario. I perform data observation and preparation to put them in the right form for transformation. Once the data are transformed, the visualization is done via using proper scales and graphs. Finally, the data are analyzed according to the given parameters and business requirements.

Data selection and preparation steps are typically the most time-consuming phases in data preprocessing. I have a vast pool of attributes in the dataset. The only attributes that lead to the required goals and knowledge have to be adequately selected.

In addition, there are many attributes and terminologies per every line that have to be jointly interconnected with attributes and terminologies in other lines. Furthermore, missing, inconsistent or erroneous records in an analyzed dataset are typically other issues that I have to face while preprocessing the data, and which take a lot of time to satisfyingly resolve them.

In real-world, there are many reasons why the stored raw data in a network system can be incorrect, invalid, or certain attributes even missing (e.g., incorrect data recording, fraud calls, network misconfiguration, device malfunctioning, etc.). Finally, the data have to be transformed into a structure that is suitable for the final presentation and visualization, and then analysis and modeling.

A decision tree model in my study is built to verify whether a phone call session is a successful or an unsuccessful call (shown in Fig. 3.6). There are several parameters in the CDR datasets that can define the accomplishment of a phone call. For instance, the duration as well as the cause value of the call are the keys to give the status of the phone call. The cause value is a code, which is provided to state that the session either ended successfully or not as described earlier in this chapter.

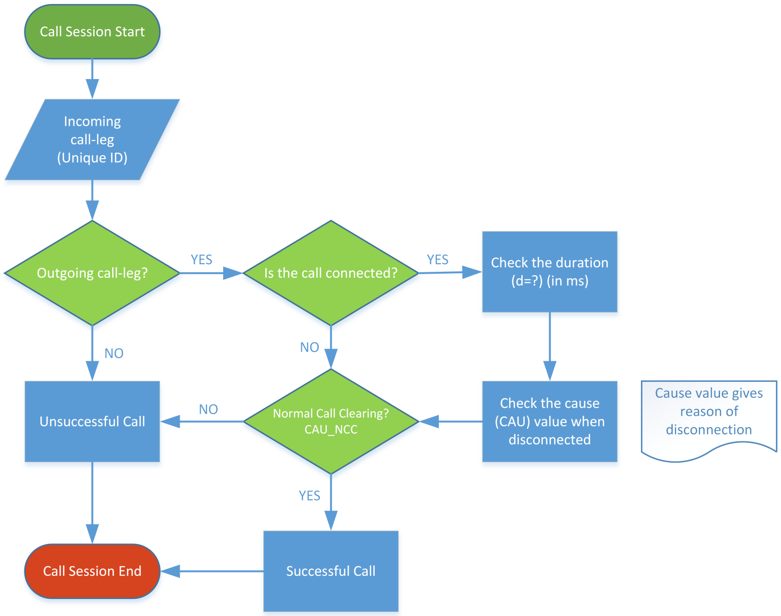

Call Session Start
Incoming call-leg (Unique ID)
Outgoing call-leg?
YES
Is the call connected?
YES
Check the duration (d=?) (in ms)
NO
NO
Unsuccessful Call
NO
Normal Call Clearing? CAU_NCC
Check the cause (CAU) value when disconnected
Cause value gives reason of disconnection
YES
Call Session End
Successful Call

Figure 3.6: Decision tree model.

3.2 Signaling data

Signaling is defined as the use of messages to control the communications between network components. The signaling in the mobile networks are responsible for:

- User connection and call establishment.
- Location services.
- Communication between the user and the network.
- User status in the network.
- Network elements status.

CDR data can contribute to providing signaling data as described in CDR research data in the previous section for voice call establishment. The data are collected through daily generated log files with several extensions, each log file details certain data for a specific purpose. There are several parameters in each log file that play a vital role in a variety of network analysis and optimization. For such investigations, I exploit selected parameters such as waiting, service and interarrival time. These three parameters are essential to analyze the response of the network delivering services to its customers in terms of voice traffic.

The waiting time represents the time period between the connecting and the connection phase of call. It is usually the sum of ringing and network call processing time. The ringing time period depends on several aspects such as the calling party patience, the called party awareness of the call, time zone difference, etc. On the other hand, the network time period goes through different processing phases before connecting the two parties. It also depends on how the two interconnected networks handle the call (e.g., system configuration, routing path, called party location area, etc.).

The service time is the total duration of the call. It refers generally to the interval between the time period when the phone is picked up (connection time) and hung up (disconnection time). Finally, the interarrival time is the time period between two consecutive call arrivals [71]. The average time between two calls is a key indicator of the traffic amount between two interconnected networks or users that is used for resource and/or network dimensioning purposes.

3.3 Population dataset

An important factor that should be taken into account when investigating the voice traffic is the number of habitants per country.

Highly populated countries naturally generate more traffic than the lower populated countries. For instance, a country of 1 million habitants with 10000 voice calls per time unit makes no difference in the voice traffic from a country of 10 million population with 10000 voice calls per time unit. That is, at first sight both countries would be seen similarly important from the traffic point of view. However, by considering the difference in population of both countries, the first one would be considered to have a higher number of calls than the second one.

To avoid this population effect and to obtain relative values, I apply normalization to the voice traffic according to the population datasets. The population datasets of countries are in form of comma separated values and have been collected and used from [72]. The source of population estimation is based on the official website of the United Nation of 2016 to match with the year of my CDR datasets. The coefficient of normalization for a country j, denoted as c, is used to determine the number of calls per 10^3 inhabitants and I calculate it as:

$$c_j = \frac{n_j}{p_j} * k \tag{3.1}$$

where the n_j represents the total number of calls for the given country j per three-month period, p_j indicates the population of country j, and the parameter $k = 10^3$. In the equation, I propose $k = 10^3$ as more reasonable results for both populations of countries and number of calls can be obtained. For instance, $k = 10^2$ is relatively small for higher populated countries, whereas $k = 10^4$ is relatively high for the least populated countries in my analyzed dataset.

3.4 Distance and Centroid of countries

The distance among countries in favor of finding the dependency of voice traffic on distance between two interconnected countries. It is an important factor influencing voice traffic characteristics. Technically speaking, the actual call routing path (number of routing points, distance between two routes, locations of routes, network media, etc.) defines the real distance between two interconnected calls (see Fig. 3.7). However, such information is hardly available. Therefore, to calculate theoretical distance between two countries, I use the great-circle distance approach [73], which is specified as the shortest distance between two points on the surface of a sphere.

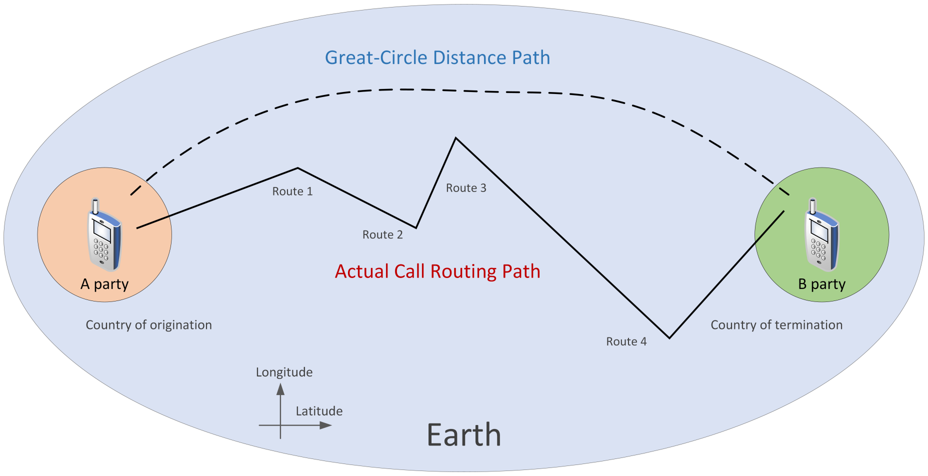


Figure 3.7: Great-Circle Distance path.

When calculating distance between two countries, I encounter 3 main issues: i) two reference location points have to be assigned ii) a country is not represented as a point but a highly irregular polygon, and iii) the major cities are sporadically located within countries, mainly in case of countries with large areas. To overcome these issues, I consider the country centroid as the reference point, i.e. I assume the distance between two countries to be the distance among their centroids [74] (see Fig. 3.8).

Geographic coordinate system allows every single location to point on earth to be defined by latitude (the north-south position of a location point on the earth), resp. longitude (the east-west position of the same location point). Geographical coordinates are exerted to measure the distance between two countries.

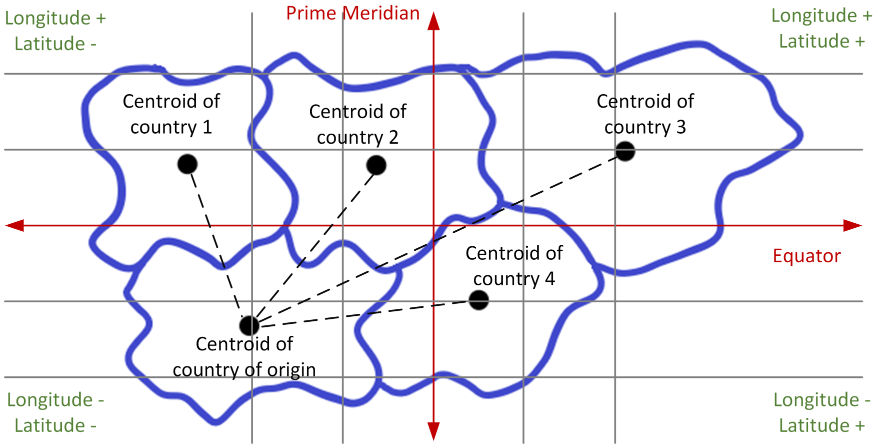


Figure 3.8: Mapping centroids of countries on spherical coordinates of earth.

To determine the great-circle distance between the centroids of origin, i and the destination countries, j, the haversine formula is applied [75]:

$$a_{ij} = \sin^2(\frac{\Delta\varphi_{ij}}{2}) + \cos\varphi_i. \cos\varphi_j. \sin^2(\frac{\Delta\lambda_{ij}}{2}) \tag{3.2}$$

$$c_{ij} = 2. \arctan 2(\sqrt{a_{ij}}, \sqrt{(1 - a_{ij}}) \tag{3.3}$$

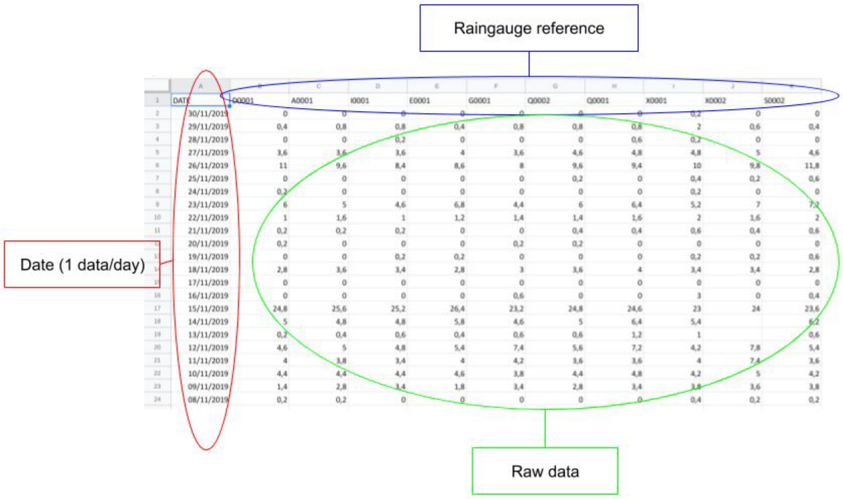
$$d_{ij} = R.c_{ij} \tag{3.4}$$

where $\varphi(\lambda)$ is the latitude (longitude) of country centroid, Δ is the difference between φ and λ, and d is the distance between centroids. The earth radius, R, is set to 6371 km [76].

Notice that the d_{ij} represents the theoretical distance between two countries i and j, as in the reality, the distance is given by the physical communication routing path.

3.5 Rainfall data

In meteorology, precipitation or rainfall is any outcome of the condensation of atmospheric water vapor that falls from clouds due to gravitational pull. There are several forms of precipitation include drizzle, rain, sleet, snow, hail, etc.



	A	B	C	D	E	F	G	H	I	J	K
1	DATE	O0001	A0001	I0001	E0001	G0001	Q0002	Q0001	X0001	X0002	S0002
2	30/11/2019	0	0	0	0	0	0	0	0,2	0	0
3	29/11/2019	0,4	0,8	0,8	0,4	0,8	0,8	0,8	2	0,6	0,4
4	28/11/2019	0	0	0,2	0	0	0	0,6	0,2	0	0
5	27/11/2019	3,6	3,6	3,6	4	3,6	4,6	4,8	4,8	5	4,6
6	26/11/2019	11	9,6	8,4	8,6	8	9,6	9,4	10	9,8	11,8
7	25/11/2019	0	0	0	0	0	0,2	0	0,4	0,2	0,6
8	24/11/2019	0,2	0	0	0	0	0	0	0,2	0	0
9	23/11/2019	6	5	4,6	6,8	4,4	6	6,4	5,2	7	7,2
10	22/11/2019	1	1,6	1	1,2	1,4	1,4	1,6	2	1,6	2
11	21/11/2019	0,2	0,2	0,2	0	0	0,4	0,4	0,6	0,4	0,6
12	20/11/2019	0,2	0	0	0	0,2	0,2	0	0	0	0
13	19/11/2019	0	0	0,2	0,2	0	0	0	0,2	0,2	0,6
14	18/11/2019	2,8	3,6	3,4	2,8	3	3,6	4	3,4	3,4	2,8
15	17/11/2019	0	0	0	0	0	0	0	0	0	0
16	16/11/2019	0	0	0	0	0,6	0	0	3	0	0,4
17	15/11/2019	24,8	25,6	25,2	26,4	23,2	24,8	24,6	23	24	23,6
18	14/11/2019	5	4,8	4,8	5,8	4,6	5	6,4	5,4		6,2
19	13/11/2019	0,2	0,4	0,6	0,4	0,6	0,6	1,2	1		0,6
20	12/11/2019	4,6	5	4,8	5,4	7,4	5,6	7,2	4,2	7,8	5,4
21	11/11/2019	4	3,8	3,4	4	4,2	3,6	3,6	4	7,4	3,6
22	10/11/2019	4,4	4,4	4,4	4,6	3,8	4,4	4,8	4,2	5	4,2
23	09/11/2019	1,4	2,8	3,4	1,8	3,4	2,8	3,4	3,8	3,6	3,8
24	08/11/2019	0,2	0,2	0	0	0	0	0	0,4	0,2	0,2

Figure 3.9: Sample of my rainfall data.

In my study, the rainfall data are provided by French public services from 18 pluviometers from 2009 to 2021 [77]. They are available to the public either through (*.json*) or (*.csv*). I choose the *.csv* file, which is structured in DateTime and pluviometer values. The dataset is a collection of two components: rows and columns (a sample of the data is shown in Fig. 3.9). Each row comprises one observation measured by a gauge per unit of time.

It is a conventional practice to split the data into two parts when I select the models: training and test data. Training data aim to estimate forecasting parameters and train the models for forecasting. Meanwhile, test data is used to evaluate the accuracy and indicate how the model performs on new data.

The absence of a measurement due to a fault or technical problem causes an empty or zero cell value. These zero values are removed while preprocessing the data to avoid data averaging issues.

The rainfall data analysis and forecasting have a significant role in many fields such as identifying potential flooding conditions, agricultural plans and strategies.

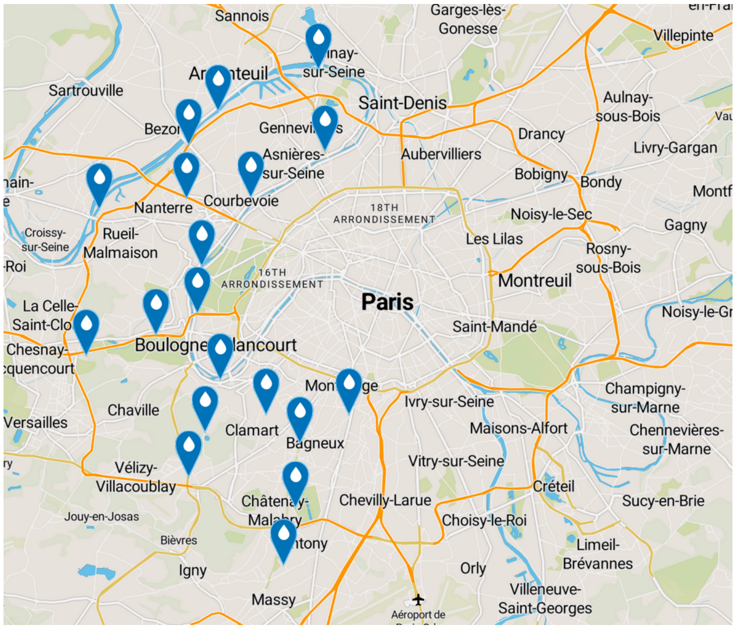


Figure 3.10: The spatial distribution of rain gauges.

Rainfall data are considered a matter of public safety and help the governments to track trends and forecast the climate. The data are measurements of 18 rain gauges deployed in the Hauts-de-Seine region, the western part of Paris, France (see Fig. 3.10).

This region is located at an elevation of 76.57 meters above sea level. It has a marine west coast with a warm summer climate. Hauts-de-Seine has a yearly temperature of around 11.7 centigrade, which is a bit higher than France's average temperature. This part of France annually receives around 641 mm of precipitation. Therefore, it is crucial to understand the irregular and seasonal rainfall along with the amount that the region receives every year.

The used rain gauge for the rainfall measurement is Pluviometer (see Fig. 3.11) is a tipping bucket rain gauge with tilting speeds that tends to record cumulative rainfall.

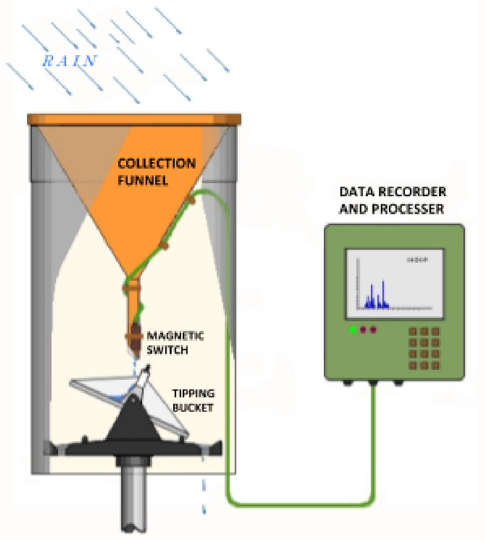


Figure 3.11: The rain gauge sensor.

Each gauge consists of a small bucket with a receiving cone. When the maximum capacity (0.2mm) of the bucket is reached, the pivot switches. The lever counting system has a tipping point. The total number of switches is then continuously transmitted by the acquisition and teletransmission equipment connected to the pluviometer. The system, therefore, has an accuracy of 0.2 mm [78].

The rain gauges are equipped with acquisition and teletransmission equipment that continuously transmits collected data to satellite stations. The data are transmitted from the satellite stations via ADSL (Asymmetric Digital Subscriber Line) to the central supervision in Suresnes, Paris. The system consumes between 300 and 500 kWh (kilowatt hour) per year and most of the electricity consumption comes from the electrical cabinet to which the rain gauge is connected.

In Fig. 3.12, the matrix shows that all the rain gauges are in agreement. On the left y-axis and x-axis, the rain gauges are represented. According to the analysis, when it rains in one place in the Hauts-de-Seine, it tends to rain everywhere in the region.

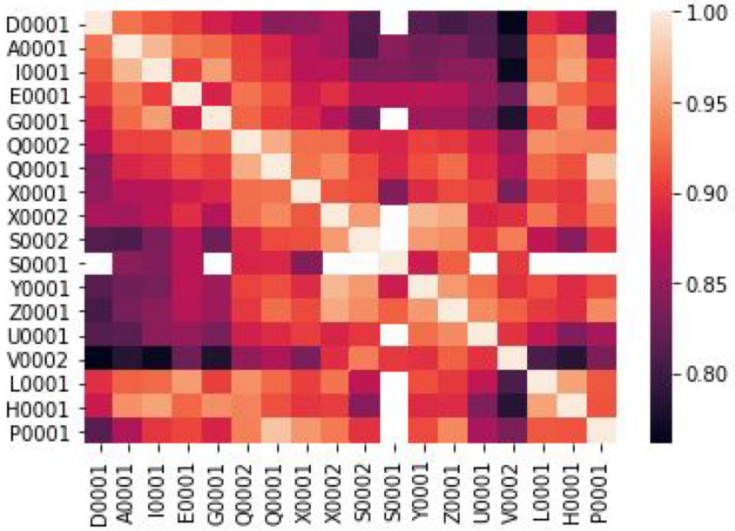


Figure 3.12: Correlation among all rain gauges.

This information, therefore, allows to tell that it is not necessarily useful to use many rain gauges in an area to know the overall rainfall in the region. Furthermore, reallocating the rain gauges offers covering a broader area to study. Hence, the reduction of the number of the current rain gauges used along with redistribution still provides similar analysis and forecasting results.

Chapter 4

Analysis of CDR data

4.1 Introduction

The telecommunications industry is witnessing rapid growth in the mobile network sector [79], driven by the adoption of advanced technologies, the introduction of novel services, and the expanding user base. With a rising number of subscribers and heightened engagement in activities like online gaming, social networking, and video streaming, the volume of user and operational data is experiencing exponential growth. Consequently, mobile operators are facing increasingly intricate challenges in managing data storage and processing [80].

Besides the storing/processing data issues, extracting useful information and its proper interpretation represents another big challenge for operators. That is why nowadays Telco companies dedicate a decent budget to hire data analysts and specialists in this field, aiming to maximize the utility derived from their data [81].

In this chapter, I deal with CDR data analysis. The definitions and details of the CDR data were explained in the previous chapter. These operational data have a crucial importance as they characterize connections in mobile networks by providing metadata about these connections [82]. Every CDR file contains information such as a phone call scenario (connection time, release time, duration, date, calling and called parties, etc.), information about incoming and outgoing call-legs, and addresses of the switching systems at both sides of connections or identification of the connection type.

A CDR is created for every single phone call ([80][82]), whether it's a local or an international call. Providing details about an entire phone call connection makes the call easily traceable, especially for the reason of network statistics or system routine health check (drops, silent calls, etc.). I analyze a short/long-term traffic evolution. The study focuses on international call scenarios by analyzing CDRs of these calls in terms of daily, weekly, monthly and yearly traffic profile. In addition, statistics of working and weekend days per year are presented as well. Finally, I analyze the original and destination countries of these calls and the total amount of traffic distribution among these countries.

This sort of analysis is usually quite sensitive to the analyzed timestamp as the investigated time frame should be carefully chosen based on the network, and/or customers' aspects. For instance, selecting an ordinary working day versus weekends, holidays or a month with no holidays, events versus a month with many holidays, etc.

Additionally, when analyzing and interpreting such data in telecommunication networks, there are usually terminology issues as various Telco companies, vendors and technologies apply different terminologies when presenting their outcomes or products. Though, they all refer to the same meaning. For instance, the terminologies used in 2G, 3G and 4G are different to the terminologies used in a VoIP network [83], e.g., termination/origination vs. incoming/outgoing, or using INVITE message in SIP vs. channel request in TDM to setup a call.

The major contribution of this chapter can be summarized as followings:

- Analysis of CDR data per different periodical scenarios.
- Voice traffic distribution among the countries.
- Peak hours during the day and week.
- Cumulative Traffic Distribution function.
- Correlation between voice traffic to population and distance.
- Dependency between the volume of traffic and the destination countries to the reference network.

4.2 Analysis of voice traffic data

4.2.1 Daily voice traffic

This section describes obtained results. Fig. 4.1 shows a daily evolution of outgoing call traffic. The daily profile analysis is done for four consecutive Wednesdays in October (5, 12, 19, and 26. 10. 2016). I have selected Wednesdays for the daily traffic analysis as it is a typical working day in the middle of a working week; in the studied geographical region, and in majority regions all over the world. Similarly, October also represents an ordinary month, without any specific holidays or events.

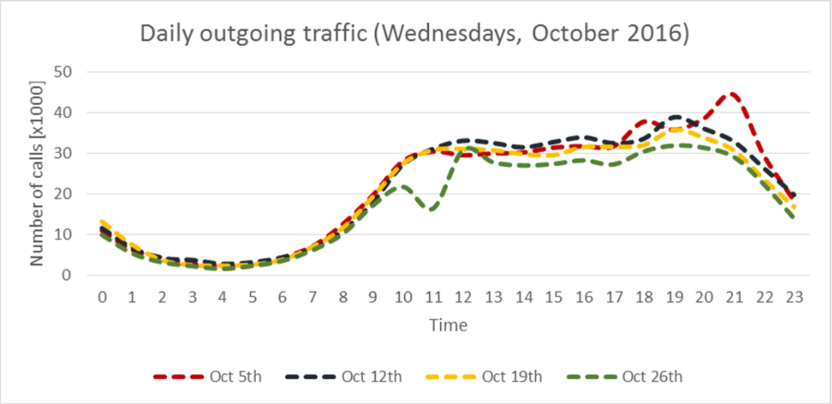


Figure 4.1: Daily outgoing traffic (Wednesdays, October 2016).

The number of calls (y-axis) is counted on a minute basis, i.e. the number of calls that begins in a given minute. If a call takes n minutes, the call is counted only once, which is at the starting minute.

As can be observed, the traffic follows the daily human behavior and activities. During the night and early morning, from 0:00 to 05:00, the users' activities are very low, as the majority of people are sleeping, and therefore the traffic is nearly zero. From 05:00, the traffic progressively starts to grow, as people wake up and start their daily activities and routines, such as going to work, school, etc. The traffic keeps growing until it reaches the peak, which occurs on average around 11:30.

From noon onward, the traffic slightly goes down due to lunches/afternoon break times. Afternoon traffic is momentarily constant, followed by a slight change, until reaching the evening. The traffic gradually grows around 18:00

and hits the second peak around 19:30. This traffic increase reflects the calling of people to their friends and relatives, who live abroad. In general, these certain habits depend on the location. The second peak, or evening peak usually takes around 2-3 hours until the traffic slowly goes down with almost none after midnight.

There are two obvious abnormalities seen in the graph. The first one, October 5th (drawn in discrete red color, in Fig. 4.1), reflects higher calling activities of people in the evening than usual. The second abnormality, October 26th (which is discrete green color in Fig. 4.1), is due to technical issues faced at the reference network around 11:00. This applies also to incoming traffic abnormalities for those two mentioned specific hours (Fig. 4.2).

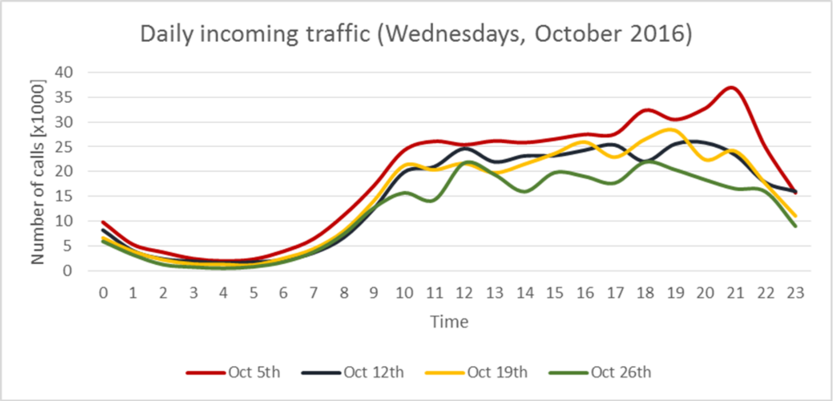


Figure 4.2: Daily incoming traffic (Wednesdays, October 2016).

Otherwise, as can be observed from the figure, there are typically two peaks in the daily traffic: midday and evening peak. The midday peak occurs around noon (due to people's working obligations), and the evening peak is in the evening (which is more related to people personal activities).

The incoming traffic profile is very similar to the outgoing traffic as shown in Fig. 4.2. During the night the traffic is nearly non-existent. From 05:00, the traffic starts to grow and keeps growing till reaches the peak on average at about 12:00. Since that moment, the traffic is somehow stable until the evening, at about 19:00, then it again starts to decrease.

By comparing Fig. 4.1 and Fig. 4.2, I can notice a higher spread in the incoming traffic, compared to the outgoing traffic. During the analysis of the dataset for this specific period of time, no specific network technical issues, events, or policies were observed. Thus, I assume this spread is due to the fact that the incoming traffic comes from all around the world, i.e., the pool of hypothesis is much larger, compared to the users of the reference network and their generated (outgoing) traffic.

As can be observed from Fig. 4.1 and Fig. 4.2, the daily profiles follow a very similar pattern, regardless of selected weekdays.

In both outgoing and incoming traffic, the season plays an important role in the midday and evening's peaks. These are shifting according to the season as well as daylight saving time and country of origin. The midday peak is slightly affected, but the evening peak usually occurs around 19:00-20:30, while in winter the peak is around 17:30-18:30.

4.2.2 Weekly and monthly voice traffic

Fig. 4.3 illustrates the weekly traffic, from Monday to Sunday. The analyzed week is the first week of May 2017, that can be considered a regular week of the month without any abnormal behavior of people. The number of calls (y-axis) indicates the unique number of calls in a given day. In case a call takes place over a midnight, the call is counted into the day when the call begins.

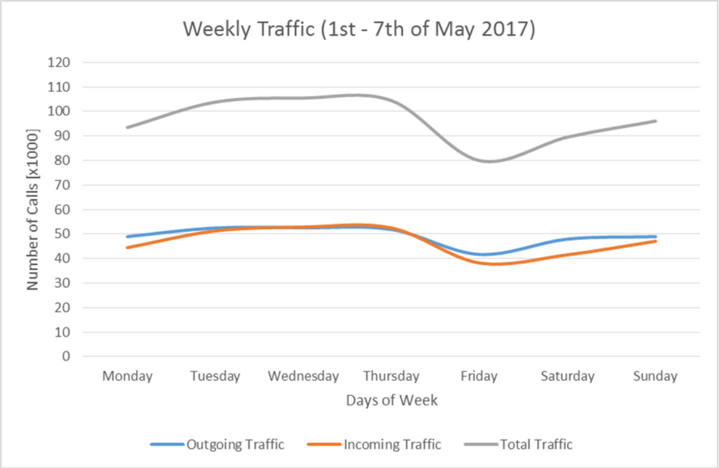


Figure 4.3: Weekly traffic ($1^{st} - 7^{th}$ of May).

The traffic is constant from Tuesday to Thursday, which are typical working days in all regions of the world. The Friday traffic drop is due to the official day off in the studied region (in the studied geographical region, the weekends are Fridays and Saturdays).

On Saturday, traffic begins to increase until reaches the maximum on Tuesday. The reason of the lower Monday traffic compared to Tuesday-Thursday traffic is a fact that Monday is the first working day of week in many countries around the world, and people are just back to their work after the weekend.

As can be seen from Fig. 4.3, the incoming and outgoing weekly traffic profiles are almost the same. Both incoming and outgoing depend on working days and days-off, origin of calling and called parties. People typically respect each other's timezone when calling each other.

Fig. 4.4 shows the monthly traffic, where I again have selected May 2017 for the analysis. Obviously, the monthly traffic consists of the weekly traffic repetition, and the traffic is higher during the weekdays and lower during the weekends. Similarly, to the weekly profile, both incoming and outgoing monthly traffic is nearly the same.

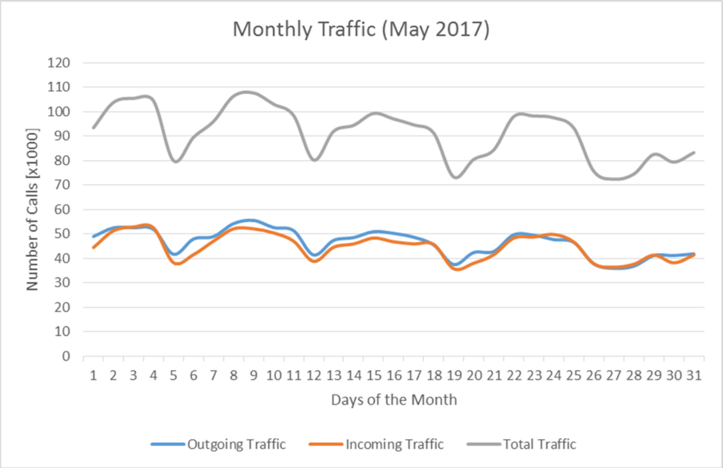


Figure 4.4: Monthly traffic (May 2017).

The first two weeks show a higher number of calls compared to the third and fourth week. This is due to the fact that May is the last academic month (in the studied geographical region) followed by summer holiday. Usually during summer holidays, less traffic is expected because of holidays, many people travel and have vacations.

4.2.3 Yearly voice traffic

Finally, Fig. 4.5 illustrates yearly traffic, from July 2016 to June 2017.

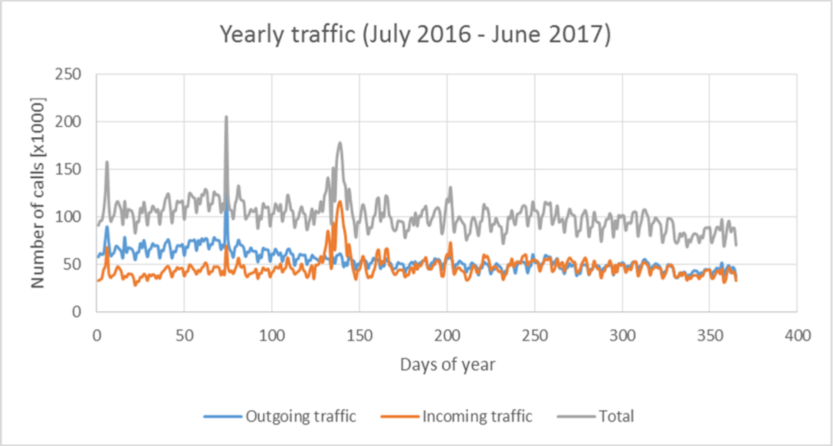


Figure 4.5: Yearly traffic (July 2016-June 2017).

During the first four months, from July 1st to November 30th, i.e., days 1-123 in Fig. 4.5, the incoming traffic is lower than the outgoing. This is due to company's policy and restrictions on the incoming traffic, from the End-user networks.

In Fig. 4.5, I can observe 3 peaks that match reported events in the analyzed region. The 1st peak (day 6), reflects religious holidays in Islamic countries, celebrating the end of Ramadan, which usually results in high traffic during that time. The 2nd peak (day 74) represents the feast days in Islamic countries, where the traffic is doubled compared to normal days. Finally, the 3rd peak (days 138-148) corresponds to a regional holiday where the incoming traffic is more affected than the outgoing traffic.

For such scenarios, I notice rapid increment in traffic. Sometimes, It is expected for some recurrent events. In contrast, there are some high traffic moments that are out of scope. For that, the network cannot handle such huge traffic, which leads to network congestion. In this case, these unpredicted incidents should be studied in order to avoid future network congestion, and mark them as anomalies.

Fig. 4.6 depicts linear regression for the yearly traffic. The incoming traffic is quite stable with no changes expected, except the predicted peaks as explained previously. On the other hand, the outgoing traffic shows notice-

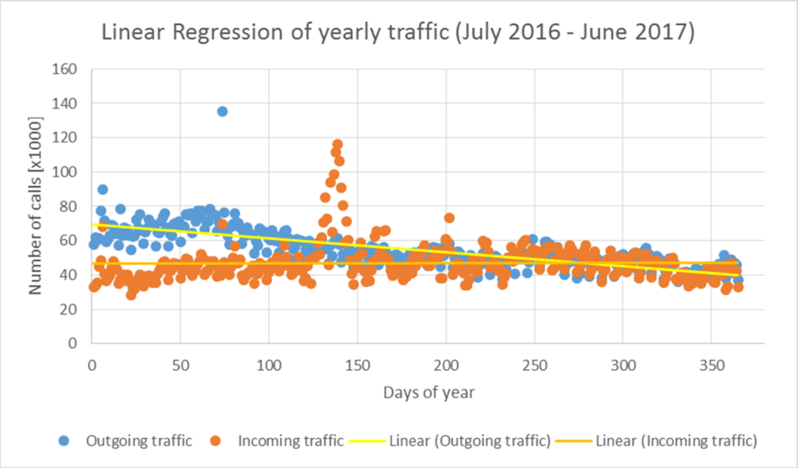


Figure 4.6: Linear Regression of yearly traffic (July 2016-June 2017).

able changes in the beginning and then gradually goes back to the level of incoming traffic. As already mentioned, the decrease is due to the operator's policy.

4.2.4 Weekdays and Weekends

The traffic changes during the whole year (July 2016-June 2017) for each day per week is illustrated in Fig. 4.7 and Fig. 4.8.

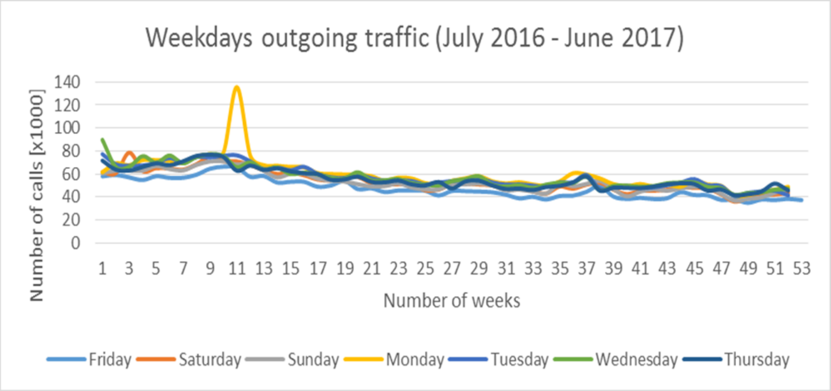


Figure 4.7: Weekdays outgoing traffic during the year.

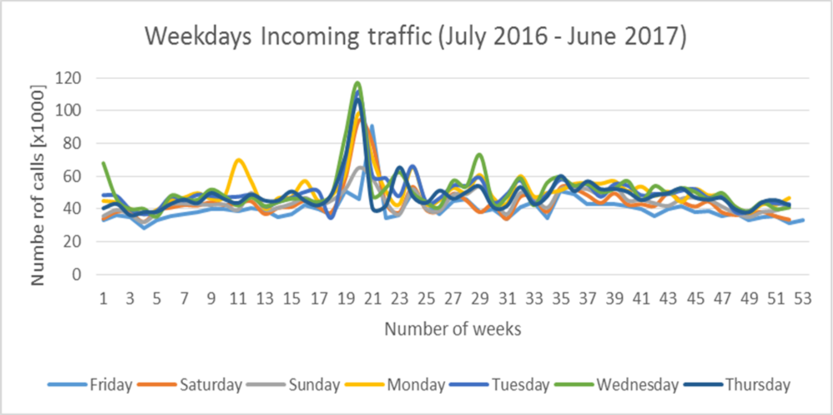


Figure 4.8: Weekdays incoming traffic during the year.

The figures describe how the traffic changes during 52x Saturdays to Thursdays, and 53x Fridays. In the figures, I can observe the abnormalities as explained above (religious holidays, and the decrease of outgoing traffic in the first part of the year).

Table 6.1 summarizes the total number of calls per year for each day (from Monday to Sunday). The mean (μ) and standard deviation (σ) for each day are provided as well.

Table 4.1: Statistics of calls per year.

Days	*Total calls per year*	μ [calls]	σ [calls]
Monday	5725454	108027	18898
Tuesday	5590481	105481	14609
Wednesday	5624193	106117	16934
Thursday	5401519	101915	13508
Friday	4625490	87273	11869
Saturday	5066378	95592	14301
Sunday	5065466	95575	10340

The lowest, resp. highest, μ are for Fridays (weekend day), resp. Monday (working day). In the table, I see Monday has the highest number of calls among other days. This is due to two of the major holidays in the studied geographical region occurred on Mondays, which have huge impact on the number of calls. If I discard these exceptional calls due to those two events, Tuesdays and Wednesdays are the days with the highest number of calls, which corresponds to the above discussion when describing the weekly traffic profile.

4.2.5 Peak hours

An example of noon/evening peaks movement for the incoming and outgoing traffic within a 3-month period (October-December 2016) is illustrated in Fig. 4.9 and Fig. 4.10. The peaks are calculated with a 60-minute window, which is shifted minute by minute to find out the maximum values at noon and in the evening for each day. The points in Fig. 4.9 and Fig. 4.10 show the beginning of the max. 60-minute window.

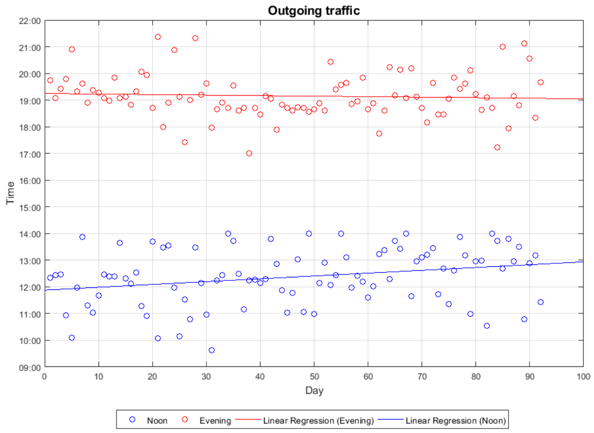


Figure 4.9: Outgoing traffic peaks of October-December 2016.

In case of the outgoing scenario (Fig. 4.9), the obtained results show oscillations around 12:00h, resp. 19:15h, with a slow movement towards 13:00h, resp. 19:00h, by the end of December. Similar values can be observed in the incoming traffic (Fig. 4.10), the initial oscillations occur around 12:15h, resp. 19:00h, with a slow movement towards 13:15h, resp. 19:15h, by the end of December.

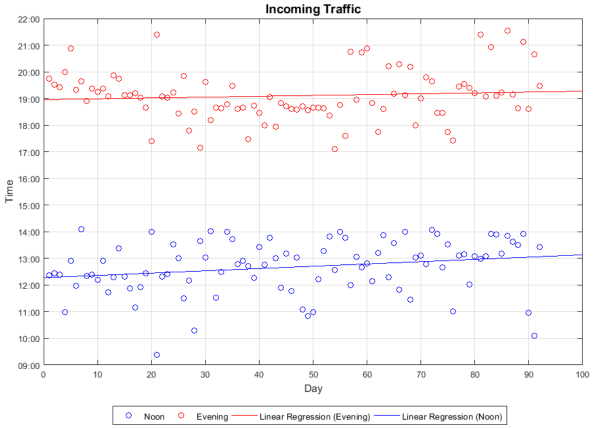


Figure 4.10: Incoming traffic Peaks of October-December 2016.

4.2.6 Cumulative Traffic Distribution

Apart from traffic profiles for different time profiles, an operator also needs to know the traffic distribution among the end-user networks and the countries with the highest traffic for incoming/outgoing. Such information helps the operator better plan for its network capacity and dimensions.

Fig. 4.11 represents the traffic distribution among all interconnected networks, more precisely countries, for 4 months (April-July 2017); similar results can be observed during these months per year.

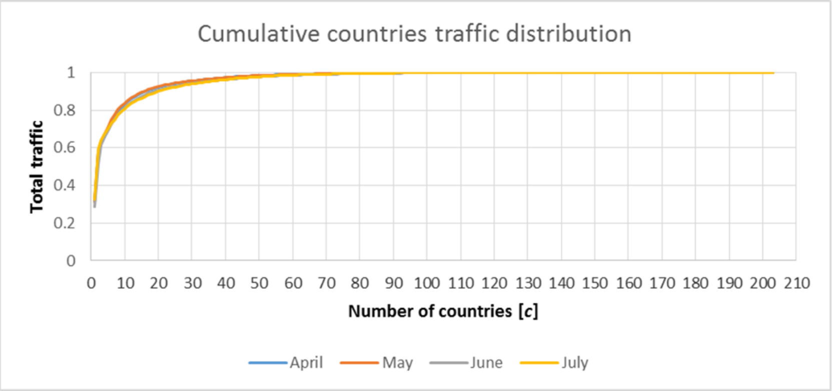


Figure 4.11: Cumulative traffic distribution of April-July 2017.

In this study, the reference network is interconnected in total to 204 countries. I lay out the countries based on their traffic, from the highest to lowest one, and I count the Cumulative Traffic Distribution function using the following formula:

$$CTD = \sum_{i=1}^{c} \frac{\text{Traffic of country}_i}{\text{Total traffic of countries}} \tag{4.1}$$

In Fig. 4.12, I show the top 10 countries with the highest traffic. All 10 countries, and their order, are the same for the analyzed 4 months. By analyzing the whole year, I observe that the given top 10 countries, and their orders, are approximately identical throughout the year.

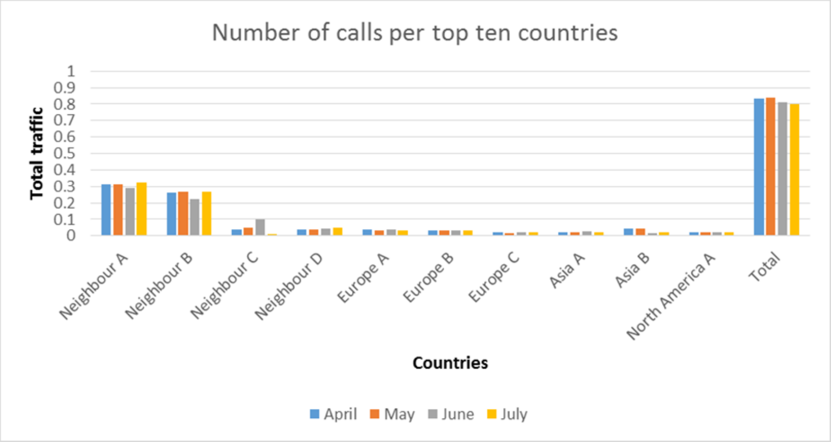


Figure 4.12: Number of calls per top 10 countries, for April-July 2017.

As can be seen, the top 10 countries in total take over 80 percent of the total amount of traffic, where about 60 percent represents the outgoing and 40 percent the incoming traffic. It means, there is 0.8 probability that an international call comes from one of the top 10 countries. On the other hand, the majority of countries only show 1-2 calls per month (see Fig. 4.11).

As expected, the largest impact on the total number of calls comes from the neighboring countries. This is typically occurs due to strong relationships among the neighboring countries in terms of business, culture, trading, tourism, etc.

4.3 Voice traffic to distance dependency

In this section, I investigate the international outgoing/incoming voice traffic dependency based on three-month Call Detail Records dataset analysis. The distance between countries, more precisely the distance between centroids of countries, is calculated by using the great-circle distance approach. Additionally, the voice traffic parameters are normalized in respect of the population of countries to obtain comparable outcome independently. For that, I employ different types of datasets that contribute to the CDR data analysis. However, these datasets need to be merged before employing.

4.3.1 Merging datasets

The dataset merging is illustrated in Fig. 4.13. At first, the voice traffic data of countries are exploited from the CDR dataset. Then, I normalize the populations of countries to voice traffic distribution to eliminate the population effect as mentioned in the previous chapter. Furthermore, I utilize the centroid data to determine the great-circle distance among the centroids of countries. Finally, the results from the calculated distance are ordered from the nearest to the furthest to the country of origin with the normalized data.

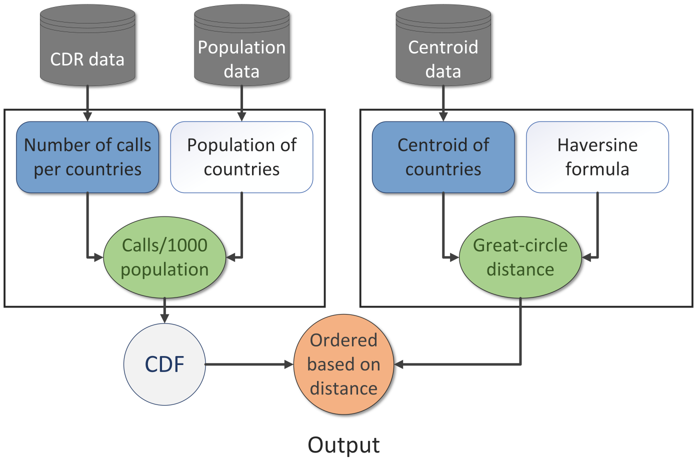


Figure 4.13: Merging and processing of data sources.

4.3.2 Voice traffic parameters

The analyses are done in two steps. In the first step, I determine and analyze the parameters of outgoing/incoming international voice traffic: i) the waiting time, ii) the service time, and iii) the interarrival time.

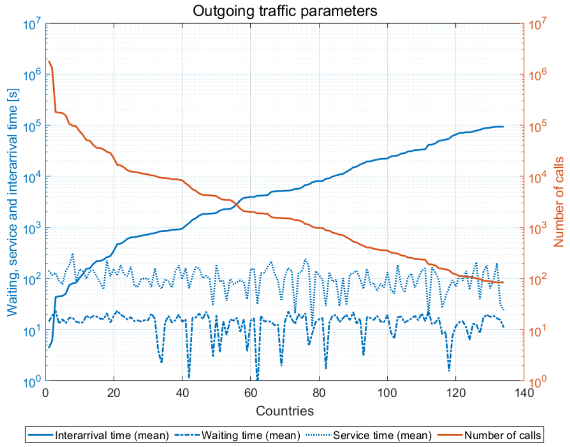


Figure 4.14: Outgoing traffic parameters.

The mean values of these parameters are shown in Fig. 4.14 and Fig. 4.15 in accordance with the total number of calls per 3-month period and per country.

In the figures, I jointly show all the three parameters for the outgoing and incoming traffic.

On x-axis, the countries are ordered from the highest number of calls to the lowest ones, i.e., the last country has only a few calls per 3-month period. On right (resp. left) y-axis, the total number of calls per country (resp. the mean values of waiting, service and interarrival time) are illustrated. From the figures, the mean values of waiting time and service time are nearly independent of the number of calls. Whereas the interarrival time manifests dependency on the number of calls.

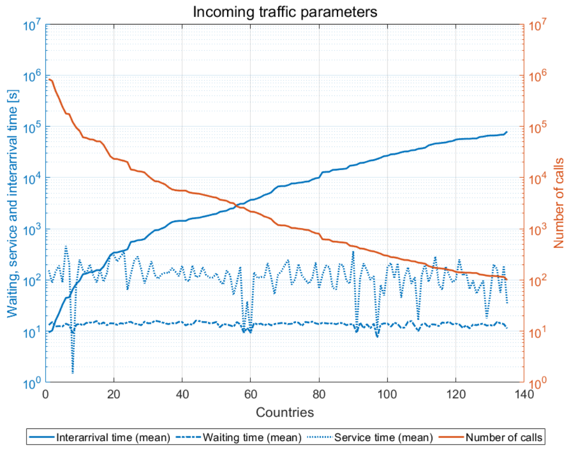


Figure 4.15: Incoming traffic parameters.

When comparing Fig. 4.14 and Fig. 4.15, the incoming waiting time shows higher stability over the outgoing one. This is due to the fact that a calling party usually experiences different waiting times according to the destination. In other words, the outgoing traffic fits the habits of each country around the world individually, while the whole world traffic behavior is with respect to the reference network.

4.3.3 Cumulative distribution function

In the second step, I demonstrate the influence of the destination countries on the generated outgoing/incoming international voice traffic. In Fig. 4.16 and Fig. 4.17, the countries are ordered based on the distance (x-axis); from the nearest to the farthest destinations from the reference country. On the right y-axis, the cumulative distribution function (CDF) of outgoing/incoming traffic ratio is shown.

The ratio is calculated by considering the number of calls per 10^3 populations unit of countries (as explained in the previous chapter). The CDF is applied to investigate the voice traffic distribution among countries based on the ordered distance. On the left y-axis, the distance of countries (in *kilometers*) to the reference country is indicated.

As can be observed from Fig. 4.16, there can be three major regions distinguished where the traffic significantly grows: a) 0 - 1500 km, b) 3000 -

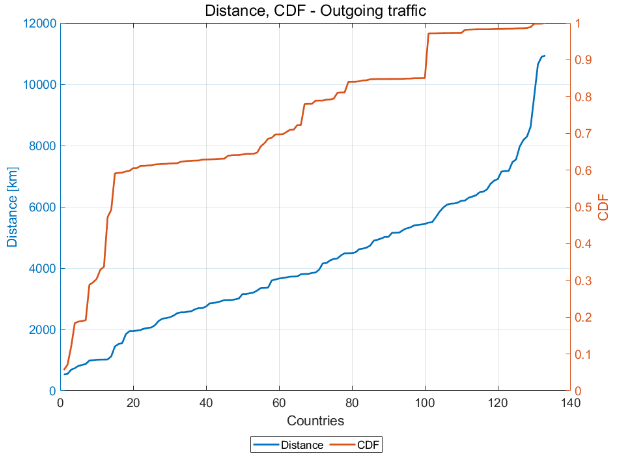


Figure 4.16: Outgoing traffic based on ordered distance.

5000 *km*, and *c*) 5000 - 6000 *km*.

The major outgoing traffic increasing is quite noticeable for the first 15 nearest destinations to the reference one, where most of these destinations are neighboring countries, directly or indirectly connected via their borders; these countries generate more than 60 percent of the total traffic.

These countries geographically share many cultural behaviors; common relating ethnicities or tribes that only political border separates them. Politically speaking, business, tourism and economic relations are among the most important factors that neighboring countries share. Indirect neighbors are those countries that the neighboring countries lie in between. Those countries are also tightly linked through their business, tourism, or economic sectors.

The countries lying in the second region, 3000 - 5000 *km*, take about 20 percent of the total traffic. These countries feasibly share historical (e.g., migration) or trading, business and tourism aspects with the reference country.

In the third region, 5000 - 6000 *km*, countries produce about 10 percent of the total traffic. This portion can be seen as relatively high compared to more distant countries, but a closer inspection reveals that only a few countries contribute to this traffic. The relationship of such countries with the reference country can be either due to tourism or business.

The remaining countries, above 6000 *km*, contribute to the total traffic in about 5 percent. Those countries are typically out of major interest for the

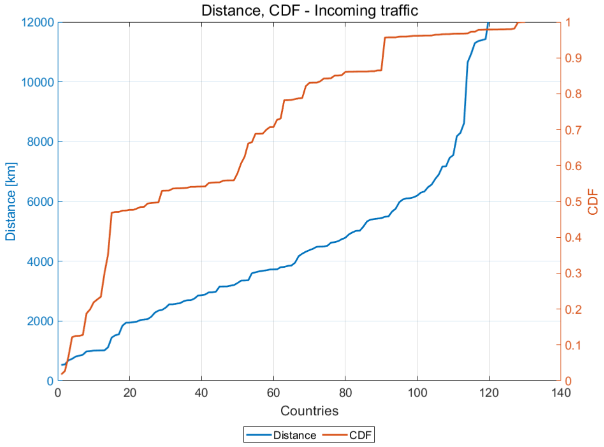


Figure 4.17: Incoming traffic based on ordered distance.

reference country, where the traffic is mainly due to short-term visits (e.g., tourism).

4.4 Conclusion

This chapter covered two different approaches based on CDR data. Firstly, I analyzed one-year CDR dataset (July 2016-June 2017), presenting international traffic of incoming and outgoing calls, for which I describe different traffic profiles based on their periodicity.

The studied area covered a geographical region, comprised of several countries, Statistics of working and weekend days per year are provided as well. Furthermore, I depicted the incoming/outgoing calls distribution among end-user networks, and determined the countries with the highest traffic. The obtained results show a long-term traffic stability and a daily/weekly traffic periodicity that reflects human activities. Additionally, as expected, I observed that a major part of the incoming/outgoing traffic comes mainly from neighboring countries.

In the second half of the study, I investigated in international voice traffic dependency per destination. The analysis was done based on a three-month CDR dataset, which consists of about 9 million CDRs of outgoing/incoming international voice traffic. The countries are in polygon/irregular shapes or large areas, which makes it hard to calculate the distance between two countries. Hence, I was using centroids of countries for references when calculating the distance between two countries.

The distance among countries, more precisely the distance among countries' centroids, was determined by using the great-circle distance approach. To avoid the population traffic effect, data normalization is applied to structure the relational voice traffic data to countries populations.

From the CDR as well as signaling data, I extracted voice traffic parameters such as waiting, service and interarrival time for each country. The obtained results show dependency of voice traffic parameters on the destinations. The nearby countries are responsible for more than 60 percent (resp. 50 percent) of outgoing (resp. incoming) traffic. This is mainly due to historical, cultural, business, tourism and economic factors.

Chapter 5

Voice Traffic Load Characteristics

5.1 Introduction

Mobile networks generate enormous amounts of data every moment, such as signaling and user data (paging, location update, CDR, Short Message Service (SMS), etc.). By analyzing such data, the network performance can be improved, and a better Quality of Services (QoS) to users can be ensured. Additionally, the data can be also utilized for other purposes, such as public transportation planning, public health analysis, social behavior studies, crime investigations, etc. [84]. One of data usage in mobile networks is teletraffic. The teletraffic in telecommunications represents the application of probability theory that is used to support network planning, dimensioning, performance evaluation, operation and maintenance [85]. The objective is to make the traffic measurable, and to quantify the relation between grade-of-service and system capacity [85].

In teletraffic theory, the word "traffic" is usually denoted as the traffic intensity, i.e. the number of calls carried by the network per time unit [86]. International Telecommunication Union (ITU-T) defines the traffic intensity as "the instantaneous traffic intensity in a pool of resources, which is the number of busy resources at a given instant of time" (ITUT B.18).

This chapter is based on CDR and signaling data that are collected from a voice traffic carrier. The topology of the studied scenario is illustrated in Fig. 5.1.

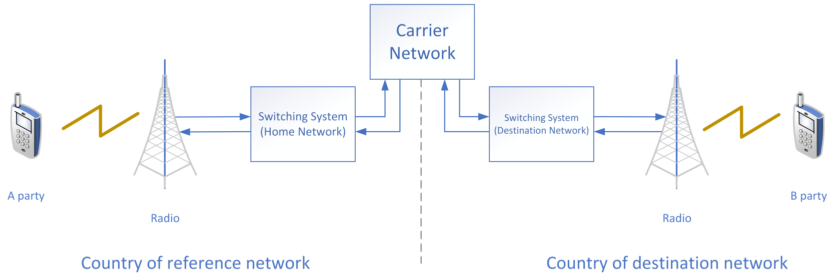

Carrier Network
Switching System (Home Network)
Switching System (Destination Network)
A party
Radio
Radio
B party
Country of reference network
Country of destination network

Figure 5.1: Voice call flow through a carrier network.

The topology consists of home network (hereafter referred as the reference network) that is located in one country while a destination network can be located in any country around the world. Both reference and destination networks are mobile operators. A third network can be involved in communication, and it is called a carrier network. The carrier network interconnects two networks and can be part of a mobile network operator.

5.2 Theoretical approach

In this section, I discuss the theoretical background. Fig. 5.2 illustrates different time phases of a call. Once a call request arrives, denoted as the arrival time t_A the waiting time period, Δt_w is initialized. The Δt_w represents the time interval between the *Connecting* and *Connected* status of a call as described in chapter 3.

At this phase, the A party, denoted as x in Fig. 5.2 is at idle state while waiting to setup the call. Once the A party is connected to the B party, at a moment, which is denoted as connection time t_C, the connection enters the active state.

The active call continues until one of the parties clears the call at the moment t_D (disconnect time). The interval between t_C and t_D is referred as a serving time, Δt_s. Thus, the whole call time period, Δt_x, consists of two periods Δt_w and Δt_s [87]. The time t_A and period Δt_s are among key parameters when analyzing the performance of a queueing system.

Having two consecutive call arrivals, x_1 and x_2, the call arrival x_2 at t_2 does not depend on the call arrival x_1 at t_1 and vice versa (Fig. 5.3). The time period between two consecutive calls is denoted as the interarrival time Δt_i, i.e. $\Delta t_i = t_2 - t_1$.

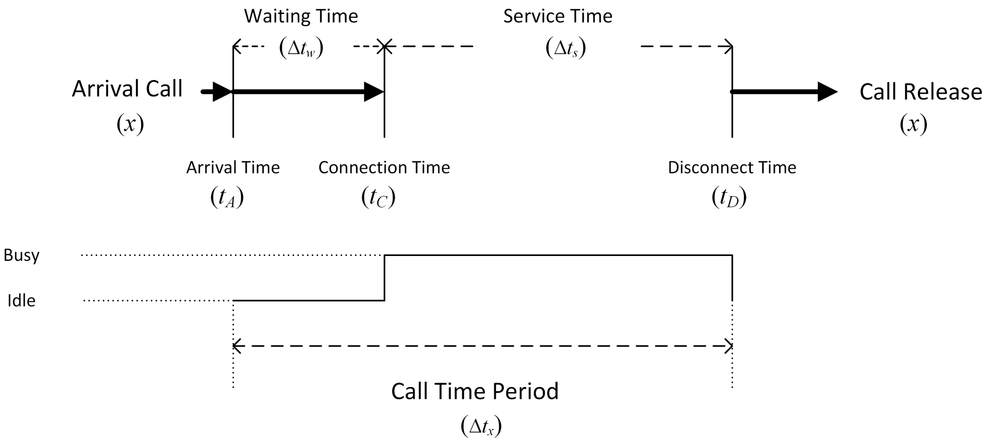


Figure 5.2: Waiting and service time phases.

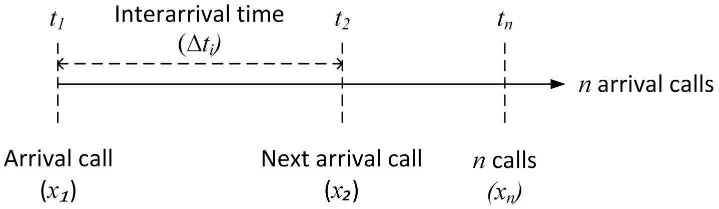


Figure 5.3: Interarrival time process.

The interarrival time basically captures the importance of the destination; the lower the average time between two consecutive calls, the more important the destination is in terms of traffic amount.

In this study, the system is considered to be ideal, i.e. a call arrival, x, starts to be served once arriving to the system. In other words, the call does not have to wait in a queue to be served by system. A delay is only due to the ringing period and the network call processing.

The ringing period in networks varies from a few seconds and can go as high as tens of seconds and has a strong influence on the waiting time. Typically, a maximum ringing time period threshold is configured by the mobile operator; for example, in our reference network, the ringing threshold is set to 30 seconds. The ringing time period depends on several aspects such as the calling party patience, the called party awareness of the incoming call, the importance of the call, the time of the call, etc.

The network call processing time consists of all partial periods before the ringing period starts; e.g., paging, processing, transmission and propagation delay, etc. The network may also perform other tasks such as the security check, the user credential check, the B party availability, finding the route and resources, etc., which increase the total network call processing time.

The number of calls within a time unit follows the Poisson distribution, i.e. the time between two consecutive calls follows the exponential distribution and call arrivals are independent of each other [88][89]. The Poisson distribution indicates the probability that a call arrival may occur in the next given time unit:

$$[h]P_j = \frac{\lambda^j}{j!}\, e^{-\lambda} \qquad (j = 0, 1, ...). \tag{5.1}$$

where j is the number of call occurrences in a time unit, and λ represents the mean value of call arrivals.

The service time Δt_s is modeled using the exponential distribution with the service time mean value represented as β. The probability density function of the exponential distribution is given by following equation [90]:

$$[h]f(t) = \begin{cases} \frac{1}{\beta}e^{-\frac{1}{\beta}n} & \text{if } n \geq 0 \\ 0 & \text{if } n < 0 \end{cases} \tag{5.2}$$

where n is the number of call occurrences in a unit of time. As shown in section VI, the interarrival time and service time are exponentially distributed.

5.3 Results

As mentioned above, I analyze 3-month CDR dataset files, which cover the period October, 2016 - December, 2016. The datasets were collected via a CDR mediation server in the switching center and consists of about 9 million records.

For the analyses, I consider incoming and outgoing traffic of one short-distance and three long-distance international call scenarios: i) one neighboring country to the reference network (country C in the following figures), ii) one European country (country B in the figures), and finally iii) two Asian countries (countries A and D in the figures). A detailed study of traffic distribution among different countries were presented in previous chapter. Additionally, results of the whole world traffic were provided as well, including the empirical and theoretical calculated values.

Notice that the incoming traffic consists of network traffic originating from the whole world while the outgoing traffic only includes traffic originating from one country, i.e. the reference network. In other words, the incoming traffic pool is much larger than the outgoing one.

Table 5.1 indicates mean values theoretically calculated for the outgoing and incoming traffic in the whole world case scenario. The outgoing traffic shows a higher mean value of waiting time with a lower mean value of service time than the incoming traffic. One of the main reasons is the cost of an outgoing international call, which is generally higher than a local call. Moreover, internet services are nowadays globally available, which can be used as an alternative to traditional voice call scenario. As to the interarrival time, the mean value of outgoing traffic is higher as more traffic is generated in the world compared to the amount of traffic originating from the reference network.

Table 5.1: Mean values of the analyzed parameters for the world scenario

Traffic direction	*Waiting time*(λ)[s]	*Service time*(β)[s]	*Interarrival time*(β)[s]
Outgoing	16.26	132.29	0.62
Incoming	13.46	154.06	0.52

Fig. 5.4 shows the probability of service time occurrences. On the right (resp. left) y-axis, the probability of occurrences for the world scenario (resp. selected countries) is presented. The graphs illustrate the frequency of occurrences for the given service time, i.e. the sum of all probabilities when the service time goes to infinity equals to 1 for each country, resp. the world.

As can be observed from Fig. 5.4, the service time is exponentially distributed for all considered scenarios; the figure also specifies the theoretical result for the world traffic scenario.

The majority of considered scenarios show the maximum for the value around 5 seconds and then the probability rapidly decreases as the service time increases. In our case, the service time is affected by many factors, such as the type of calls (private, business calls), the network operator price policy (distant calls are typically cheaper), the distance between the reference and destination network, etc.

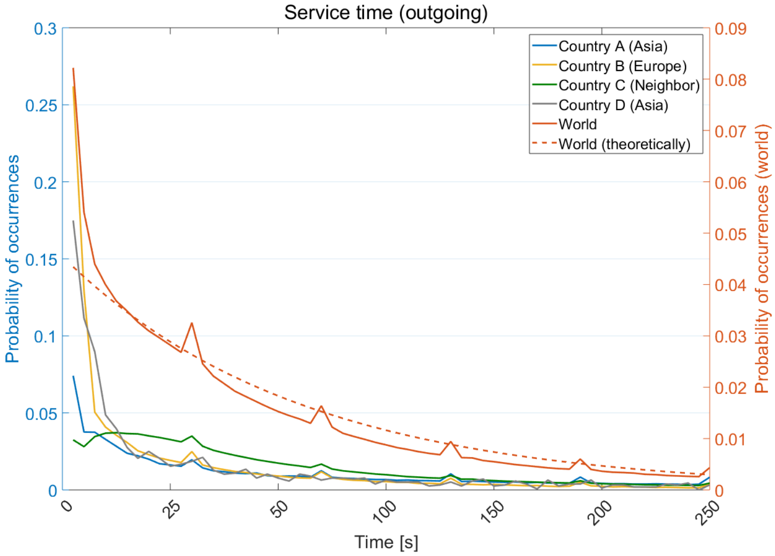


Figure 5.4: Service time, outgoing traffic.

Compared to the long-distance call scenarios, the probability of occurrences of the short-distant call scenario (country C) is smaller, but the probability is relatively constant, up to 60 seconds. In other words, the probability of having a longer call duration is higher for the short-distance call scenarios than for the long-distance ones. This is typically due to the strong business and/or personal relationships among the neighboring countries.

In Fig. 5.5, the waiting time outcome is shown. Correspondingly to Fig. 5.4, the right y-axis illustrates the probability of occurrences for the world scenario case, whereas the left y-axis, for the 4 selected countries. Based on the scenario, the peaks occur between 6 to 21 seconds.

The waiting time in case of neighboring countries (country C), and the country A, follows roughly the world scenario. Whereas in case of countries B and D, a slightly different curve progress is observed, with the peak around 3 seconds. This is possibly due to:

- A user intentionally clears the call after the called party is being rung, as the ringing is only used to notify the called party.
- The called party relatively quickly hangs up as the party knows in advance about the incoming call and expects the ringing.

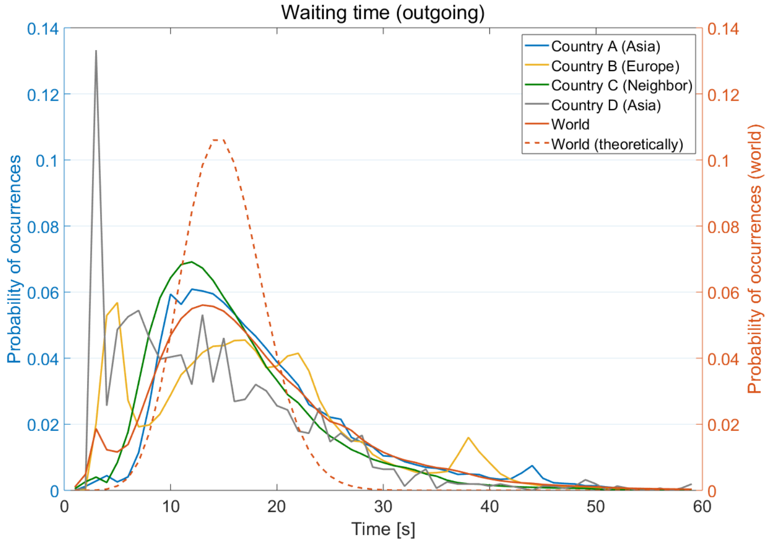


Figure 5.5: Waiting time, outgoing traffic.

The theoretical world traffic scenario shows a higher waiting time within the interval 10 to 20 seconds and a lower one within the interval 30 to 60 seconds compared to the real values. Such variations between the empirical and theoretical values are due to the nature of human behavior around the world; for instance, lifestyle, network policies, time zone difference, cultural differences, etc.

Fig. 5.6 shows the interarrival time, where the description of x-y axes is analogous to the previous figures. I can see in the graph that the peak of interarrival time for the world scenario is very small, close to 0s. According to table 1, the probability of having 2 consecutive calls in the world scenario within 1 second is about 0.6.

The lowest interarrival time manifests the neighboring country C, which is due to the strong relationships among the neighboring countries as explained previously. The interarrival time grows as destinations are farther and farther (see countries A, B and D). For example, the interarrival time between two calls in case of a faraway destination could reach even as high values of 10^6 seconds.

Additionally, based on the daily/weekly traffic profiles, I can deduce the country relationship nature, i.e. if the relationship is much more business or personal oriented.

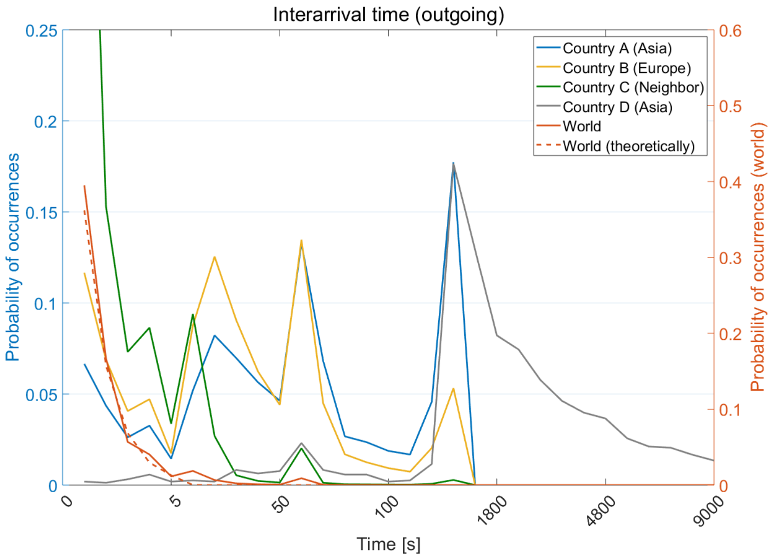


Figure 5.6: Interarrival time, outgoing traffic.

Fig. 5.7 illustrates the service time for the incoming traffic case. In general, the probability of occurrences is higher for the incoming traffic than for the outgoing one (shown in Fig. 5.4).

As the incoming calls are from a much larger pool of users (the whole word, or all network operators in the given countries), a call with a longer service time typically occurs more frequently.

The waiting time of incoming traffic is illustrated in Fig. 5.8. The waiting time of incoming traffic for countries A-D manifests higher peaks compared to the outgoing traffic case (Fig. 5.5). In case of world scenario, the theoretical values are, similarly to the outgoing scenario, higher than the real ones. The reason is the same as explained in Fig. 5.5.

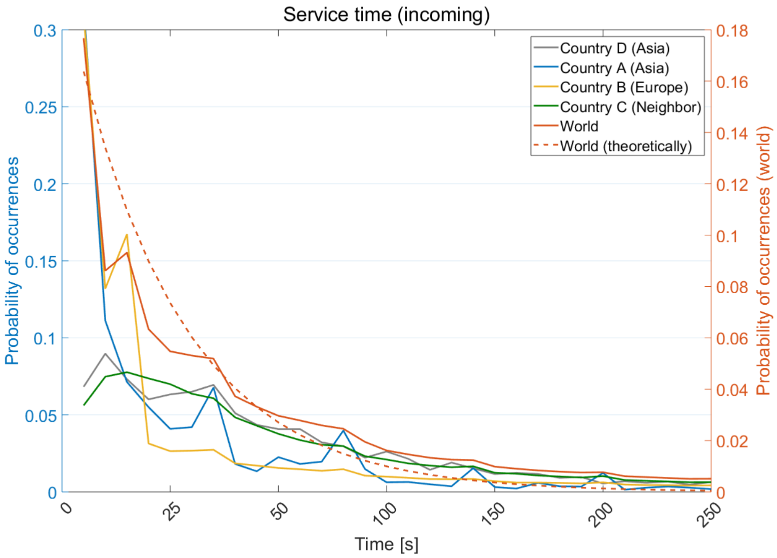


Figure 5.7: Service time, incoming traffic.

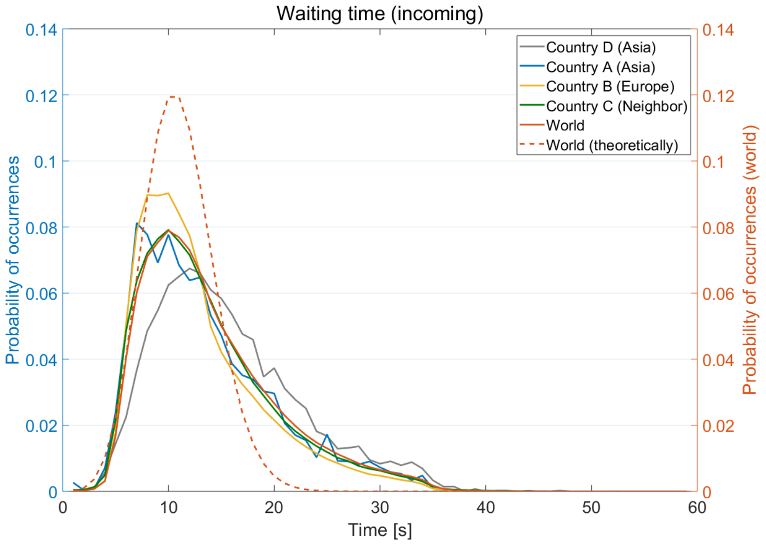


Figure 5.8: Waiting time, incoming traffic.

As to the interarrival time, the probability of occurrences shows similar behavior for the outgoing (Fig. 5.6) and incoming traffic (Fig. 5.9), where the probability is higher for the incoming interarrival time (due to the larger pool of users). However, compared to Fig. 5.6, the incoming interarrival time peaks are shifted to right, resp. left (around 500s, resp. 10s) for the countries *A* and *D*, resp. *B* and *C*.

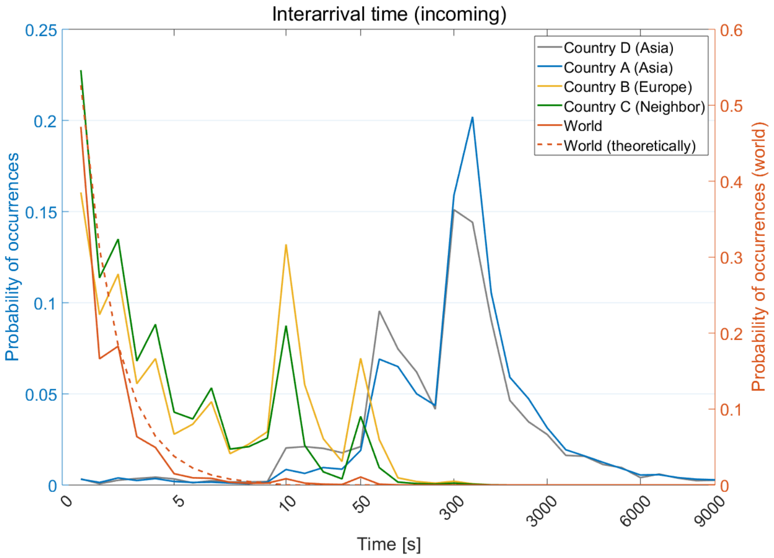


Figure 5.9: Interarrival time, incoming traffic.

5.4 Conclusion

In this chapter, I proposed a study on characteristics of the interarrival, waiting, and service time for outgoing and incoming traffic of an international voice traffic carrier. The analyzed CDR dataset covers 3 months (October-December 2016).

For both traffic, outgoing and incoming, the interarrival, and service time follow the exponential distribution. The interarrival time increases as the destination to the reference network becomes less important. A longer service time can be observed from the nearby countries to the reference country.

The waiting time follows the Poisson distribution, and it varies based on the network configuration, the user resilience, or the call importance.

In the world scenario case, the theoretical values show higher peaks than the real measurements. Such difference is due to the nature of human behavior around the world; for instance, lifestyle, network policies, time zone difference, cultural differences, etc.

Chapter 6

Modeling Voice Traffic Patterns

6.1 Introduction

MObile network data are constantly used nowadays for better understanding of users' needs and improving the quality of services. Meanwhile, the demand for user data is increasing due to the importance of the content that offers crucial information about users' behaviors and mobility in the network. However, telco networks generate on daily basis a huge amount of data originating either from users or the network itself through signaling messages. Signaling is the process of using the signal to control communications among the mobile network elements as well as documenting the network activities and user details. One of the important data that are provided by users is CDR.

Dimensioning of mobile networks usually depends on the number of users and their demands. The number of users is a measure of telco marketing success, available services, offers, and quality of services. In contrast, user demand is an indicator of overall user satisfaction improvement. Alternatively, there are two main criteria that designate the network services availability (i.e., voice calls); the number of calls and the average call duration. Though, the duration of calls can also signify the quality of the call. The longer call duration implies a better quality of the call [91]. Nonetheless, there are several problems that the network faces while handling the needs of users such as peak hours throughout the day. This is happening when the highest traffic load hits the network that might lead to network congestion when the network cannot handle more traffic, or critical incidents such as events that users request to use the network intensively. Another network issue can be low coverage, which increases with the city and urban expansion along with population growth, and that leads to low quality of services and poor signal reachability.

The outliers or anomalies in the mobile network are also among the major issues that the network encounters, possibly at any time. Network anomalies can come from a network malfunction (per specific location or cell), defects, misconfiguration, power outage or software errors, which leads to network outage. Consequently, the network fails to satisfy the users' needs.

In this chapter, I discuss the modelling of voice traffic patterns and profiles from CDR data in mobile networks to address user's behavior and network anomaly detection. The metadata is provided by a mobile network that has deployed numerous network generations including the latest network generations.

I propose a novel study using multi-algorithm approach to achieve qualitative outcomes. The workload comes with two phases. First phase derives the definition of ordinary or normal voice traffic behaviors and patterns out of the data in the mobile network using Gaussian Mixture model (GMM) [92] and Mean shift clustering. The technique is to target certain attributes separately using different unsupervised learning algorithms. Unsupervised learning [93] is the method of structuring points into groups that are similar in certain ways or determining the procedure of data distribution in the space known as density estimation [94].

In the second phase I introduce four distinct algorithms to detect and predict anomalies. I deseasonalize the data to obtain a higher accuracy followed by the distribution function to comprehend the patterns in the data. The first algorithm is Z-score thresholding, which is the representation of standard deviations from the mean value whether it is above or below in the data that are normally distributed.

The second algorithm is Isolation Forest. This is an unsupervised learning algorithm based on decision tree algorithm. It uses isolation of anomalies through selecting a feature in a provided set randomly, and then chooses a separate value between maximum and minimum values of that feature. This leads to generate shorter paths in the trees to values that are presumably considered outliers or anomalies. The third algorithm is density-based spatial clustering of applications with noise (DBSCAN). This algorithm identifies the groups of data based on their density in a set of features. The concept of DBSCAN requires certain parameters and types of data points. The fourth algorithm is the K-means clustering.

I evaluate the results from each algorithm based on six criteria: i) Recall score is defined as the number of true positives divided by the sum of true positives and false negatives. ii) Precision is in return the true positives divided by the sum of true positives and false positives. iii) F1-score is combination of recall and precision scores in a model. iv) Receiver Operating Characteristic-Area Under the ROC Curve (ROC-AUC) represents the performance of a model. v) Accuracy score is the evaluation metric to measure the correct predictions to the total number of predictions. vi) Efficiency of the algorithm that consists of computational time and memory usage.

6.2 Research Methodology

This section discusses the methodology used in this study. I exploit one year of CDR data that hold 37 million calls. After the data are preprocessed. The study goes through two phases. In the first phase, I get familiarized with the voice traffic profiles and attributes, followed by the linear function between call duration and the number of calls, and then I represent normal voice traffic patterns in the network by using certain algorithms to target certain parts or attributes in the data.

GMM is used to define the clusters of call duration that users intend to select. Mean Shift clustering on the other hand is employed to identify the expected daily peak-hour patterns from abnormal traffic spikes that might happen due to network issues (misconfiguration, malfunction, mismatch, etc.), incidents on national or regional level, malicious intent, etc.

In the second phase of the study, I visualize the data of the entire dataset. However, at this point, I deseasonalize the data for higher accuracy achievement, trend and irregular component exploring (see Fig. 6.1). Then, the normal distribution of number of calls and total call duration along with average call duration are illustrated to comprehend the patterns in the data. Next, Z-score, Isolation Forest and DBSCAN algorithms are used to detect and predict anomalies and outliers. Finally, I evaluate the outcome from the algorithms based on several performance metrics to measure the accuracy of each algorithm.

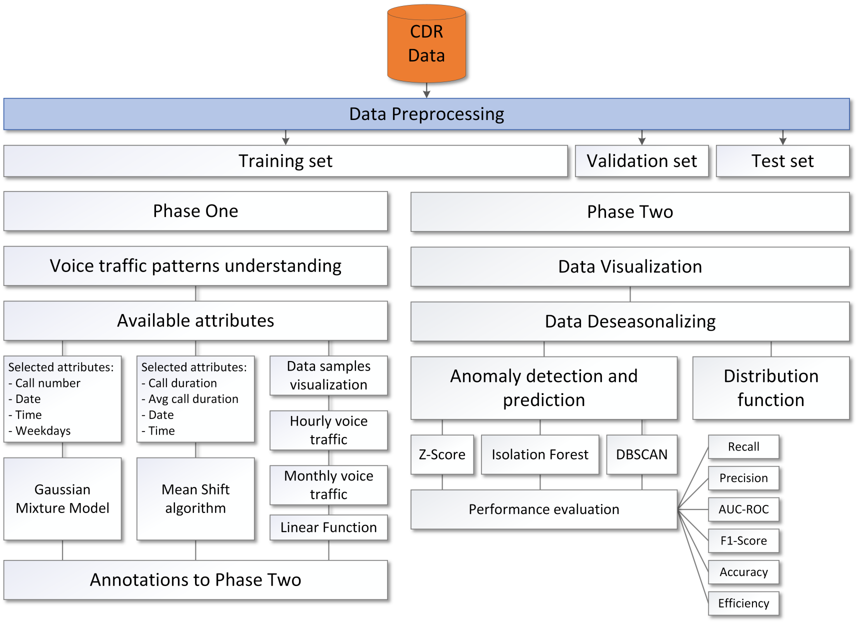


Figure 6.1: Research methodology.

6.3 Theory of Algorithms

In this section, I explain the mathematical approach behind the algorithms used in this study along with the steps that are used to achieve the final output of the algorithm.

6.3.1 Gaussian Mixture Model

In real-world data, a single Gaussian distribution cannot handle a mixture of several stochastic processes [95]. Besides, I don't use K-means clustering since it relies on only one component, which is the mean of the cluster. Subsequently, there is a problem when it comes to multiple clusters with overlapping means and different covariance matrices. Therefore, GMM is used to explain K Gaussian distribution.

GMM is a probabilistic semi-parametric distribution-based soft clustering model. It assumes all data points are generated from a mixture of a finite number of Gaussian distributions with unknown parameters. GMM builds on three parameters [96]: the number of clusters K, mean, and covariance. The parameters are determined by maximum likelihood, normally utilizing the Expectation Maximization (EM) algorithm [97]. EM consists of two steps: E-step and M-step. At any given GMM, the objective is to maximize the likelihood function.

To theoretically explain this model, I assume I have a dataset of d-dimensional data that a multivariate Gaussian distribution can be used. In that case, the probability density function will be [98]:

$$N(x;\mu,\Sigma) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right) \tag{6.1}$$

Where:
x represents N sets of multi-dimensional data
μ is the mean of each dimension
$\sum$ is the covariance.
Let's assume there is a training set of N points in a dataset where:

$$x = x_1, x_2, \ldots, x_i, \ldots, x_N$$

However, x_i is a multi-dimensional in our case, then I need to employ multivariate GMM:

$$p(x \mid \Theta) = \alpha_k \frac{1}{\sqrt{(2\pi)^d\left|\Sigma_k\right|}} \exp\left[-\frac{1}{2}\left(x-\mu_k\right)^T \sum_k^{-1}\left(x-\mu_k\right)\right] \tag{6.2}$$

Θ represents all parameters and α_k is the prior probability of kth Gaussian model.

In order to estimate the GMM parameters as I assumed the dataset is a mixture of K Gaussian distributions, I use Maximum Likelihood Estimation (MLE). The objective is to maximize the likelihood to achieve the best set of parameters (αk, μk, Σk). The likelihood function and MLE respectively are [99]:

$$p(x \mid \Theta) = \prod_i p\left(x_i \mid \Theta\right) \tag{6.3}$$

$$\Theta = \operatorname{argmax}_\Theta \prod_i p\left(x_i \mid \Theta\right) \tag{6.4}$$

Then, the likelihood function for multivariate after MLE is:

$$p(x \mid \Theta)=\prod_i p(x \mid \Theta)=\prod_i\left[\sum_k \alpha_k N\left(x_i \mid \mu_k, \Sigma_k\right)\right] \tag{6.5}$$

At this point, I use EM algorithm to simplify the GMM likelihood function. Firstly, I take the log of the likelihood function as monotonically maximizing function of its argument:

$$L(x \mid \Theta)=\sum_i \ln \left[p\left(x_i \mid \mu_k, \sigma_k\right)\right]=\sum_i \ln \left[\sum_k \alpha_k N\left(x_i \mid \mu_k, \sigma_k\right)\right] \tag{6.6}$$

This is when I reach a point where the single Gaussian reaches its limit after taking the derivatives with respect to the k th mean to 0 . Therefore, the EM introduces latent parameter z and provides a new set of parameters for GMM to explain the probability of $p\left(z \mid x_i, \mu_k, \sigma_k\right)$. The probability distribution of xi with introduction of z is:

$$p\left(x_i \mid \Theta\right)=\sum_k p\left(x_i \mid z=k, \mu_k, \sigma_k\right) p(z=k) \tag{6.7}$$

I introduce the latent parameter into the log likelihood function:

$$L(x \mid \Theta)=\sum_i \ln \sum_k p\left(z=k \mid x_i, \mu_k, \sigma_k\right) \frac{p\left(x_i \mid z=k, \mu_k, \sigma_k\right) p(z=k)}{p\left(z=k \mid x_i, \mu_k, \sigma_k\right)} \tag{6.8}$$

To simplify the likelihood function, I recall Jensen's inequality [100] to simplify a function in an EM process:

$$f[E(x)] \geq E[f(x)] \tag{6.9}$$

Therefore, Jensen's inequality and the posterior probability derived by the Bayes' law:

$$L(x \mid \Theta) \geq \sum_i \sum_k \omega_{i, k}^t \ln \frac{\alpha_k N\left(x_i \mid \mu_k, \sigma_k\right)}{\omega_{i, k}^t} \tag{6.10}$$

Here, I specify the iterative function with denoted t for the EM algorithm. This will result in Θ^t and latent $\omega_{i,k}^t$. With the latent parameters applied in the iterative function $Q\left(\Theta, \Theta^t\right)$ and using maximization to update Θ^{t+1}.

Now, I employ the EM for multivariate GMM [101]. The E-step is to estimate the latent parameters:

$$\omega_{i,k}^{t}=\frac{\alpha_{k}^{t} N\left(x_{i} \mid \mu_{k}^{t}, \Sigma_{k}^{t}\right)}{\sum_{k} \alpha_{k}^{t} N\left(x_{i} \mid \mu_{k}^{t}, \Sigma_{k}^{t}\right)} \tag{6.11}$$

The M-step is maximization step with iteration t + 1 to update the parameters $\left(\alpha_{k}^{t}, \mu_{k}^{t}, \Sigma_{k}^{t}\right)$:

$$\alpha_{k}^{t+1}=\frac{\sum_{i} \omega_{i,k}^{t}}{N} \tag{6.12}$$

$$\mu_{k}^{t+1}=\frac{\sum_{i} \omega_{i,k}^{t} x_{i}}{\sum_{i} \omega_{i,k}^{t}} \tag{6.13}$$

$$\Sigma_{k}^{t+1}=\frac{\sum_{i} \omega_{i,k}^{t}\left(x_{i}-\mu_{k}^{t+1}\right)\left(x_{i}-\mu_{k}^{t+1}\right)^{T}}{\sum_{i} \omega_{i,k}^{t}} \tag{6.14}$$

The final step is to evaluate the log likelihood function and I investigate the convergence of the parameters or the log likelihood function. If the convergence criterion is not satisfied, I return back to E and M steps and again log likelihood function evaluation.

6.3.2 Mean Shift Clustering

Mean Shift (MS) is a non-parametric density-based unsupervised learning clustering algorithm through allocating the data points to the clusters iteratively, shifting the points to the mode, which is with highest data points intensity. The shifting is done until the points converge to a local maximum of the density function. Mean Shift is also known as mode-seeking algorithm. follows:

1. Set a sliding window for the data points.
2. Every sliding window is shifted towards higher density of distributed points through shifting the centroids to the mean. I repeat until no more shifts can generate a higher density.
3. I delete overlapping windows. However, when there are several overlapping occurrences, I keep the window with highest data points, and I delete the rest.

4. Allocating the data points to the belonging sliding window.

Mean Shift is built based on the Kernel Density Estimation (KDE) [102], which is a method to indicate the underlying probability density functions. xi is a finite number of data points and h is size of window.

$$\tilde{f}(x) = \frac{1}{nh^d} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) \tag{6.15}$$

And the Mean Shift vector:

$$m(x) = \frac{\sum_{i=1}^{n} x_i g_i}{\sum_{i=1}^{n} g_i} - x \tag{6.16}$$

$$g_i = g\left(\left\|\frac{(x - x_i)}{h}\right\|^2\right) \tag{6.17}$$

$$g(r) = k'(r) \tag{6.18}$$

Then, the succeeding location of the kernel is [103]:

$$y_{i+1} = \sum_{i=1}^{n} x_i g\left(\left\|\frac{y_i - x_i}{h}\right\|^2\right) / \sum_{i=1}^{n} g\left(\left\|\frac{y_i - x_i}{h}\right\|^2\right) \tag{6.19}$$

6.3.3 Z-Score

Z-score [104] is also known as the standard score that provides the positions of data points from the mean and how they are related to each other. It is measured based on the standard deviation from the mean value. It can be formulated as follows:

$$z = \frac{(x - \mu)}{\sigma} \tag{6.20}$$

z = Z-score
x = the value being evaluated
μ = the mean
σ = the standard deviation

6.3.4 Isolation Forest

Isolation Forest algorithm [105] is a technique that uses an assortment of isolation or binary trees to detect anomalies. It is constructed using data recursive segmentation. Each segmentation split is made by selecting a feature and a splitting value randomly within the array of available features. The isolation of individual data points and/or reaching a predefined max tree height process is done by iteratively splitting the data. Then, the anomalies isolation is achieved through association with the path-length since anomalies require less splits to be isolated as they are only few and different.

The mathematical approach is attained via the construction of trees and calculation of path lengths. The path length is about the depth of a data point in an isolation tree. As mentioned, the shorter the path length, the data point is more probably an anomaly score. The mean path length can be calculated of n data points as follows:

$$c(n) = 2H(n-1) - \frac{2(n-1)}{n} \tag{6.21}$$

The $H(i)$ is the harmonic number that can be estimated as:

$$H(i) = \ln(i) + \gamma \tag{6.22}$$

where γ is the Euler-Mascheroni constant ($\gamma \approx 0.5772156649$).

The anomaly score can be determined after constructing the isolation trees. The formula goes as following:

$$s(x, n) = 2^{-\frac{E(h(x))}{c(n)}} \tag{6.23}$$

where x is an anomaly score for a data point, and $E(h(x))$ is the average path length of x throughout the isolation trees. $c(n)$ is the average path length for a failed search in a randomly created isolation tree.

The score range of an anomaly is between 0 and 1 . A higher value shows a higher probability of being an anomaly.

6.3.5 K-means

In this section, I discuss first the K-means clustering algorithm adaptation. Then, I go through the theory and concepts of the algorithm.

Algorithm Adaptation

Adapting k-means clustering algorithm for anomaly detection and prediction with CDR data has involved several key modifications and considerations:

1. Feature Selection: Instead of using all available features and attributes in the CDR dataset, I carefully select relevant attributes that capture characteristics indicative of anomalous behavior in the mobile network. This includes attributes and features such as call duration, frequency of calls, geographical locations, date and time, etc.

2. Normalization: Since the features in the CDR dataset may have different scales or units, normalization technique has been added to ensure that all features contribute equally to the clustering process.

3. Distance Metric: Since I deal with numerical data, and anomaly detection, common metrics like Euclidean metric is not ideal for anomaly detection due to their susceptibility to biases from duplicate features or irrelevant features that do not effectively predict target attributes. I selected Mahalanobis metrics since it measures the distance of a point from the mean along each principal component in terms of standard deviations.

4. Outlier Handling: Annotations have been provided by network experts for the previously reported incidents along with the historical data to enhance the identification, handling and introducing outlier detection mechanisms.

5. Thresholding: In anomaly detection, it's common to define a threshold to distinguish between normal (hourly/daily/weekly traffic behavior) such as peak hours, recurrent events and anomalous clusters. I introduce thresholding techniques to identify clusters that deviate significantly from the expected behavior, indicating potential anomalies.

6. Iterative Refinement: Given the dynamic nature of mobile networks and evolving security threats, I iteratively refine our adapted algorithm based on feedback from real-world observations and ongoing monitoring of network behavior.

The theory of the algorithm

The K-means clustering algorithm is a popular method for partitioning a given dataset into K distinct, non-overlapping clusters. Mathematically, the K-means algorithm can be described as follows [106]:

- K: Number of clusters
- n: Number of data points
- d: Number of dimensions (features)
- x_i: Data point i (where $i = 1, 2, \ldots, n$)
- c_k: Centroid of cluster k (where $k = 1, 2, \ldots, K$)

Objective: The objective of K-means clustering is to minimize the within-cluster variance, also known as inertia or distortion. It is defined as the sum of squared distances between each data point and its assigned centroid within the cluster.

Mathematical Representation [107]:

1. **Initialization:**
   - Randomly initialize K centroids c_k for each cluster.
2. **Assignment Step (Expectation):**
   - Assign each data point x_i to the nearest centroid based on Euclidean distance:
   $$\text{argmin}_k ||x_i - c_k||^2 \tag{6.24}$$
3. **Update Step (Maximization):**
   - Update the centroids c_k by computing the mean of all data points assigned to cluster k:
   $$c_k = \frac{1}{|S_k|} \sum_{x_i \in S_k} x_i \tag{6.25}$$
   where S_k is the set of data points assigned to cluster k.
4. **Repeat Steps 2 and 3 until Convergence:**

- Iterate Steps 2 and 3 until the centroids no longer change significantly, or a predefined number of iterations is reached.

Objective Function: The objective function of K-means clustering is to minimize the within-cluster sum of squares (WCSS), given by:

$$\text{WCSS} = \sum_{k=1}^{K} \sum_{x_i \in S_k} ||x_i - c_k||^2 \tag{6.26}$$

where S_k is the set of data points assigned to cluster k.

Convergence Criteria: K-means clustering typically converges when one of the following conditions is met:

- The centroids do not change significantly between iterations.
- The maximum number of iterations is reached.

Algorithm 1 k-means clustering

```
Initialise Cluster Centroids
for every iteration $l$ do
    Compute $r_{nk}$:
    for for every data point $x_n$ do
        Assign every data point to a cluster:
        for every cluster $k$ do
            if $\text{k} == \text{argmin} \left\| \text{x}_n - \mu_k^{l-1} \right\|$ then
                $r_{nk} = 1$
            else
                $r_{nk} = 0$
            end if
        end for
    end for
    for every cluster $k$ do
        Update cluster centroids as the mean of each cluster:
        $\mu_k^l = \frac{\sum r_{nk} x_n}{\sum r_{nk}}$
    end for
end for
```

The final output of K-means clustering is a set of K clusters, each represented by its centroid c_k, and each data point assigned to one of the clusters based on proximity to its centroid. The pseudo-code for K-means clustering is presented in Algorithm 1.

The algorithm aims to minimize the within-cluster variance, also known as inertia or sum of squared distances, by iteratively assigning data points to the nearest cluster centroid and updating the centroids to the mean of the data points in each cluster.

6.3.6 DBSCAN

The DBSCAN algorithm [108] stands for Density-Based Spatial Clustering of Applications with Noise. It is a densitybased algorithm marking anomalies that do not lie in any cluster. The idea is to group densely congregated data points into a single cluster.

The algorithm requires two parameters to identify outliers or anomalies. The first parameter is Epsilon (ε), which is the radius of circle that is created around densely clustered data points, and minPoints, which is the minimum number of data points.

A point in the circle can be classified under three categories: Core, Border and Noise. The DBSCAN algorithm locates the data points using Euclidean distance. However, two major concepts are considered before making decisions on any data point: Reachability and Connectivity.

Reachability is about the position of a data point whether it is accessible by another data point (directly or indirectly). Connectivity is to know if two data points lie in the same cluster.

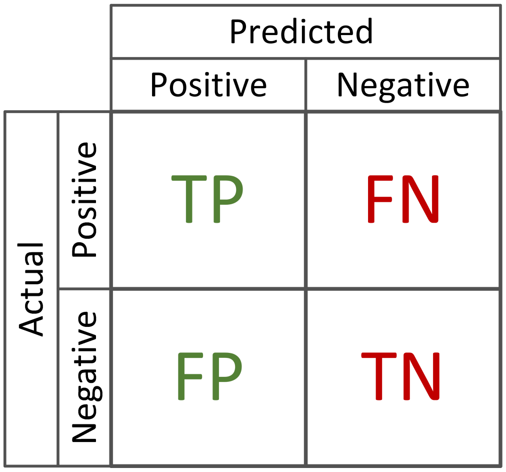
6.4 Performance Metrics

This section is about the evaluation metrics that are applied in the study. These parameters are used specifically in the second phase, which is about anomalies and outlier detection and prediction. These metrics evaluate each algorithm individually. The algorithms are Z-score, Isolation Forest and DBSCAN.

Before I start explaining the parameters, I need to know the confusion matrix, which is the base for defining the assumptions for the evaluation metrics.

6.4.1 Confusion Matrix

It is simply the combination of all possible hypotheses for a predicted and actual value in a matrix form (see Fig. 6.2). There are four criteria in the matrix for both actual and predicted values: True Positives, False Positives, False Negatives and True Negatives.



		Predicted	
		Positive	Negative
Actual	Positive	TP	FN
	Negative	FP	TN

Figure 6.2: Confusion Matrix.

Precision

Precision is the ratio of the true positives to all the positives. It is a useful measure to evaluate the accuracy and performance of algorithms. The metric can be formulated as follows:

$$P = \frac{T_P}{T_P + F_P} \tag{6.27}$$

Recall

It is the measurement of the algorithm correctly identifying the True Positives. This metric can be considered as True Positive Rate to avoid incorrectly identified True Positives. The formula is as follows:

$$P = \frac{T_P}{T_P + F_N} \tag{6.28}$$

F1-Score

This metric is a decision-making presumption that a tradeoff is needed between Precision and Recall metrics based on the needs of a model.

In our case, I would prefer high Recall than high Precision since every detected anomaly should be taken into account and checked further since the possibility of having an anomaly correctly identified will affect the business and huge loss in revenue especially in large voice traffic carriers or mobile operators. For that, I use the Harmonic mean of Precision and Recall, which is called F1-Score.

$$F1Score = 2 * \frac{\text{Precision } * \text{ Recall}}{\text{Precision } + \text{ Recall}} \tag{6.29}$$

ROC-AUC

ROC stands for Receiver Operating Characteristic curve, while AUC stands for Area Under ROC Curve. They are both performance measurements. ROC is the probability curve that depends on two parameters: True-Positive Rate (TPR) and False-Positive Rate (FPR). I can define them as following:

$$TPR = \frac{T_P}{T_P + F_N} \tag{6.30}$$

$$FPR = \frac{F_P}{F_P + T_N} \tag{6.31}$$

On the other hand, AUC is the measure of separability. It measures the entire area under the ROC curve. The higher the AUC is, the better is the model at predicting TP and TN. The graph below (Fig. 6.3) explains both ROC and AUC.

Accuracy

The accuracy score evaluates the performance of the algorithm and provides the percentage of the correct predictions made by the algorithm. The more formal formula goes as following:

$$Accuracy = \frac{T_N + T_P}{T_P + F_P + T_N + F_N} \tag{6.32}$$

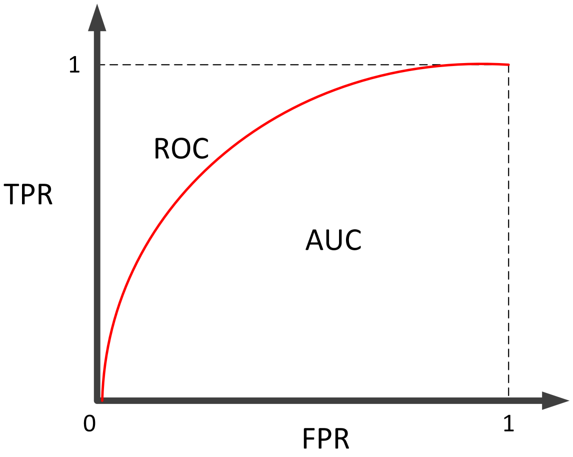


Figure 6.3: ROC-AUC graph.

6.5 Results and discussion

I discuss in this section the results from the algorithms are provided and discuss them further. The results are separately treated based on the input of the algorithms, and that is done in two phases. In the first phase, I discuss the normal behavior of voice traffic profiles based on two algorithms. GMM is used to provide the call duration profiles that customers intend to have throughout the day. MS on the other hand, takes care of finding the peak hours on a daily basis. Furthermore, I provide samples of the usual daily number of calls as well as the monthly number of calls and call duration. In addition, I present the linear function between number of calls and call duration.

In the second phase, I show the distribution function of number of calls, call duration and mean call duration. Finally, I demonstrate the results from the algorithms used to detect and predict anomalies based on annotations made during extracting the CDR data along with the evaluation metrics.

6.5.1 Phase One

In Fig. 6.4, an hourly basis of a normal daily traffic in the network is illustrated. The number of calls is presented on y-axis and the time on x-axis.

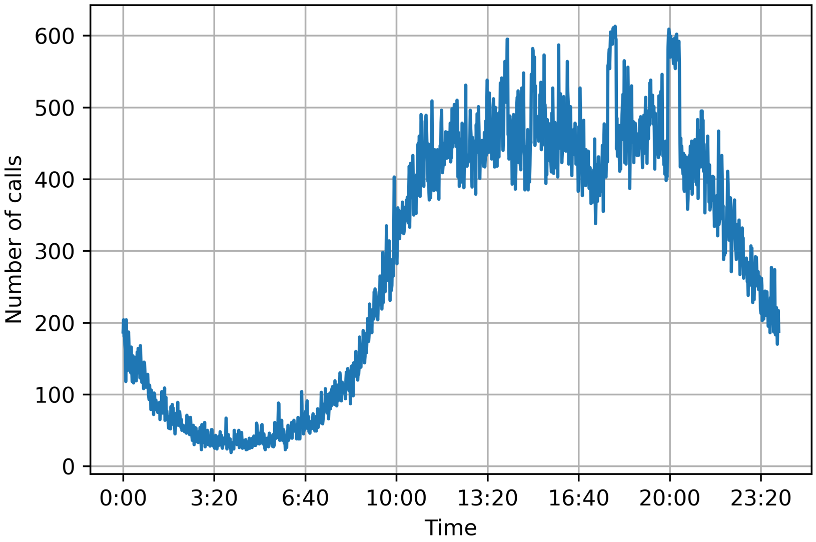


Figure 6.4: Hourly basis number of calls.

I can see that the traffic slowly goes down starting from midnight until early in the morning due to sleeping hours and no working hours. The traffic noticeably goes back up starting from 6:00 morning until it encounters the first peak hour from 13:0014:00 afternoon. There is slightly lower traffic than the peak hours from 16:00-17:00 afternoon. However, that is for a short time, which can be due to the end of working hours. I encounter the second peak hour in the evening around 18:0020:00. Then, the traffic starts to decrease until the next morning.

In Fig. 6.5, I present a three-month based traffic sample to explain the monthly traffic patterns. The number of calls is presented on y-axis and the days on x-axis. Here I can see the effects of weekdays and weekends along with seeing two major anomalies in the graph annotated. A strong seasonal pattern can be seen in the graph, and that is due to having lower traffic during the weekends and higher during the weekdays. Additionally, the traffic shows rhythmic patterns that reflect the customers' behavior on a long-term daily basis.

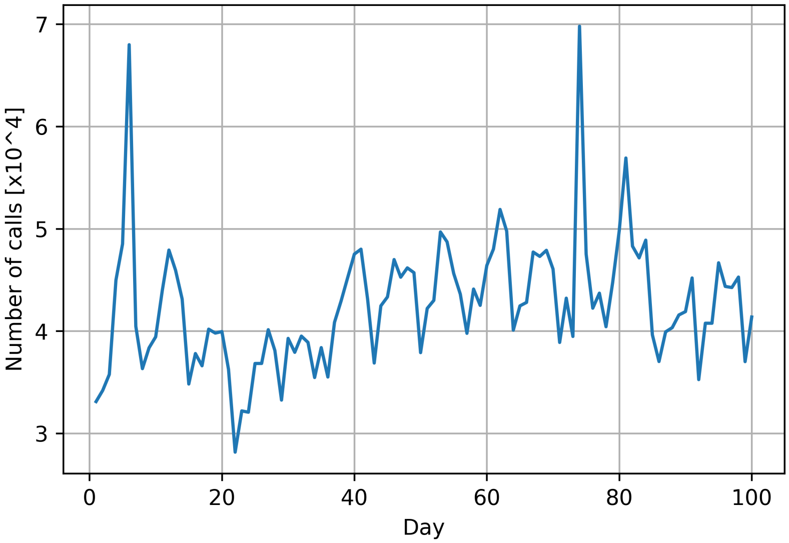


Figure 6.5: Monthly number of calls.

In contrast, the two major spikes in the graph are anomalies that can be differentiated from the rest of the peaks. Anomalies are defined as deviations from the normal behavior in the network. There are various reasons that can cause such conducts like network issues, events, malicious attacks, etc. Detecting and/or predicting such behaviors in time saves the network from huge impact on revenue and business loss. However, anomalies can be negative, which means lower traffic than usual that can be seen on 23^{rd} day.

Table 6.1 shows the statistics of our entire CDR data. In the table I have weekdays and weekends number of calls along with mean and standard deviation per each day. I can see the majority of traffic lies on weekdays and least on weekends.

Table 6.1: Statistics of daily voice calls

Days	*Total calls per year*	μ [calls]	σ [calls]
Monday	5725454	108027	18898
Tuesday	5590481	105481	14609
Wednesday	5624193	106117	16934
Thursday	5401519	101915	13508
Friday	4625490	87273	11869
Saturday	5066378	95592	14301
Sunday	5065466	95575	10340

In Fig. 6.6, the duration of calls is presented for the same selected days in Fig. 6.5. On y-axis I have call duration, and the days are presented on the x-axis. Similar pattern can be noticed in Fig. 6.6 as seen in Fig. 6.5. Hence, I calculate the function between the number of calls and the call duration to understand the relation between call duration and the number of calls.

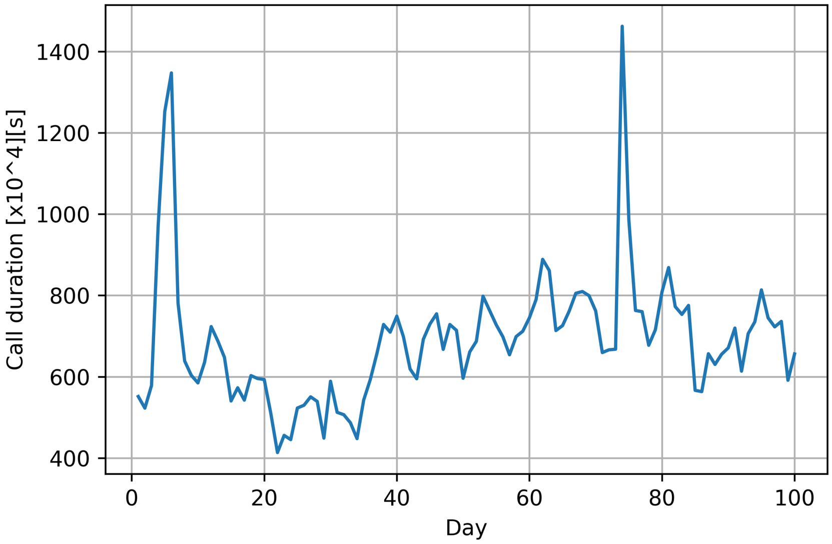


Figure 6.6: Monthly total call duration.

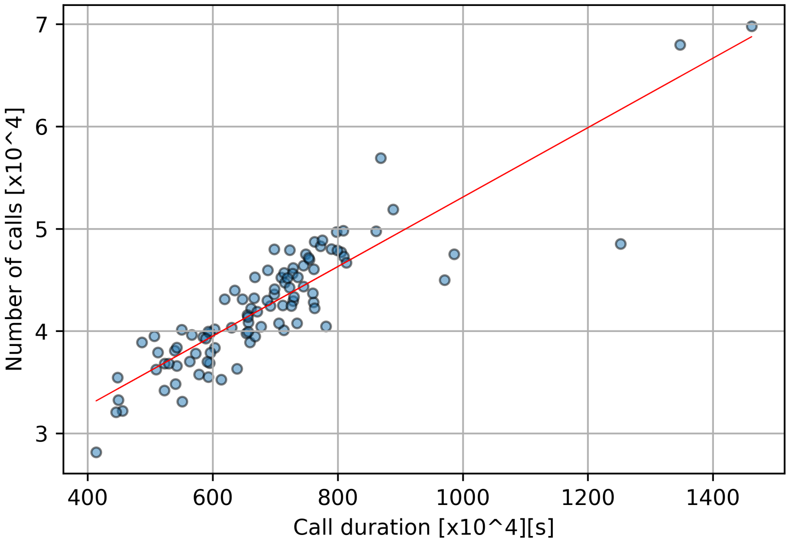


Figure 6.7: The linear function between call numbers and duration.

A linear function is determined when I try to find the function between the number of calls and call duration as seen in Fig. 6.7. I can say based on Fig. 6.7 that the number of calls are proportional to the call duration. The majority of findings are all placed linearly with the expection of a few outliers.

As now I are familiar with the traffic patterns, I use GMM to calculate the clusters of call duration that majority of customers have in the network. In Fig. 6.8, I see there are four major clusters of call duration.

The four major clusters of call duration are the mean call duration, and they are ordered from short to long:

- The first cluster shows 125.35 seconds, which can be defined as a standard average call duration. It is indicated as blue in Fig. 10.

- The second cluster is 142.81 seconds, which is the second highest percentage of call duration that customers have.

- The third cluster is 154.13 seconds. This is the lowest percentage among all other clusters.

- The final cluster shows 188.67 seconds with the third highest percentage of call mean call duration.

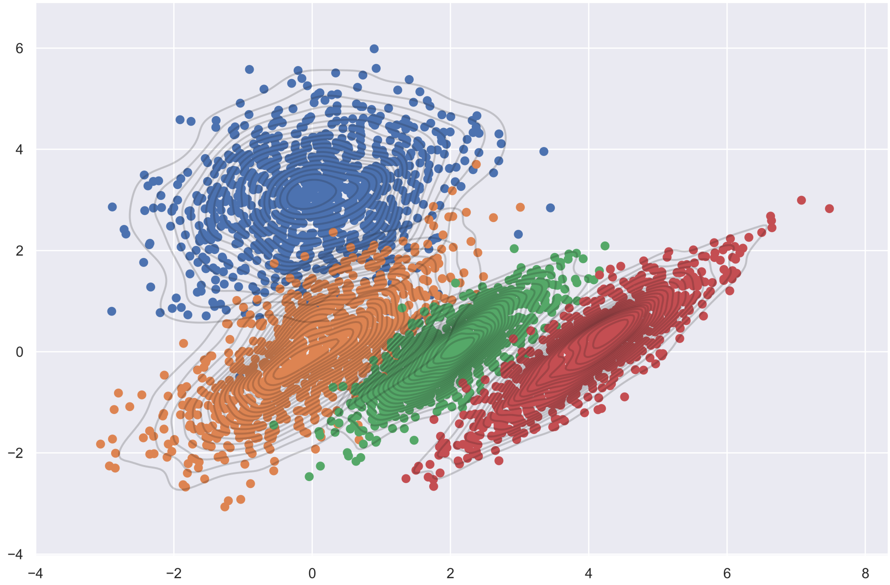


Figure 6.8: GMM for call duration cluster determination.

There are calls with more than 60 minutes duration. However, the calls cannot go beyond 120 minutes duration due to the settings made in the network. Other than that, there are outliers of course as can be seen in the Fig.

In Fig. 6.9, I have MS algorithm presented to determine the peak hours during the day for a hundred days-based data. On xaxis, the days are presented, and y-axis shows the time of the day. Based on the study, the average peak hours mainly are located around two different timestamps. The first one occurs 11:00-14:00, and the second one around 18:00-20:00.

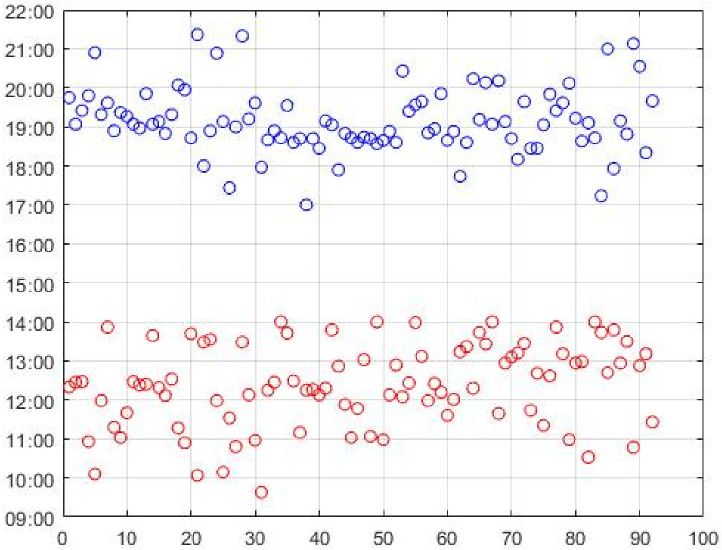


Figure 6.9: MS algorithm for peak hours.

6.5.2 Phase Two

Now, I have basic understanding of the normal voice traffic behavior and patterns in the network based on phase one.

It is essential to have the background of normal voice traffic forms in order to extinguish the abnormal forms and behaviors. Moreover, I want to detect them and predict them using certain algorithms. The parameters that are used at this phase are: Date, Time, number of calls, call duration and average call duration. I visualize first the data in Fig. 6.10.

The top graph shows the average call duration per day, the data as mentioned are one-year based on July 2017-June 2016. The second graph displays the number of calls per day for the entire year. The bottom graph is the total call duration.

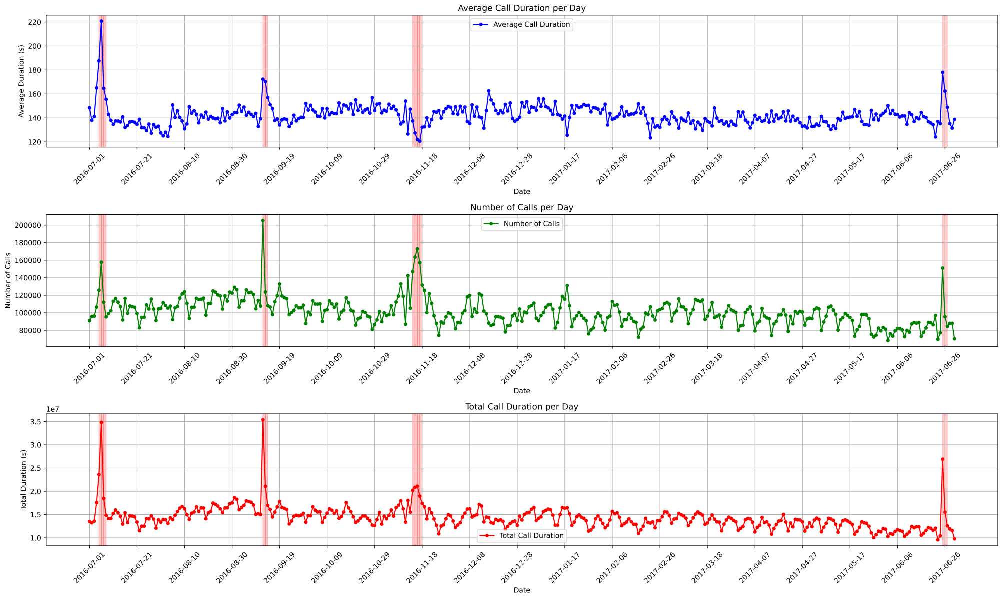


Figure 6.10: Data visualization.

However, the data is seasonal based. To avoid complexity and have higher accuracy results, I deseasonalize the data or seasonal adjustment (see Fig. 6.11). This is a statistical method to remove seasonal components since our main objective is to detect and predict anomalies.

Deseasonalization is a process used to remove the seasonal patterns or fluctuations from a time series dataset. These seasonal patterns often repeat in a regular and predictable manner over a specific period, such as daily, weekly, monthly, or yearly cycles. Deseasonalization aims to isolate the underlying trend and irregular components of the data, making it easier to analyze and interpret.

To deseasonalize our data, the following steps have been followed:

1. Identification of Seasonal Patterns: Initial examination of the time series data to identify recurring seasonal patterns or fluctuations. There are certain patterns in our data that are following daily, weekly and monthly trends.

2. Estimation of Seasonal Component: Application of appropriate techniques, such as moving averages or seasonal decomposition methods, to estimate the seasonal component of the data.

3. Using Additive Model: Additive model decomposes a time series into three components: trend, seasonal, and residual (or irregular). The seasonal component is estimated by averaging the values of the data over each seasonal period (e.g., monthly averages for monthly data) and subtracting these seasonal averages from the original data to obtain the deseasonalized series. Additive model explicitly separates trend, seasonal, and irregular components, providing a clearer understanding of underlying patterns. In addition, it can handle different types of seasonal patterns, including multiplicative ones.

4. Subtraction of Seasonal Component: Removal of the estimated seasonal component from the original data to obtain the deseasonalized data.

5. Analysis of Deseasonalized Data: Examination of the deseasonalized data to identify the underlying trend and any remaining irregular components.

Overall, deseasonalization is a critical preprocessing step in time series analysis that helps to isolate and analyze the underlying trend and irregular components of the data by removing the seasonal patterns or fluctuations. It allows for a clearer understanding of the underlying patterns in the data and facilitates more accurate analysis and modeling.

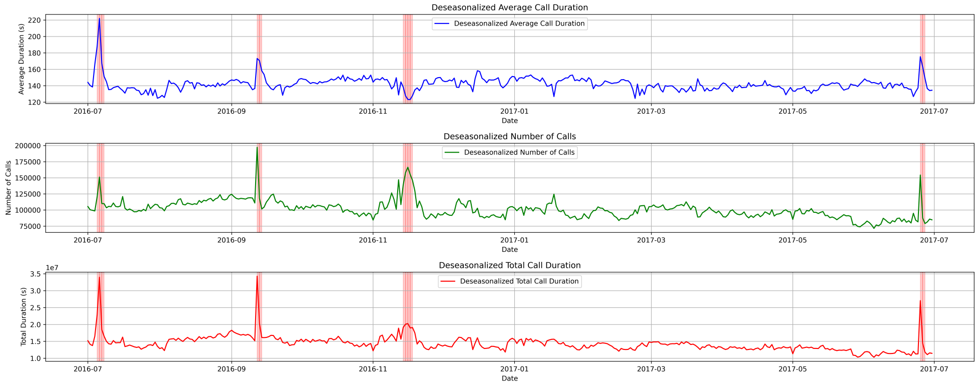


Figure 6.11: Data deseasoanlization.

Once I have the data deseasonalized, I visualize the distribution of the data. Data distribution deals with the frequency of the event occurrences within a certain interval. In Fig. 6.12, I have the distribution of average call duration, number of calls and total call duration respectively. The distribution shows Poisson distribution, which determines the likelihood of an event occurring over a period of time or distance. The events are independent of each other, with no limit on the time of occurrence as stated in Poisson distribution.

The distribution of data can be visualized using graphical representations such as histograms, box plots, and probability density functions. In Fig. 6.12, I present the distributions of average call duration, number of calls, and total call duration, respectively. These visualizations help in assessing the shape, spread, and skewness of the data distribution.

Data distribution is a fundamental aspect of data understanding that provides insights into central tendency, variability, shape, outliers, relationships between variables, and modeling assumptions.

Analyzing data distribution helps in summarizing the dataset, identifying patterns and trends, detecting outliers, and making informed decisions about data analysis and modeling techniques. In addition, it helps to understand the spread of data values around the mean, helping to assess the stability and consistency of the data distribution.

In Fig. 6.12, I can see the mean values of average call duration, number of calls, and total call duration. On the y-axis, I have the frequency. On the x-axis, the distribution of data is presented. This tells us how the majority of calls along with their duration are spread over time to provide a meaningful insight into the normal traffic distribution versus outliers.

The distributions exhibit a Poisson distribution pattern, which assesses the probability of an event occurring over a period of time or distance. In Poisson distribution, events are independent of each other, and there is no restriction on the timing of their occurrence.

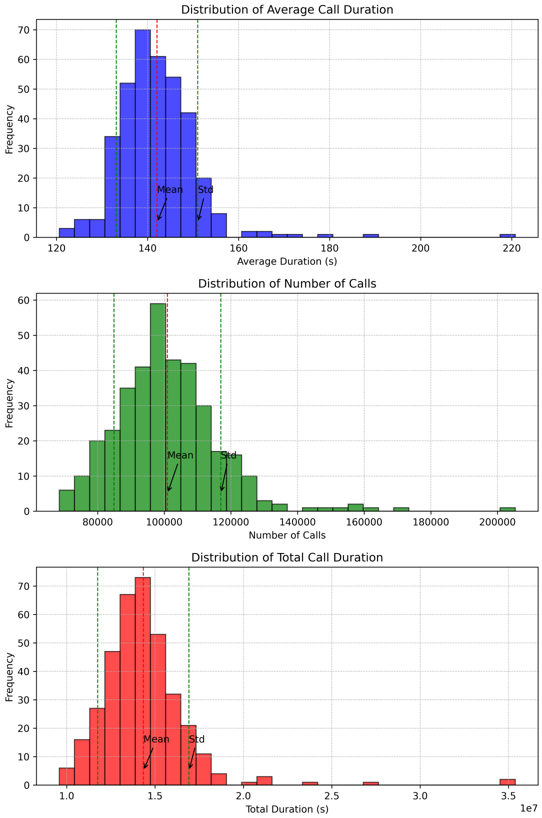


Figure 6.12: Data distribution.

I carry out the experiments with the first algorithm, which is Z-score to detect and predict anomalies in the data based on factual annotations. To add one extra information in the graphs, I use both original and deseasonalized data for accuracy comparison along with the evaluation metrics. In Fig. 6.13, the orange lines show the agreement between the detected and predicted anomalies, while the yellow lines tend to reveal more anomalies predicted that actual data annotations don't detect.

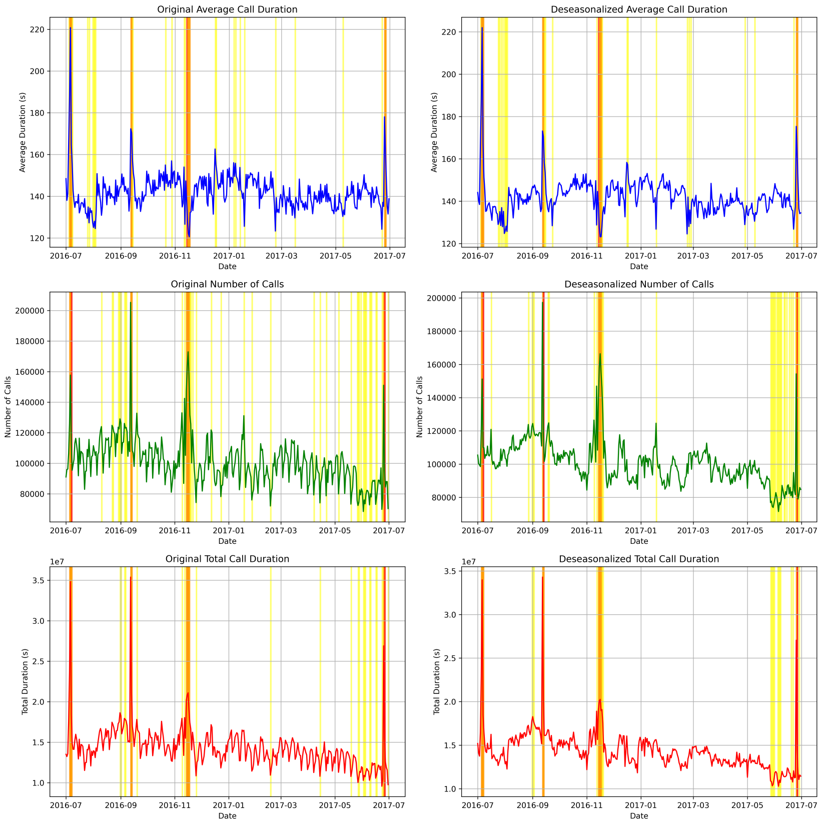


Figure 6.13: Z-score algorithm.

The performance evaluation for Z-score algorithm based on the metric scores is presented in Fig. 6.14. As mentioned earlier, the metric scores are Precision, Recall, F1-score, ROC-AUC and Accuracy. The figure manifests the evaluation of original and deseasonalized data. In the left column, the metrics show better performance and higher accuracy when the data is deseasonalized.

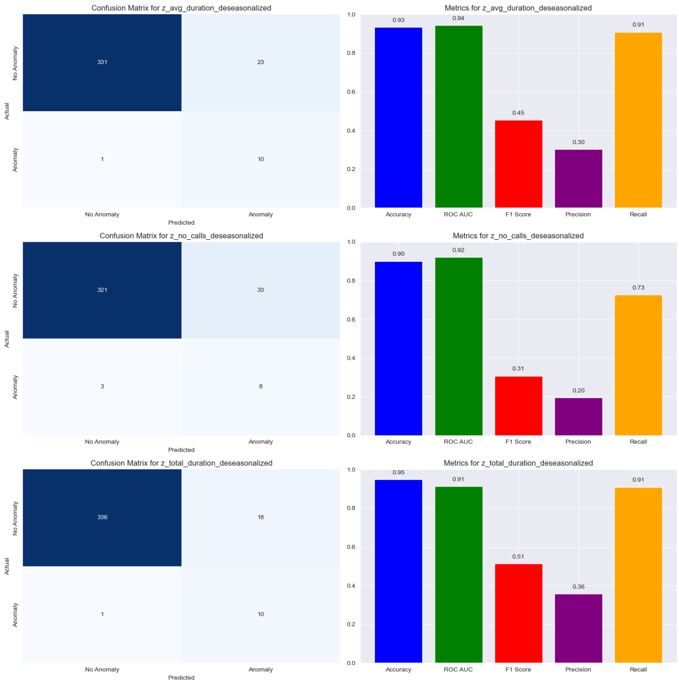


Figure 6.14: Z-score evaluation metrics.

The performance metrics show how well the algorithm is performing as well as the efficiency. However, the Precision, Recall and F1-score do not show promising results. The algorithm made promising results when it comes to high detection of anomalies but failed with low F1-score, which is the harmonic mean of Precision and Recall.

I move to the second algorithm in our study to detect and predict anomalies, which is Isolation Forest. This algorithm is specifically developed for data anomaly detection. The respectable feature of this algorithm is low memory requirement, which makes it work well with large dataset.

In Fig. 6.15, the algorithm with deseasonalized data is located on the right side and on the left side, I have the original data. The darker orange is where the actual and predicted anomalies meet, while the yellow lines show possibilities of anomalies predicted that actual annotations don't mark them.

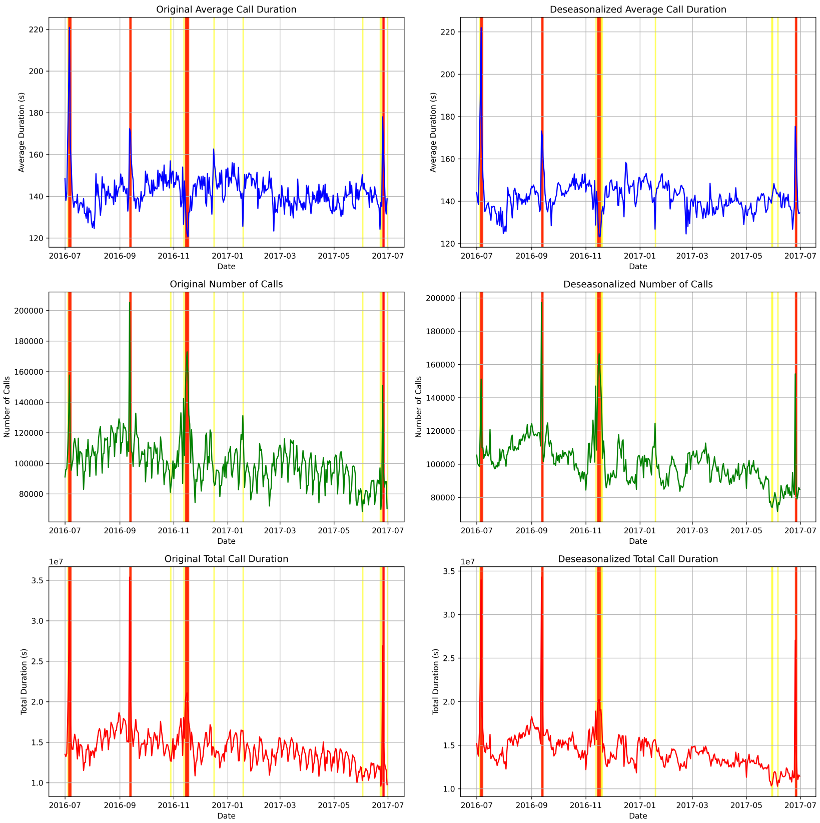


Figure 6.15: Isolation Forest algorithm.

The performance metrics of this algorithm appear to be higher than the ones of Z-score algorithm (see in Fig. 6.16). The accuracy is about 98% with the deseasonalized data, higher than when using original data. In addition, I see also all other metrics showing higher percentages in measurements.

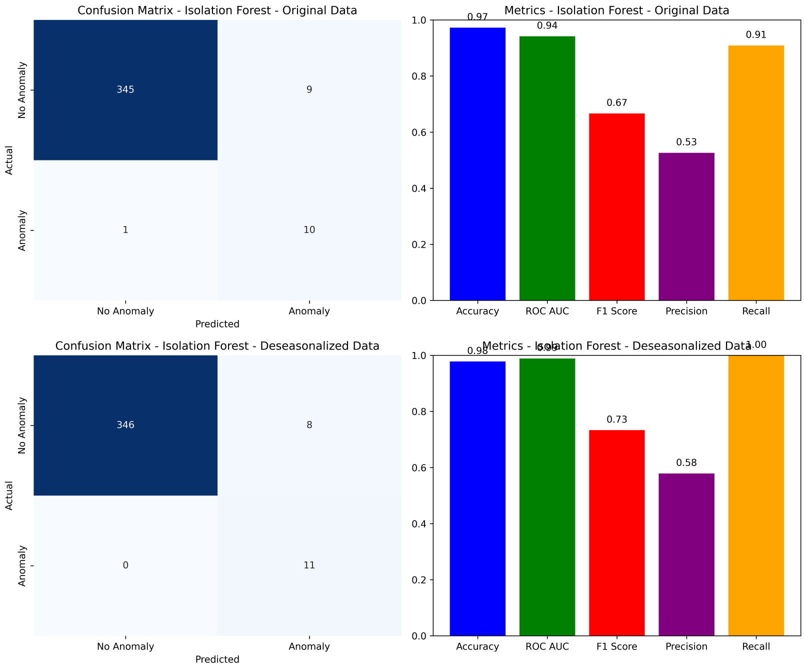


Figure 6.16: Isolation Forest evaluation metrics.

I perform experiments employing the K-means clustering algorithm for anomaly detection and prediction in the data, relying on factual annotations.

I utilize the silhouette score [109], which is a metric used to measure the goodness of a clustering technique. It quantifies how well-defined the clusters are in the data. The score ranges from -1 to 1. For each data point, the silhouette score measures how similar it is to its own cluster compared to other clusters. Higher silhouette scores indicate better-defined clusters. This metric helps in selecting the optimal number of clusters for techniques like K-means clustering and evaluating the overall quality of clustering results. The obtained result of silhouette score is about 0.54.

The number of clusters that maximizes the silhouette score is typically chosen as the optimal number of clusters. Th number of clusters were found to be 5 in my study.

I incorporate both original (on the left) and deseasonalized (on the right) data for accuracy comparison to enhance the graphs with additional insights along with performance metrics. In Figure 6.17, the grey lines show agreement between detected and predicted anomalies, while the yellow lines depict instances where more anomalies are predicted compared to those detected by the actual data annotations.

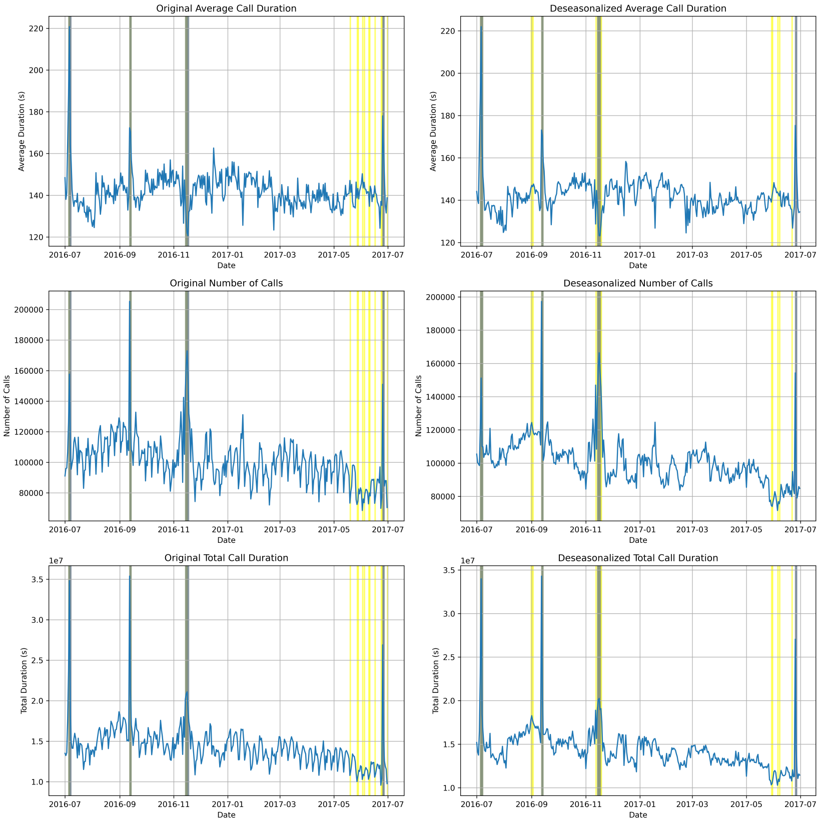


Figure 6.17: K-means clustering algorithm..

The evaluation of the K-means algorithm's performance, based on metric scores, is shown in Figure 6.18. As mentioned earlier, the metrics include Precision, Recall, F1-score, ROC-AUC, and Accuracy. The figure represents the evaluation of both original and deseasonalized data. In the left column, the metrics indicate better performance and higher accuracy when the data is deseasonalized.

While the performance metrics manifest insights into the algorithm's effectiveness and efficiency, the Precision, Recall, and F1-score do not exhibit promising results. Although the algorithm demonstrates a high detection rate for anomalies, it falls short in achieving a satisfactory F1-score, which represents the harmonic mean of Precision and Recall. However, the algorithm fits well in providing a high accuracy results in detecting the annotated anomalies along with the future prediction and possibilities of additional outliers that may not be easily apparent to the factual annotations.

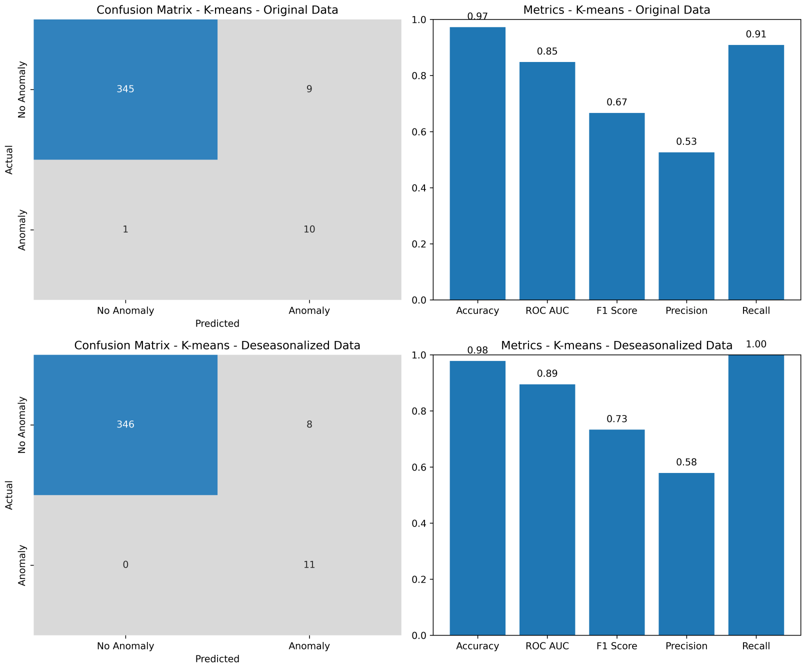


Figure 6.18: K-means clustering evaluation metrics.

The final algorithm in our study is DBSCAN, which is quite robust to anomalies and has a high notion of noise. Another advantage is that the algorithm does not need to specify the number of clusters in advance.

In Fig. 6.19, the deseasonalized data used in the algorithm on the left side, and the original data graphs are on the right side. I see similarity between Isolation Forest and DBSCAN algorithms when it comes to accuracy. Though, the DBSCAN algorithm tends to find more anomalies than Isolation Forest and with higher accuracy.

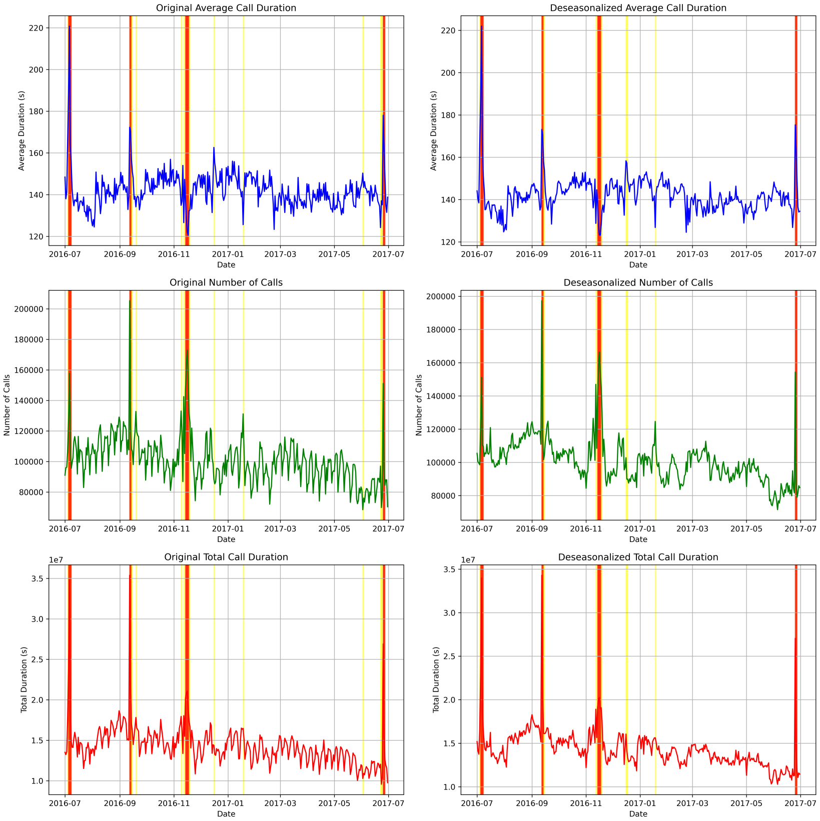


Figure 6.19: DBSCAN algorithm.

The evaluation metrics in Fig. 6.20 show high accuracy of 98% with deseasonalized data. The other metrics also indicate the efficiency and effectiveness of the DBSCAN algorithm. Yet, I notice significantly similar results with the Isolation Forest algorithm. That means both algorithms equally show promising results for anomaly detection and prediction.

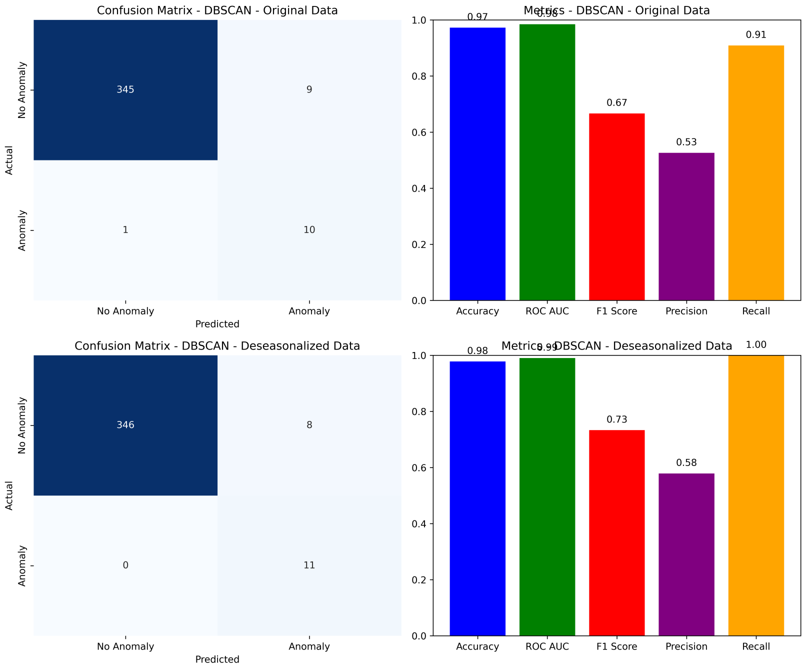


Figure 6.20: DBSCAN evaluation metrics.

6.6 Conclusion

In this chapter, research about characteristics of voice traffic profiles and patterns using CDR data is presented. The data cover 37 million CDR records of one-year data. The study consists of two phases. To understand the anomaly and outlier behavior, I have to know first how the normal behavior of voice traffic looks like.

In the first phase, I manifest the normal voice traffic profiles using linear regression function, GMM and MS algorithms to identify the call duration groups and peak hours respectively. In addition, I represent data samples of daily and monthly traffic along with the linear function between call numbers and duration. The first phase is about defining a set of boundaries to isolate a daily normal traffic behavior from outliers that the network might face throughout the year. This borderline between normal and abnormal behaviors is crucial for reference and annotations for the second phase of this study in terms of number of calls, call duration, peak hours and daily traffic profiles. Moreover, it helps us to understand and choose the right attributes for the modeling.

The second phase, I visualize the whole dataset and deseasonalize them for higher accuracy results. Then, I show the normal distribution of the available attributes. The distribution shows Poisson distribution, which determines the likelihood of an event occurring over a period of time or distance. The events are independent of each other, with no limit on the time of occurrence as stated in Poisson distribution.

Afterwards, I employ three known algorithms in detecting and predicting anomalies. Z-score (standard score), Isolation Forest, K-means and DBSCAN algorithms. Z-score is well-known in identifying outliers but it fails when it comes to crucial and extreme outlier values. On the other hand, Isolation Forest shows promising results in detection and prediction of anomalies in my study with accuracy results. However, poorly selected hyperparameters may result in lower accuracy.

In contrast, the K-means clustering algorithm exhibits a notably high performance, achieving an accuracy rate of 96% in effectively detecting and predicting underlying anomalies. Moreover, this performance is particularly enhanced when the dataset has undergone deseasonalization, a process aimed at removing seasonal patterns or variations from the data. In retun, DBSCAN shows great performance at splitting high-density clusters from low-density clusters. The DBSCAN algorithm provides higher accuracy, computational time and efficiency than other algorithms with 98%. I evaluate them based on several performance metrics. This level of accuracy holds true even when applied to datasets sourced from the latest mobile network generations, including those involving 5G Call Detail Record (CDR) data.

Chapter 7

Multi-sensory precipitation forecast

7.1 Introduction

Time series data forecasting utilizes models to fit historical data. The data are commonly a sequence measured at consecutive equally spaced points in time. These measurements are tracked, monitored, and aggregated over time. This is opposed to cross-sectional data that evaluate individuals at a single point in time. In addition, time series data may have an internal structure like autocorrelation, trend, or seasonal deviation [110].

Numerous objectives can be achieved in studying and analyzing such data since they generate mechanisms, understanding, and forecasting future events.

Time series data analysis takes place in a range of fields. For instance, it is used in agriculture for annual production. It also unveils considerable results in business and economics in terms of stock prices recorded daily or sales over a certain period of time. In meteorology that signifies for instance daily, monthly, or annual amounts of rainfall data [111]. These instances manifest the importance of unit of time in time series data analysis.

In this work, I use time series models to analyze and forecast the collected rainfall data over the region for a better understanding of the climate changes throughout the years. I implement three time series models to forecast the amount of rainfall to achieve more accurate results. The selected models are i) SARIMA model, which is a statistical analysis model used to model univariate time series data that may contain trend and seasonal components [112]. ii) a CNN model for univariate time series, a class of deep neural networks that was originally developed for image analysis and recognition. Nevertheless, CNN can be used successfully to model univariate time series forecasting. iii) Prophet model, which is an additive regression model. The model is an open source introduced by Facebook to cover several forms of seasonal/univariate data and can be implemented as an additive time series forecasting model.

In Fig. 7.1, I exhibit the process of rainfall time series forecasting in our study. A common assumption that is used in time series techniques is stationarity. Stationarity is a process when the mean, variance, and covariance do not change over time. In addition, I go through the time series components, which can be decomposed into trend, seasonality, and irregularity in the data. The trend is the time series long-term pattern whether it is increasing or decreasing. The seasonality is an event when the time series shows regular variations over the same period of time. In contrast, irregularity is about unpredicted events which makes it a random variable.

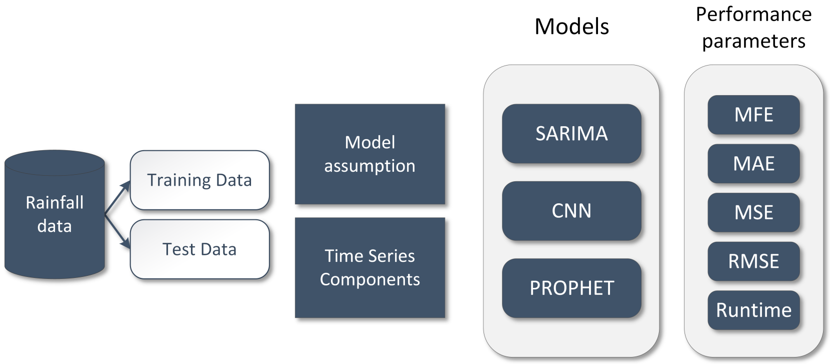


Figure 7.1: Rainfall forecasting process.

The models are tested and evaluated based on several performance parameters. These parameters are used for accuracy measurement. The performance parameters that are used are Forecast Error, Mean Forecast Error, Mean Absolute Error, Mean Square Error, Root Mean Square Error, and runtime. The error implies here the unpredictable part of an observation.

7.2 Theory of the models

In this section, I discuss the theory behind the models and their methods of implementation. Furthermore, the requirements and components of each model are presented. I also show the formulation of these components along with their procedure on how they are composed.

7.2.1 SARIMA model

Seasonal Autoregressive Integrated Moving Average (SARIMA) is a statistical analysis model and direct modeling of seasonal components as it is an extension of the ARIMA model. It is a time series forecasting technique that deals with univariate data including trends and seasonality. The reason for using SARIMA rather than ARIMA is that ARIMA does not model seasonal data.

There are requirements in SARIMA configuration for both trend and seasonal components [113]:

$$\text{SARIMA}(p, d, q)(P, D, Q)_s \tag{7.1}$$

Three trend components are involved as they are similar to the ARIMA model, and four seasonal components are involved in SARIMA.

p and seasonal P : autoregression order, or lag order d and seasonal D : difference order/degree

q and seasonal Q : moving average order

s : time steps for a one seasonal period

However, the SARIMA model is composed of several models combined to build SARIMA [114]:

Auto-Regressive (AR) model: The model works as the previous values influence future values. p is a parameter used to consider the number of lagged observations.

$$Y_t = \alpha + \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + \cdots + \beta_p Y_{t-p} + \epsilon_1 \tag{7.2}$$

Moving Average (MA) model: It is utilized to define the stationarity of a time series. It is the result of dependency between observed values and residual error applied on lagged observations. It is usually indicated as q and stated as:

$$Y_t = \alpha + \epsilon_t + \phi_1 \epsilon_{t-1} + \phi_2 \epsilon_{t-2} + \cdots + \phi_q \epsilon_{t-q} \tag{7.3}$$

Auto-Regressive Moving Average (ARMA) model: The combination of AR and MA models. Forecasting of a time series is based on the impact of residuals and previous lags.

$$Y_t = c + \epsilon_t + \sum_{i=1}^{p} \phi_i Y_{t-i} + \sum_{i=1}^{q} \theta_i \epsilon_{t-i} \tag{7.4}$$

Auto-Regressive Integrated Moving Average (ARIMA): This model describes the autocorrelations in the data. It is a combination of several differences already applied to the model to have it stationary, the number of previous lags along with residuals errors to forecast future values.

$$Y_t = \alpha + \beta_1 Y_{t-1} + \cdots + \beta_p Y_{t-p} + \epsilon_t + \phi_1 \epsilon_{t-1} + \cdots + \phi_q \epsilon_{t-q} \tag{7.5}$$

7.2.2 CNN model

CNN model was developed for 2-D image data, but it can also be used to model univariate time series data. CNN is a class of deep learning and has a convolutional hidden layer, which runs over a one-dimensional sequence. The convolutional and pooling layers are followed by a dense fully connected layer that interprets the features extracted using the convolutional section of the model [115].

The use of the model is done as following steps [116]:

1. The spatial and temporal dependencies are captured by the neural network in the input feature map by utilizing a convolutional kernel on this input tensor.

2. Each element within the input feature map is parsed iteratively by the layer through a sliding convolutional kernel.

3. A convolved output feature map at this phase is produced that models the translational invariance nature of the input feature map.

4. The convolved feature map is down-sampled by the pooling layer. Tensor operations along with a sliding window are applied for this process.

Moreover, the pooling is either maximum or average. The pooling layer is used to reduce the number of parameters and therefore, reduces the overfitting risk.

5. Epoch in neural networks is defined as one cycle through a full training dataset. Every training of a neural network usually takes several epochs. In other words, I feed a network the training data for more epochs so that the network is given a chance to see the data history and readjust the model parameters. The model then is not biased toward a few data points, especially in large training sets.

7.2.3 Prophet model

It is an open-source tool, introduced by Facebook and used for time series data forecasting. The Prophet model is based on an additive model where non-linear trends fit with seasonality including holidays. The advantage of using the Prophet model is the robustness of missing data as well as the outliers.

The Facebook model is founded on the curve fitting technique that belongs to the Bayesian model. The technique is a perfect fit when there is a strong seasonality attribute in the time series data as an influencing factor. The model deals with four major components as a procedure that is based on an additive regression model [117]:

1. A piecewise linear, which can be interpreted as a logistic growth curve trend. In general, the model finds the changes in trends automatically, selecting changepoints from the data.

2. Fourier series is being used to model yearly seasonal components.

3. Dummy variables for weekly seasonal components.

4. Holidays that are provided which I do not consider for our study.

The model function is formed as follows:

$$f(t) = g(t) + s(t) + h(t) + e(t) \tag{7.6}$$

Where: $f(t)$ is the forecast function based on the additive regression model. $g(t)$ represents the trend, and it models non-periodic changes. $s(t)$

stands for seasonality, and it describes the periodic changes. $h(t)$ is used to show the effects of holidays. $e(t)$ refers to the error term that reports odd changes not adjusted by the model.

7.3 Performance Evaluation

The performance of each model can be tested and evaluated using a set of different forecast accuracy measurement parameters including the executing time of each model [118]:

7.3.1 Forecast Error (FE)

It is defined as the difference between the actual value and forecast value of a time series. It is usually calculated as below:

$$e = y - \hat{y} \tag{7.7}$$

Where y is the observation and $\hat{y}$ denotes the forecast value from all previously observed values.

7.3.2 Mean Forecast Error (MFE)

It is a measure of forecasting accuracy. Forecast error is the difference between actual and forecast values for a given time. A value other than zero tends to over-forecast (negative error) or under-forecast (positive error).

$$MFE = \frac{\sum_{i=1}^{n} y_i - \hat{y}_i}{n} = \frac{\sum_{i=1}^{n} e_i}{n} \tag{7.8}$$

7.3.3 Mean Absolute Error (MAE)

Absolute error is the difference between measured values and true values. On the other hand, the mean absolute error is the average of all absolute errors. A mean absolute error of zero shows no error. Below is the MAE formula:

$$MAE = \frac{\sum_{i=1}^{n} |y_i - \hat{y}_i|}{n} = \frac{\sum_{i=1}^{n} |e_i|}{n} \tag{7.9}$$

7.3.4 Mean Square Error (MSE)

It is a mean square deviation. MSE is the average of squared forecast error values. When the forecast error values are squared, this will force the values to be positive. A zero value indicates the best skill or no error, as shown in the below equation:

$$MSE = \frac{\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2}{n} = \frac{\sum_{i=1}^{n} e_i^2}{n} \tag{7.10}$$

7.3.5 Root Mean Square Error (RMSE)

It is the square root of the mean square error and the values are in the same units of prediction. RMSE is the difference between forecast values. A zero RMSE shows no error. The formula can be represented as follows:

$$RMSE = \sqrt{MSE} = \sqrt{\frac{\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2}{n}} \tag{7.11}$$

7.3.6 Runtime

It is executing time for a script to run on an operating system. The lower the value is, the faster the model implementation. The script is processed, read, and then compiled into bytecodes. Finally, the bytecodes are executed to run the program.

7.4 Results and discussion

In this section, I discuss the results of implementing the models. I show the forecast along with the actual precipitation through the graph to explain the behavior of the models. Furthermore, I present performance parameters for each model to demonstrate the accuracy measurement values.

7.4.1 SARIMA model

In this section, I discuss the methodology of forecasting with SARIMA. Before modeling, there are a few assumptions that need to be considered.

Stationarity

Stationarity states that the statistical properties and parameters of a stochastic process are constant through time [119]. Stationary is an important assumption in time series analysis to enable forecasting since it has a predictable probability distribution. Thus, the mean, variance, and covariance should have time dependency characteristics.

As a first step to ensure that our data is time series stationary, I perform a test to determine if the process has a unit root. This test is called the Dickey-Fuller test [120]. The results from the Dickey-Fuller test are shown in Fig. 7.2.

```
Dickey-Fuller test results:

ADF = -5.16878383817796
p-value = 1.0178254081600643e-05
  1%: -3.479
  5%: -2.883
  10%: -2.578

p-value is less than or equal to 0.05
reject the null hypothesis because the data does not have a unit root and is stationary.
```

Figure 7.2: Dickey-Fuller test results.

A unit root presence in the time series data indicates the non-stationarity of the data and it is represented as a null hypothesis. According to the test on our data, the null hypothesis is rejected (p-value < 0.05), which means the data do not have a unit root. Therefore, the data are stationary.

Autocorrelation Function

ACF is a measure of a linear relationship among variables, and it calculates the correlation between lagged values of a time series [121]. From a mathematical point of view, I apply ACF to find the right q parameter from the MA of the model. From our analysis of the data, I select $q = 3$ for the MA parameter as shown in Fig. 7.3.

Partial Autocorrelation Function

PACF is the correlation between observations at two given time points, which I consider both observations to be correlated to observations at other time

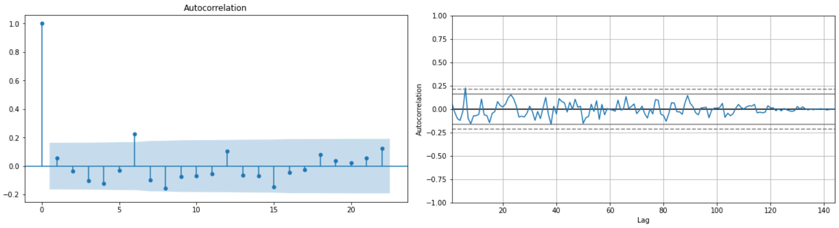


Figure 7.3: Autocorrelation Function.

points [121]. According to PACF, I can see that the parameter corresponding to the Autoregressive (AR) part is $p = 5$ as shown in Fig. 7.4.

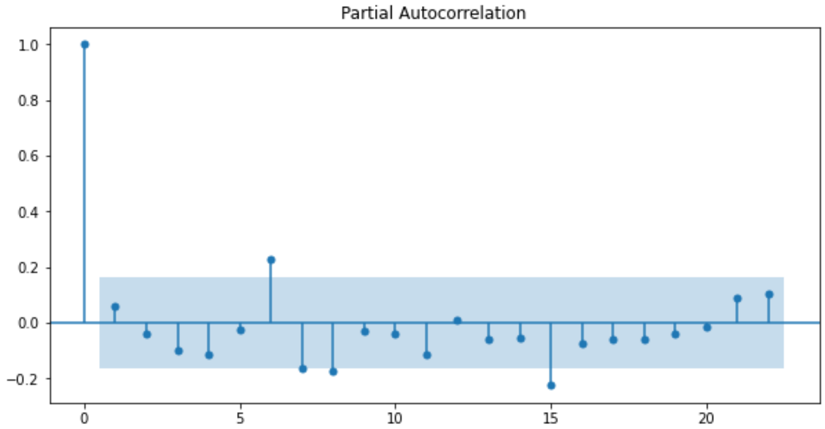


Figure 7.4: Partial Autocorrelation Function.

SARIMA Function

In this section, I try different values for the Integrated (I) part after choosing the AR and MA parameters. The values usually are $d = 0$ or $d = 1$. Based on testing all possible values, I find out the value that fits better is d = 1.

Once the values of AR, MA, and I are found, I need to find the seasonal values, which can be found after several iterations. As to the available data, the seasonal length is to be $s = 12$ due to the annual period (12 months).

$$\text{SARIMA}(5, 1, 3)(0, 1, 1)_{12} \tag{7.12}$$

On the other hand, I try several iterations, and I find the parameters with the best Akaike Information Criterion (AIC) as the following:

The Akaike information criterion is an estimator of prediction error and thereby the relative quality of statistical models for a given set of data.

As the appropriate parameters were found, I can state the function in our Python script through the SARIMA library. I can plot the output of the model including the next 12 months' prediction in Fig. 7.5.

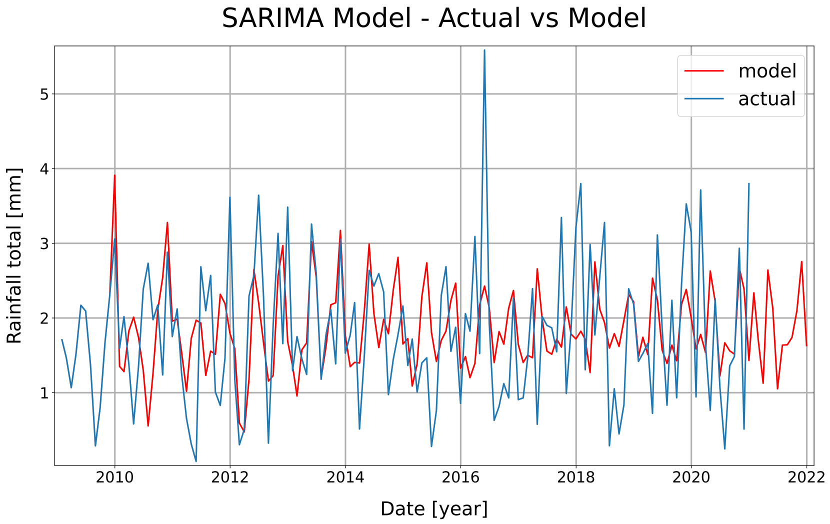


Figure 7.5: SARIMA model.

I have on the y-axis the amount of precipitation in [mm]. On the x-axis, the unit of time is exhibited in [year] from 2009. The blue line represents the actual amount of rainfall, and the red line shows the prediction. Once the model is defined, I evaluate the model using forecast evaluation parameters as indicated in Table 7.1 below:

Table 7.1: SARIMA Forecast Evaluation

Model	runtime (s)	MFE	MAE	MSE	RMSE
SARIMA	7,962	$\mathbf{0,003}$	$\mathbf{0,721}$	$\mathbf{0,930}$	$\mathbf{0,965}$

The runtime of the script takes $7,962$ s to forecast the rainfall data. Based on the recorded parameters, the SARIMA model does not perform well as they are supposed to be close to zero to achieve the best performance.

In addition to presenting and fitting the model, I can check using residuals diagnostic tools (shown in Fig. 7.6) whether the model has adequately captured the information in the data. In a time series model, residuals are left over after fitting the model and it is characterized as the difference between the observations and the corresponding fitted values. I can divide the residual diagnostic [122] findings into 4 parts with their correspondents: i) Standardized residual or residual errors fluctuate around zero mean and with a uniform variance. ii) Histogram, density plot suggests normal distribution with zero mean. iii) Normal Q-Q, which tells that all the dots fall roughly in line with the red line. iv) Correlogram (ACF), the plot shows the residual errors are not autocorrelated. Any autocorrelation would imply that there are some patterns in the residual errors which are not explained in the model.

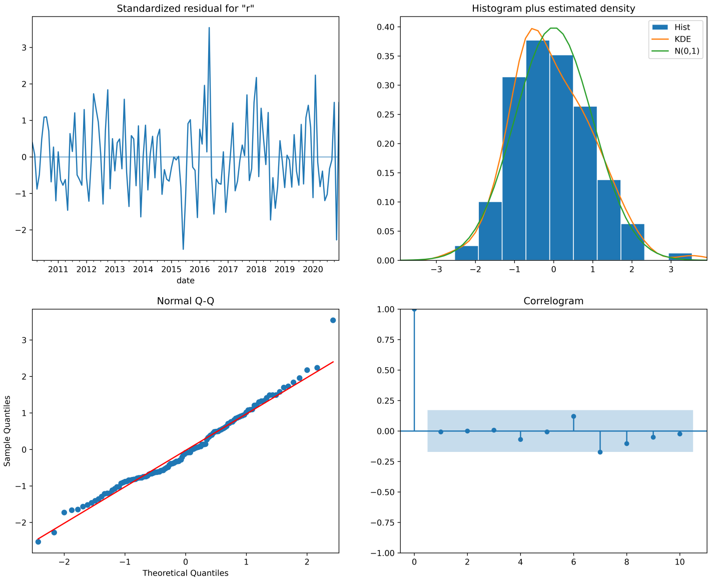


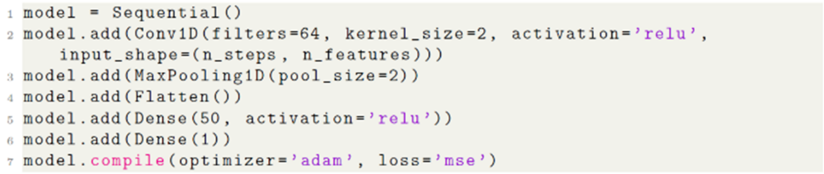
Figure 7.6: Residual diagnostics.

7.4.2 CNN model

In this section, I work on the CNN model for the univariate dataset. The model has a convolutional hidden layer that works over a 1-D sequence. The convolutional and pooling layers are followed by a dense fully connected layer that interprets the features extracted by the convolutional part of the model.

I almost always have multiple samples; therefore, I have to re-shape the input component of the training data to fit with the requests from the model: samples, timesteps, and features.

I can define the CNN model with the correct corresponding input shape. The key in the definition is the shape of the input, which is specified in the input shape argument on the definition of the first hidden layer. Now that I have the correct input shape, I can define the CNN model as follows (Fig. 7.7):



```
model = Sequential()
model.add(Conv1D(filters=64, kernel_size=2, activation='relu',
    input_shape=(n_steps, n_features)))
model.add(MaxPooling1D(pool_size=2))
model.add(Flatten())
model.add(Dense(50, activation='relu'))
model.add(Dense(1))
model.compile(optimizer='adam', loss='mse')
```

Figure 7.7: Defining the CNN model.

The model must meet the expectation of input for every sample with steps = 12, and in terms of the number of features = 1. The proposed CNN model consists of 5 layers.

The output model forecast is plotted in Fig. 7.8. I present on the y-axis the precipitation in [mm] and on the x-axis the time in [$year$]. The year represents the measurements since 2009 and the leading forecast of 12 months.

The model is tested with two epochs, the first experiment is with 100 epochs and then with 1000 epochs. They are both evaluated with the selected forecast evaluation parameters as shown in Table 7.2 for forecast evaluation per 100 and 1000 epochs respectively.

I can see that 100 epochs performs less accurately than 1000 epochs. The only downside is that running the training set 1000 times (epochs) takes longer than 100 epochs. The parameters for 1000 epochs show that it is very likely with higher performance forecasting compared to the SARIMA model and CNN model with 100 epochs.

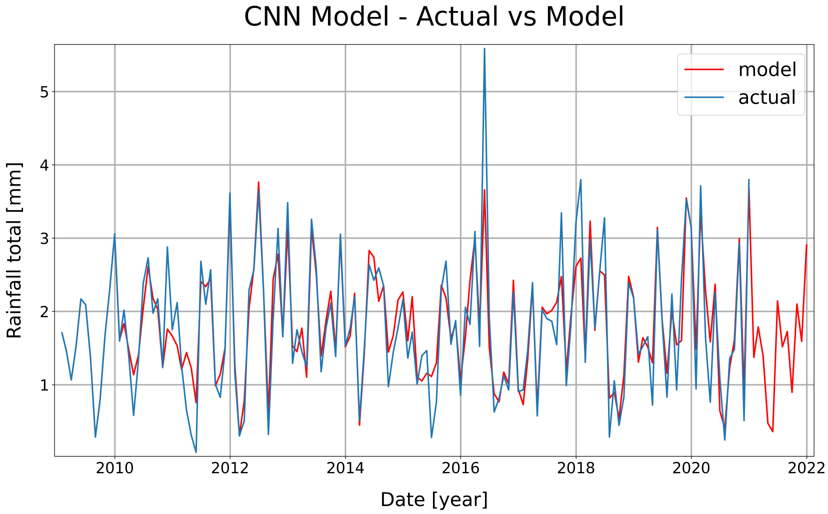


Figure 7.8: CNN model with 1000 epoch.

Table 7.2: CNN Forecast Evaluation

Model	runtime (s)	MFE	MAE	MSE	RMSE
CNN 100epoch	5,759	$-0,083$	0,646	0,710	0,843
CNN 1000 epoch	12,246	0,008	0,246	0,128	0,357

7.4.3 Prophet model

The Prophet model [123] requires a dataset with a date and time column and a sample values column, which is the precipitation. Once I have the input ready, I can define the model components.

In the Prophet model, I fit the trend component flexibly which allows us to model the seasonality more accurately and the result is a more accurate forecast (see Fig. 7.9). On the left side, I present in the three figures the trend, yearly and monthly respectively. In the trend graph, the y-axis displays the precipitation, and the x-axis the trend on a yearly basis for our entire data. By default, Prophet provides uncertainty intervals for the trend component by simulating the future trend changes to our time series.

The trend graph shows an increment in the precipitation in 2013 and rapidly decreasing in 2015. The precipitation is increasing gradually in 2017 and 2018 respectively, then diminishing slowly with a sign of stabilizing trend.

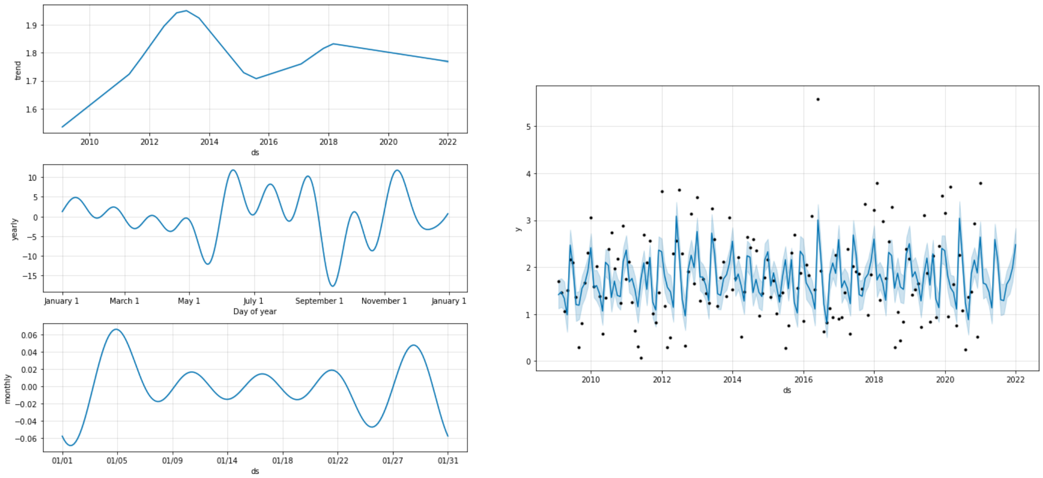


Figure 7.9: Prophet model components.

I find negative values in yearly and monthly components due to variations in the results. The yearly component varies between the mid and the end of the year. In contrast, the monthly component varies between the month's beginning and end.

On the right side of Fig. 11, the model is presented. On the y-axis, (y) represents the precipitation, and (ds) on the x-axis represents the years of forecasting. The dots are the historical data, while the deep blue line is the rainfall forecasting model. The light blue shadow is a 95% confidence interval around the forecasting (deep blue line).

In Fig. 7.10, I represent the Prophet forecast model against the actual data including the next 12 months' prediction. On the y-axis, I have the precipitation in [mm], and on the x axis, the time is presented on a yearly basis. The prediction with the Prophet model indicates good accuracy based on the forecast evaluation shown in Table 7.3.

Table 7.3: Prophet Forecast Evaluation

Model	runtime (s)	MFE	MAE	MSE	RMSE
Prophet	$\mathbf{1,992}$	$\mathbf{0,001}$	$\mathbf{0,640}$	$\mathbf{0,644}$	$\mathbf{0,803}$

The runtime spent executing the model was better compared to CNN and SARIMA models. On the other hand, I have almost perfect MFE compared

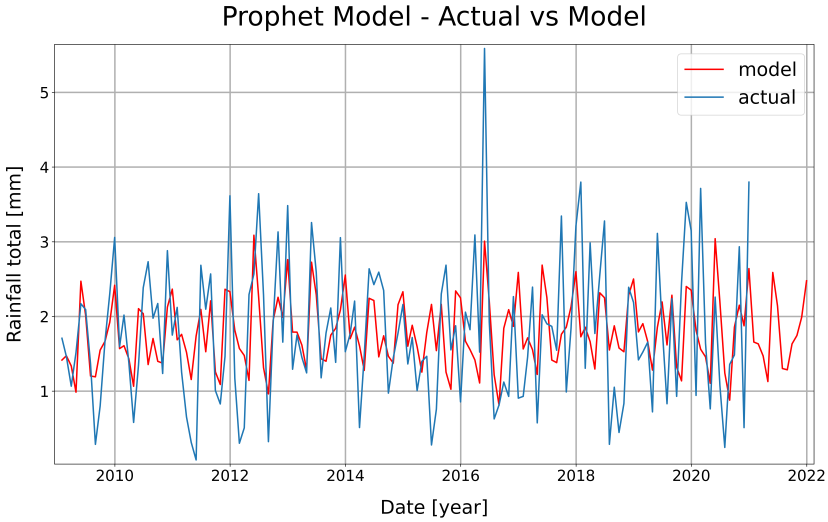


Figure 7.10: Prophet model.

to the other models but with higher MAE and MSE than the CNN and SARIMA.

7.5 Conclusion

The objective of this study is to implement three different models for rainfall time series forecasting. There are two major aims this study achieved, I manifest the importance of rain gauge distribution and numeral reduction over certain areas, and I conclude that SARIMA is not an ideal time series forecasting with univariate data compared to neural network models. The reduction of the number of rain gauges to 67% still provides high accuracy forecasting results as shown in Fig. 7.11.

The forecasting is built on the classic statistical analysis model SARIMA, a neural network based on the CNN model, and the additive regression Prophet model. SARIMA is a direct modeling of seasonal components, dealing with univariate time series data. On the other hand, CNN is a feedforward neural network model and has a convolutional hidden layer running on one-dimensional sequences of time series data. Prophet is Facebook's open-source tool for time series forecasting. It is a procedure based on an additive model as the non-linear trends are fit on data with seasonality components.

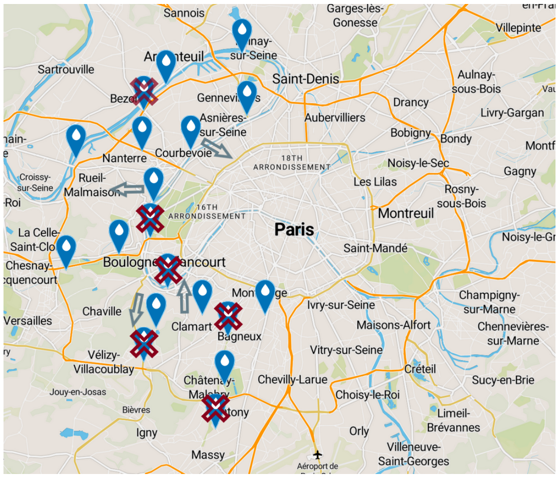


Figure 7.11: Rain gauges reduction and redistribution.

The results show that SARIMA is not a good option for rainfall forecasting with an error ratio slightly higher than the other two models. In addition, it needs an intensive analysis of the dataset to find its parameters, thus, a longer execution time is required. In contrast, the CNN model delivers lower Mean Absolute Error and Mean Square Error than the other two models. Therefore, it provides more accurate results for forecasting with low errors. Finally, the Prophet model shows significant results in terms of Mean Forward Error, and a shorter time is needed to be executed. However, it shows a higher Mean Absolute Error and Mean Square Error than the CNN model.

Based on our study, I see the importance of reallocating the rain gauges and/or using fewer number of rain gauges since I see agreement in rainfall amount recorded from all rain gauges distributed over the area of study.

Chapter 8

Conclusion

This chapter concludes the studied cases and summarizes the thesis through providing the works that have been done on the available datasets, offering new approaches and understanding. Finally, it is followed by the research contribution based on the presented work.

8.1 Thesis summary

This thesis aims to propose machine learning models for pattern identification and predictive analytics delivering several case studies. Each study is based on different datasets and novel approaches.

The first chapter introduced the entire thesis, thesis objectives and organization. Chapter two discussed the state of the art and present works related to our studies. The work that I have done throughout the study was confirmed to be up to date and even novel according to the current proposed studies in nowadays research fields.

In chapter three, I explained in detail the datasets being used for our studies. Some of the studies have involved using different datasets with various forms and structures, each needed to be preprocessed and cleansed for use according to the models' requirements.

Various attributes were discussed and even eliminated from the data since there was no use in our research. In addition, several hypotheses were proposed and proven through data analytics. I did a novel analysis on CDR data for both local and international voice traffic for various scenarios including daily, weekly, monthly and yearly data based. Furthermore, I have proved how weekdays and weekends, even holidays, influence traffic patterns. On the other hand, the influence of the neighboring countries and the distance to the reference network in terms of countries affecting the overall volume of traffic when it comes to international voice calls.

The study was based on four different datasets and merging all four sources of data was challenging. Chapter four is purely based on mathematical approach of queueing theory using both signaling and voice traffic data. I researched and discussed theoretically the current mobile network traffic patterns following a hundred-year-old queuing theory and probability distribution. The proposal has mathematically and practically proven that each customer is independent of the number of arrivals. The waiting time follows the Poisson distribution and varies based on the system. Service and interarrival time are exponentially distributed.

In chapter five, I studied the normal patterns in voice traffic and proposing different models to determine and predict anomalies. I described the normal behavior of voice traffic via various attributes using certain models to target specific attributes in the data. Then, I used three models for anomaly prediction. Unlike other data sources, a forecast study and time series data have been proposed in chapter six for rainfall forecasting based on many sensors in a particular region. The research discussed the potential elimination and relocation of several sensors and employing the correct models providing similar accuracy and efficiency.

The study indicates using 67 percent of the current used sensors to be able to offer similar results. However, I can also recommend redistribution of the sensors based on their coordinates, and that is one of my future objectives.

In general, there were of course challenges along the way struggling with finding a reliable sources of data as well as fulfilling the requirements of our studies. Another challenge was the ability to provide high accuracy and efficiency in results to be competitive in the related fields since there are hundreds of researches published on a daily basis.

8.2 Contribution of thesis

This thesis is based on several studies combining data analysis and modeling. Each study serves in specific fields including different understanding, performance improvement, cost reduction, accuracy in detection and prediction anomaly.

There are novel machine learning models used for the first time in specific fields and comparing the outcomes using several performance metrics to evaluate the accuracy and effectiveness of each model. I deliver a detailed analysis of CDR data for many different scenarios, including the distance influence on the international voice traffic as well as the traffic distribution. In addition, a novel approach is proposed to study the voice traffic that provides different views and understanding.

I split the study into two phases, a novel approach by studying the normal voice traffic patterns and trends using three different models targeting certain attributes in the data for more accurate outcomes, then using three known models to detect and predict anomalies.

Furthermore, I theoretically constructed the applied queueing theory using signaling and user data in mobile networks. This is to validate that the obtained results still follow the expected Poisson and Exponential distributions.

Moreover, I provide a novel study of multi-sensory precipitation forecast is presented using time series models. The study suggests that the reduction of rain gauge numbers to 67 percent of currently distributed over the area with reallocating, would provide similar accuracy in forecasting results. In general, the sensor distribution phase is done at an early stage of the study, that needs to be reconsidered over a long period of time.

Many times, reduction and relocation are needed after deep analysis of the area of study and comparing the results of both cases using the right models and performance parameters.

8.3 Future Works

In the future, this work can be put into practice in live systems since all the data are real and generated from live systems. It can also be extended to cover more algorithms and models. Another scenario is to test under diverse environments such as other locations where the data were generated from and evaluate the results to see if the models work the same way and provide similar results and high accuracy. This way I can understand whether they are only applicable for specific scenarios or environments. Alternatively, to see how the performance parameters perform under certain conditions.

Publications of Author

Aziz, Z., Bestak, R. (2018). ”Analysis of Call Detail Records of International Voice Traffic in Mobile Networks,” 2018 Tenth International Conference on Ubiquitous and Future Networks (ICUFN), Prague, Czech Republic, 2018, pp. 475-480, doi: 10.1109/ICUFN.2018.8436669.

Aziz, Z., Bestak, R. (2019). Mobile Voice Traffic Load Characteristics. In: Gaj, P., Sawicki, M., Kwiecień, A. (eds) Computer Networks. CN 2019. Communications in Computer and Information Science, vol 1039. Springer, Cham. https://doi.org/10.1007/978-3-030-21952-9_15

Aziz, Z., Bestak, R. (2020). Dependency Between the Distance and International Voice Traffic. In: Habachi, O., Meghdadi, V., Sabir, E., Cances, JP. (eds) Ubiquitous Networking. UNet 2019. Lecture Notes in Computer Science(), vol 12293. Springer, Cham. https://doi.org/10.1007/978-3-030-58008-7_16

Aziz, Z., Bestak, R. (2024). Insight into Anomaly Detection and Prediction and Mobile Network Security Enhancement leveraging K-means Clustering on Call Detail Records. Accepted in Sensors Journal, 2024.

Aziz, Z., Bestak, R. ”Modeling Voice Traffic Patterns for Anomaly Detection and Prediction in Cellular Networks based on CDR Data”. Submitted to IEEE Transactions on Mobile Computing.

Aziz, Z., Bestak, R. ”A multi-sensory precipitation forecast study using statistical analysis, neural network, and additive regression models”. Submitted to IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing.

Bibliography

[1] M. A. Habibi, M. Nasimi, B. Han and H. D. Schotten, "A Comprehensive Survey of RAN Architectures Toward 5G Mobile Communication System", in IEEE Access, vol. 7, pp. 70371-70421, 2019, doi: 10.1109/ACCESS.2019.2919657.

[2] A. Basta, A. Blenk, K. Hoffmann, H. J. Morper, M. Hoffmann and W. Kellerer, "Towards a Cost Optimal Design for a 5G Mobile Core Network Based on SDN and NFV," in IEEE Transactions on Network and Service Management, vol. 14, no. 4, pp. 1061-1075, Dec. 2017, doi: 10.1109/TNSM.2017.2732505.

[3] D. Rupprecht, A. Dabrowski, T. Holz, E. Weippl and C. Pöpper, "On Security Research Towards Future Mobile Network Generations," in IEEE Communications Surveys and Tutorials, vol. 20, no. 3, pp. 2518-2542, thirdquarter 2018, doi: 10.1109/COMST.2018.2820728.

[4] D. Kreuzberger, N. Kühl and S. Hirschl, "Machine Learning Operations (MLOps): Overview, Definition, and Architecture," in IEEE Access, vol. 11, pp. 31866-31879, 2023, doi: 10.1109/ACCESS.2023.3262138.

[5] L. Xu, C. Jiang, J. Wang, J. Yuan and Y. Ren, "Information Security in Big Data: Privacy and Data Mining," in IEEE Access, vol. 2, pp. 1149-1176, 2014, doi: 10.1109/ACCESS.2014.2362522.

[6] D. Keysers, W. Macherey, H. Ney and J. Dahmen, "Adaptation in statistical pattern recognition using tangent vectors," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 26, no. 2, pp. 269-274, Feb. 2004, doi: 10.1109/TPAMI.2004.1262198.

[7] X. -Y. Zhang, C. -L. Liu and C. Y. Suen, ”Towards Robust Pattern Recognition: A Review,” in Proceedings of the IEEE, vol. 108, no. 6, pp. 894-922, June 2020, doi: 10.1109/JPROC.2020.2989782.

[8] A. Moubayed, M. Injadat, A. B. Nassif, H. Lutfiyya and A. Shami, ”E-Learning: Challenges and Research Opportunities Using Machine Learning and Data Analytics,” in IEEE Access, vol. 6, pp. 39117-39138, 2018, doi: 10.1109/ACCESS.2018.2851790.

[9] Ying-Huey Fua, M. O. Ward and E. A. Rundensteiner, ”Structure-based brushes: a mechanism for navigating hierarchically organized data and information spaces,” in IEEE Transactions on Visualization and Computer Graphics, vol. 6, no. 2, pp. 150-159, April-June 2000, doi: 10.1109/2945.856996.

[10] K. Adnan, R. Akbar and K. S. Wang, ”Development of Usability Enhancement Model for Unstructured Big Data Using SLR,” in IEEE Access, vol. 9, pp. 87391-87409, 2021, doi: 10.1109/ACCESS.2021.3089100.

[11] L. C. Monticone, R. E. Snow and F. Box, ”Minimizing Great-Circle Distance Ratios of Undesired and Desired Signal Paths on a Spherical Earth,” in IEEE Transactions on Vehicular Technology, vol. 58, no. 9, pp. 4868-4877, Nov. 2009, doi: 10.1109/TVT.2009.2025281.

[12] E. Winarno, W. Hadikurniawati and R. N. Rosso, ”Location based service for presence system using haversine method,” 2017 International Conference on Innovative and Creative Information Technology (ICITech), Salatiga, Indonesia, 2017, pp. 1-4, doi: 10.1109/INNOCIT.2017.8319153.

[13] Budynas, R.G. and Nisbett, J.K. (2011) Shigley’s Mechanical Engineering Design. 9th Edition, McGraw-Hills, New York, p96. ISBN: 978–0–07–352928–8.

[14] Peter Chiu, Jussi Reunanen, Riku Luostari Harri Holma (June 2017). Big Data Analytics for 4.9G and 5G Mobile Network Optimization. Vehicular Technology Conference (VTC Spring), 2017 IEEE 85th.

[15] Artjom Lind, Amnir Hadachi, Oleg Batrashev (2017). A new approach for mobile positioning using the CDR data of cellular networks. IEEE,

2017 5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS).

[16] Dr. Manish Kumar, Dr. M. Hanumanthappa (2017). Crime Investigation and Criminal Network Analysis Using Archive Call Detail Records. Advanced Computing (ICoAC), IEEE.

[17] Anish Kurien, Barend Jacobus Van Wyk, Yskander Hamam (2008). Mining Time Series Data in Mobile Cellular Networks. IEEE, Third International Conference on Broadband Communications, Information Technology Biomedical Application.

[18] Sara B. Elagib, Aisha-Hassan A. Hashim, R. F. Olanrewaju (2015). CDR analysis using Big Data technology. IEEE, International Conference on Computing, Control, Networking, Electronics and Embedded Systems Engineering.

[19] Sihai Zhang, Dandan Yin, Yanqin Zhang, Wuyang Zhou (2015). Computing on Base Station Behavior Using Erlang Measurement and Call Detail Record. IEEE Transactions on Emerging Topics in Computing Year: 2015, Volume: 3, Issue: 3, Pages: 444 – 453.

[20] Diala Naboulsi, Razvan Stanica, Marco Fiore (2014). Classifying Call Profiles in Large-scale Mobile Traffic datasets. IEEE Conference on Computer Communications (IEEE INFOCOM).

[21] Ying He, Fei Richard Yu, Nan Zhao, Hongxi Yin, Haipeng Yao, Robert C. Qiu (2016). Big Data Analytics in Mobile Cellular Networks. IEEE, pp. 1985-1996.

[22] Ali, A., R.: Real-time big data warehousing and analysis framework. In: IEEE 3rd International Conference on Big Data Analysis (ICBDA), Shanghai, China. DOI: 10.1109/ICBDA.2018.8367649 (2018).

[23] Joshi, A., Oberoi, M., Bose, R.: Analyzing CDR/IPDR Data to Find People Network from Encrypted Messaging Services. In: IEEE 4th International Conference on Collaboration and Internet Computing (CIC), Philadelphia, PA, USA. DOI: 10.1109/CIC.2018.00013 (2018).

[24] Werayawarangura, N., Pungchaichan, Th., Vateekul, P.: Social network analysis of calling data records for identifying influencers and communities. In: 13th International Joint Conference on Computer Science and Software Engineering (JCSSE), Khon Kaen, Thailand. DOI: 10.1109/JCSSE.2016.7748864 (2016).

[25] Sun, W., Miao, D., Qin, X., Wei, G.: Characterizing User Mobility from the View of 4G Cellular Network. In: 17th IEEE International Conference on Mobile Data Management (MDM), Porto, Portugal. DOI: 10.1109/MDM.2016.19 (2016).

[26] Wang, X., Dong, H., Zhou, Y., Liu, K., Jia, L., Qin, Y.: Travel distance characteristics analysis using call detail record data. In: 29th Chinese Control and Decision Conference (CCDC), Chongqing, China. DOI: 10.1109/CCDC.2017.7979109 (2017).

[27] Wang, Zh., Zhang, S.: CDR Based Temporal-Spatial Analysis of Anomalous Mobile Users. In: IEEE 14th Intl Conf on Dependable, Autonomic and Secure Computing, 14th Intl Conf on Pervasive Intelligence and Computing, 2nd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech), Auckland, New Zealand. DOI: 10.1109/DASC-PICom-DataCom-CyberSciTec.2016.126 (2016).

[28] Thuillier, E., Moalic, L., Lamrous, S., Caminada, A.: Clustering Weekly Patterns of Human Mobility Through Mobile Phone Data. In: IEEE Transactions on Mobile Computing, volume: 17, issue: 4, pp. 817-830. DOI: 10.1109/TMC.2017.2742953 (2018).

[29] Lu Fan, Zhifeng Zhao, Chen Qi, Rongpeng Li, and Honggang Zhang: A Revisiting to Queueing Theory for Mobile Instant Messaging with Keep-Alive Mechanism in Cellular Networks. In: IEEE International Conference on Communications (ICC), pp. 1-6. IEEE Press, Paris, France. DOI: 10.1109/ICC.2017.7996707 (2017).

[30] Kamal Adli Mehr, Sina Khoshabi Nobar, Javad Musevi Niya: Interarrival time distribution of IEEE 802.15.4 under saturated traffic condition. In: 23rd Iranian Conference on Electrical Engineering, pp. 2164-7054. IEEE Press, Tehran, Iran. DOI: 10.1109/IranianCEE.2015.7146211 (2015).

[31] Brian L. Mark, Yariv Ephraim: On modeling network congestion using continuous-time bivariate Markov chains. In: 45th Annual Conference on Information Sciences and Systems, pp. 1-6. IEEE Press, Baltimore, MD, USA. DOI: 10.1109/CISS.2011.5766118 (2011).

[32] S. Lirio Castellanos-Lopez, Felipe A. Cruz-Perez, Genaro Hernandez-Valdez, Jose Raul Miranda-Tello: Performance Analysis of Mobile Cellular Networks with MMPP Call Arrival Patterns. In: 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS), pp. 2157-4960. IEEE Press, Paris, France. DOI: 10.1109/NTMS.2018.8328711 (2018).

[33] Jeferson Wilian de Godoy Stenico, Lee Luan Ling, Flavio Henrique Teles Vieira: Queuing Modeling Applied to Admission Control of Network Traffic Flows Considering Multifractal Characteristics, vol. 11, pp. 749-758. IEEE Press. DOI: 10.1109/TLA.2013.6533964 (2013).

[34] Achille Pattavina, Alessandra Parini: Modeling voice call inter-arrival and holding time distributions in mobile networks. In: ITC19/ Performance Challenges for Efficient Next Generation Networks, pp. 729-738. (2005).

[35] Lubos Nagy, Jurgen Tombal, Vit Novotny: Proposal of a Queueing Model for Simulation of Advanced Telecommunication Services over IMS Architecture. In: 36th International Conference on Telecommunications and Signal Processing (TSP), pp. 326-330. IEEE Press, Rome, Italy. DOI: 10.1109/TSP.2013.6613945 (2013).

[36] Guido C. Ferrante, Tony Q. S. Quek, Moe Z. Win: Timing Capacity of Queues with Random Arrival and Modified Service Times. In: IEEE International Symposium on Information Theory (ISIT), pp. 370-374. IEEE Press, Barcelona, Spain. DOI: 10.1109/ISIT.2016.7541323 (2016).

[37] Bart lomiej B laszczyszyn, Mohamed Kadhem Karray, Holger Paul Keeler: Using Poisson processes to model lattice cellular networks. In: Proceedings IEEE INFOCOM, pp. 773-781. IEEE Press, Turin, Italy. DOI: 10.1109/INFCOM.2013.6566864 (2013).

[38] Liu Jun, Li Tingting, Cheng Gang, Yu Hua, Lei Zhenming: Mining and modelling the dynamic patterns of service providers in cellular data

network based on big data analysis, vol. 10, pp. 25-36. IEEE Press. DOI: 10.1109/CC.2013.6723876 (2013).

[39] M. S. Parwez, D. B. Rawat and M. Garuba, ”Big Data Analytics for User-Activity Analysis and User-Anomaly Detection in Mobile Wireless Network,” in IEEE Transactions on Industrial Informatics, vol. 13, no. 4, pp. 2058-2065, Aug. 2017, doi: 10.1109/TII.2017.2650206.

[40] H. Nan, X. Zhu and J. Ma, ”An efficient correlation-aware anomaly detection framework in cellular network,” in China Communications, vol. 19, no. 8, pp. 168-180, Aug. 2022, doi: 10.23919/JCC.2022.08.013.

[41] K. Sultan, H. Ali and Z. Zhang, ”Call Detail Records Driven Anomaly Detection and Traffic Prediction in Mobile Cellular Networks,” in IEEE Access, vol. 6, pp. 41728-41737, 2018, doi: 10.1109/ACCESS.2018.2859756.

[42] B. Hussain, Q. Du, A. Imran and M. A. Imran, ”Artificial Intelligence-Powered Mobile Edge Computing-Based Anomaly Detection in Cellular Networks,” in IEEE Transactions on Industrial Informatics, vol. 16, no. 8, pp. 4986-4996, Aug. 2020, doi: 10.1109/TII.2019.2953201.

[43] N. Ruan, Z. Wei and J. Liu, ”Cooperative Fraud Detection Model With Privacy-Preserving in Real CDR Datasets,” in IEEE Access, vol. 7, pp. 115261-115272, 2019, doi: 10.1109/ACCESS.2019.2935759.

[44] S. Zhang, D. Yin, Y. Zhang and W. Zhou, ”Computing on Base Station Behavior Using Erlang Measurement and Call Detail Record,” in IEEE Transactions on Emerging Topics in Computing, vol. 3, no. 3, pp. 444-453, Sept. 2015, doi: 10.1109/TETC.2015.2389614.

[45] E. Thuillier, L. Moalic, S. Lamrous and A. Caminada, ”Clustering Weekly Patterns of Human Mobility Through Mobile Phone Data,” in IEEE Transactions on Mobile Computing, vol. 17, no. 4, pp. 817-830, 1 April 2018, doi: 10.1109/TMC.2017.2742953.

[46] D. Cortés-Polo, L. I. J. Gil, J. Calle-Cancho and J. -L. González-Sánchez, ”A Novel Methodology Based on Orthogonal Projections for a Mobile Network Data Set Analysis,” in IEEE Access, vol. 7, pp. 158007-158015, 2019, doi: 10.1109/ACCESS.2019.2949804.

[47] M. Yin, M. Sheehan, S. Feygin, J. -F. Paiement and A. Pozdnoukhov, ”A Generative Model of Urban Activities from Cellular Data,” in IEEE Transactions on Intelligent Transportation Systems, vol. 19, no. 6, pp. 1682-1696, June 2018, doi: 10.1109/TITS.2017.2695438.

[48] A. Zoha, A. Saeed, H. Farooq, A. Rizwan, A. Imran and M. A. Imran, ”Leveraging Intelligence from Network CDR Data for Interference Aware Energy Consumption Minimization,” in IEEE Transactions on Mobile Computing, vol. 17, no. 7, pp. 1569-1582, 1 July 2018, doi: 10.1109/TMC.2017.2773609.

[49] J. M. DeAlmeida et al., ”Abnormal Behavior Detection Based on Traffic Pattern Categorization in Mobile Networks,” in IEEE Transactions on Network and Service Management, vol. 18, no. 4, pp. 4213-4224, Dec. 2021, doi: 10.1109/TNSM.2021.3125019.

[50] M. Ghahramani, M. Zhou and C. T. Hon, ”Mobile Phone Data Analysis: A Spatial Exploration Toward Hotspot Detection,” in IEEE Transactions on Automation Science and Engineering, vol. 16, no. 1, pp. 351-362, Jan. 2019, doi: 10.1109/TASE.2018.2795241.

[51] A. Kusiak, X. Wei, A. P. Verma and E. Roz (2013). Modeling and Prediction of Rainfall Using Radar Reflectivity Data: A Data-Mining Approach. n IEEE Transactions on Geoscience and Remote Sensing, vol. 51, no. 4, pp. 2337-2342. DOI: 10.1109/TGRS.2012.2210429.

[52] Dananjali, T., Wijesinghe, S., Ekanayake, J. (2020). Forecasting Weekly Rainfall Using Data Mining Technologies. From Innovation to Impact (FITI). E-ISBN: 978-1-6654-1471-5. DOI: 10.1109/FITI52050.2020.9424877.

[53] Nhita, F., Adiwijaya (2013). A rainfall forecasting using fuzzy system based on genetic algorithm. International Conference of Information and Communication Technology (ICoICT). E-ISBN: 978-1-4673-4992-5. DOI: 10.1109/ICoICT.2013.6574557.

[54] Hasan, N., Nath, N., Ch., Rasel, R., I. (2015). A support vector regression model for forecasting rainfall. 2nd International Conference on Electrical Information and Communication Technologies (EICT). E-ISBN: 978-1-4673-9257-0. DOI: 10.1109/EICT.2015.7392014.

[55] N. Madhukumar, E. Wang, Y. -F. Zhang and W. Xiang (2021). Consensus Forecast of Rainfall Using Hybrid Climate Learning Model. In IEEE Internet of Things Journal, vol. 8, no. 9, pp. 7270-7278. DOI: 10.1109/JIOT.2020.3040736.

[56] A. Haidar and B. Verma (2018). Monthly Rainfall Forecasting Using One-Dimensional Deep Convolutional Neural Network. In IEEE Access, vol. 6, pp. 69053-69063. DOI: 10.1109/ACCESS.2018.2880044.

[57] Bin Mohd Khairudin, N., Binti Mustapha, N., Binti Mohd Aris, T., Binti Zolkepli, M. (2020). Comparison of Machine Learning Models for Rainfall Forecasting. International Conference on Computer Science and Its Application in Agriculture (ICOSICA). E-ISBN: 978-1-7281-6907-1. DOI: 10.1109/ICOSICA49951.2020.9243275.

[58] Y. Chen et al. (2022). CNN-BiLSTM Short-Term Wind Power Forecasting Method Based on Feature Selection. In IEEE Journal of Radio Frequency Identification, vol. 6, pp. 922-927. DOI: 10.1109/JRFID.2022.3213753.

[59] Feng, J., Yuan, D., Zhou, A. (2016). A rainfall estimation method based on RBFNN. 2016 IEEE Region 10 Conference (TENCON). E-ISBN: 978-1-5090-2597-8. DOI: 10.1109/TENCON.2016.7848565.

[60] Hossain, M. M., Anwar, A. F., Garg, N., Prakash, M., and Bari, M. (2022). Monthly Rainfall Prediction at Catchment Level with the Facebook Prophet Model Using Observed and CMIP5 Decadal Data. Hydrology, 9(6), 111. https://doi.org/10.3390/hydrology9060111.

[61] Nwokike, C. C., Offorha, B. C., Obubu, M., Ugoala, C. B., and Ukomah, H. I. (2020). Comparing SANN and SARIMA for forecasting frequency of monthly rainfall in Umuahia. Scientific African, 10, e00621. https://doi.org/10.1016/j.sciaf.2020.e00621.

[62] Ray, S., Das, S. S., Mishra, P., Al Khatib, A. M. G. (2021). Time series SARIMA modelling and forecasting of monthly rainfall and temperature in the South Asian countries. Earth Systems and Environment, 5(3), 531-546. https://doi.org/10.1007/s41748-021-00205-w.

[63] Nhita, F., Saepudin, D., Adiwijaya (2015). Comparative Study of Moving Average on Rainfall Time Series Data for Rainfall Forecasting Based

on Evolving Neural Network Classifier. 3rd International Symposium on Computational and Business Intelligence (ISCBI). E-ISBN: 978-1-4673-8501-5. DOI: 10.1109/ISCBI.2015.27.

[64] Q. Zhao, Y. Liu, W. Yao and Y. Yao (2022). Hourly Rainfall Forecast Model Using Supervised Learning Algorithm. In IEEE Transactions on Geoscience and Remote Sensing, vol. 60, pp. 1-9, Art no. 4100509. DOI: 10.1109/TGRS.2021.3054582.

[65] R. Sikder, M. J. Uddin and S. Halder, "An efficient approach of identifying tourist by call detail record analysis," 2016 International Workshop on Computational Intelligence (IWCI), Dhaka, Bangladesh, 2016, pp. 136-141, doi: 10.1109/IWCI.2016.7860354.

[66] S. H. Islam, P. Vijayakumar, M. Z. A. Bhuiyan, R. Amin, V. Rajeev M. and B. Balusamy, "A Provably Secure Three-Factor Session Initiation Protocol for Multimedia Big Data Communications," in IEEE Internet of Things Journal, vol. 5, no. 5, pp. 3408-3418, Oct. 2018, doi: 10.1109/JIOT.2017.2739921.

[67] Alan Johnston, SIP: Understanding the Session Initiation Protocol , Artech, 2000.

[68] William Stallings: Data and Computer Communications, Eighth Edition, Ch. 10, pp. 307-308 (2007).

[69] Behrouz A. Forouzan: TCP/IP Protocol Suite, Fourth Edition, Ch. 25, pp. 748-751 (2010).

[70] Bruce Hartpence: Packet Guide to Voice over IP, Ch. 3, pp. 70-75 (2013).

[71] Feng, H., Shu, Y.: Statistical Analysis of Packet Interarrival Times in Wireless LAN. In: International Conference on Wireless Communications, Networking and Mobile Computing, Shanghai, China. DOI: 10.1109/WICOM.2007.473 (2007).

[72] UN home page: `https://population.un.org/wpp/Download/Standard/Population/`.

[73] Fukushima, T.: Fast transform from geocentric to geodetic coordinates. In: Journal of Geodesy, vol. 73, pp. 603-610. https://doi.org/10.1007/s001900050 (1999).

[74] Thomas, G., B., Weir, M., D., Haas, J., Heil, Ch.: Thomas' Calculus, 13th edition, Ch. 6, pp. 410 (2014).

[75] Chopde, N., R., Nichat, M., K.: Landmark Based Shortest Path Detection by Using A* and Haversine Formula. In: International Journal of Innovative Research in Computer and Communication Engineering, vol. 1. ISSN (Online): 2320-9801 (2013).

[76] Poirier, J. P.: Introduction to the Physics of the earth's interior, second edition, Ch. 7, pp. 223. (2003).

[77] Etalab gouv.fr. Pluviometrie. url: `https://www.data.gouv.fr/en/datasets/pluviometrie`.

[78] Raghava, T., K., V., Wani, S., P. (2014). Internet enabled tipping bucket rain gauge. International Conference on Computer Communication and Informatics. E-ISBN: 978-1-4799-2352-6. DOI: 10.1109/ICCCI.2014.6921828.

[79] Shui Yu, Song Guo (2016). Big Data Concepts, Theories, and Applications.

[80] Md Salik Parwez, Danda B. Rawat, Moses Garuba (2017). Big Data Analytics for User-Activity Analysis and User-Anomaly Detection in Mobile Wireless Network. IEEE Transactions on Industrial Informatics (Volume: 13, Issue: 4).

[81] Yojna Arora, Dinesh Goyal (2017). Big data: A review of analytics methods and techniques. IEEE, 2nd International Conference on Contemporary Computing and Informatics (IC3I).

[82] Gajendra Kumar Vaikar, Prateema Gautam (2016). Data Mining Method Use in Crime Investigation Network CDR Analysis. International Journal of Advances in Computer Science and Cloud Computing, Volume-4, Issue- 1, ISSN: 2321-4058.

[83] Nicola Chemello (2016). Correlating CDR with other data sources. IEEE International Conference on Cybercrime and Computer Forensic (ICCCF).

[84] Kashif Sultan, Hazrat Ali, Zhongshan Zhang: Call Detail Records Driven Anomaly Detection and Traffic Prediction in Mobile Cellular Networks, vol. 6, pp. 41728-41737. IEEE Press. DOI: 10.1109/ACCESS.2018.2859756 (2018).

[85] M.M. Zonoozi, P. Dassanayake, M. Faulkner: Teletraffic modelling of cellular mobile networks. Proceedings of Vehicular Technology Conference (VTC 96), vol. 2, pp. 1274-1277. IEEE Press, Atlanta, USA. DOI: 10.1109/VETEC.1996.501517 (1996).

[86] Andres Rico-Paez, Felipe A. Cruz-Perez, Genaro Hernandez-Valdez: Teletraffic Analysis Formulation Based on Channel Holding Time Statistics. In: IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, pp. 326-330. IEEE Press, Marrakech, Morocco. DOI: 10.1109/WiMob.2009.62 (2009).

[87] G.W. Tunnicliffe, A.R. Murch, A. Sathyendran, P.J. Smith: Analysis of traffic distribution in cellular networks. In: VTC '98. 48th IEEE Vehicular Technology Conference. Pathway to Global Wireless Revolution (Cat. No.98CH36151), vol. 3, pp. 1984-1988. IEEE Press, Ottawa, Ont., Canada. DOI: 10.1109/VETEC.1998.686103 (1998).

[88] Geza Schay: Introduction to Probability with Statistical Applications, Ch. 6, pp. 183-184 (2007).

[89] Robert B. Cooper: Introduction to Queueing theory, second edition, Ch. 2, pp. 50-56 (1981).

[90] Sheldon M. Ross: Introduction to Probability and Statistics for Engineers and Scientists, Fourth Edition, Ch. 5, pp. 176-182 (2009).

[91] Jan Holub, Michael Wallbaum, Noah Smith, Hakob Avetisyan (2018). Analysis of the Dependency of Call Duration on the Quality of VoIP Calls. IEEE Wireless Communications Letters, volume: 7, issue: 4. Pages: 638-641. DOI: 10.1109/LWC.2018.2806442.

[92] Y. Yuan, J. Liu, W. Chi, G. Chen and L. Sun, "A Gaussian Mixture Model Based Fast Motion Planning Method Through Online Environmental Feature Learning," in IEEE Transactions on Industrial Electronics, vol. 70, no. 4, pp. 3955-3965, April 2023, doi: 10.1109/TIE.2022.3177758.

[93] Min Tang, B. Pellom, K. Hacioglu (2003). Call-type classification and unsupervised training for the call center domain. IEEE Workshop on Automatic Speech Recognition and Understanding (IEEE Cat. No.03EX721). St Thomas, VI, USA. DOI: 10.1109/ASRU.2003.1318429.

[94] Pedro Domingos (2012). A few useful things to know about machine learning. Communications of the ACM magazine, volume: 55, issue: 10, Pages: 78-87. DOI: 10.1145/2347736.2347755.

[95] Christopher M Bishop, Pattern Recognition and Machine Learning (Information Science and Statistics), Berlin, Heidelberg:Springer-Verlag, 2006, p430, ISBN-13: 978-0387-31073-2.

[96] Y. Gao, X. Xu, Z. Yan and M. Shahidehpour, ”Gaussian Mixture Model for Multivariate Wind Power Based on Kernel Density Estimation and Component Number Reduction,” in IEEE Transactions on Sustainable Energy, vol. 13, no. 3, pp. 1853-1856, July 2022, doi: 10.1109/TSTE.2022.3159391.

[97] J. Zhu, R. Guo, Z. Li, J. Zhang and S. Pang, ”Registration of Multi-View Point Sets Under the Perspective of Expectation-Maximization,” in IEEE Transactions on Image Processing, vol. 29, pp. 9176-9189, 2020, doi: 10.1109/TIP.2020.3024096.

[98] C. -M. Hsu, F. -L. Lian and C. -M. Huang, ”A Systematic Spatiotemporal Modeling Framework for Characterizing Traffic Dynamics Using Hierarchical Gaussian Mixture Modeling and Entropy Analysis,” in IEEE Systems Journal, vol. 8, no. 4, pp. 1129-1138, Dec. 2014, doi: 10.1109/JSYST.2013.2253197.

[99] I. S. Gradshteyn and I. M. Ryzhik, Table of Integrals Series and Products, Oxford:Academic Press, 2007, p1066, ISBN-13: 978-0-12-373637-6.

[100] N. Gallego-Ortiz and D. S. Femández-Mc-Cann, ”Efficient implementation of the EM algorithm for mammographic image texture analysis with multivariate Gaussian mixtures,” 2011 IEEE Statistical Signal Processing Workshop (SSP), Nice, France, 2011, pp. 821-824, doi: 10.1109/SSP.2011.5967832.

[101] S. Fong, J. Harmouche, S. Narasimhan and J. Antoni, ”Mean Shift Clustering-Based Analysis of Nonstationary Vibration Signals

for Machinery Diagnostics," in IEEE Transactions on Instrumentation and Measurement, vol. 69, no. 7, pp. 4056-4066, July 2020, doi: 10.1109/TIM.2019.2944503.

[102] R. Yamasaki and T. Tanaka, "Properties of Mean Shift," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 42, no. 9, pp. 2273-2286, 1 Sept. 2020, doi: 10.1109/TPAMI.2019.2913640.

[103] D. Comaniciu and P. Meer, "Mean shift: a robust approach toward feature space analysis," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 5, pp. 603-619, May 2002, doi: 10.1109/34.1000236.

[104] Z. Gao, L. Ding, Q. Xiong, Z. Gong and C. Xiong, "Image Compressive Sensing Reconstruction Based on z-Score Standardized Group Sparse Representation," in IEEE Access, vol. 7, pp. 90640-90651, 2019, doi: 10.1109/ACCESS.2019.2927009.

[105] Ł. Gałka, P. Karczmarek and M. Tokovarov, "Isolation Forest Based on Minimal Spanning Tree," in IEEE Access, vol. 10, pp. 74175-74186, 2022, doi: 10.1109/ACCESS.2022.3190505.

[106] M. Yang, L. Huang and C. Tang, "K-Means Clustering with Local Distance Privacy," in Big Data Mining and Analytics, vol. 6, no. 4, pp. 433-442, December 2023, doi: 10.26599/BDMA.2022.9020050.

[107] C. Sandoval, E. Pirogova and M. Lech, "Adversarial Learning Approach to Unsupervised Labeling of Fine Art Paintings," in IEEE Access, vol. 9, pp. 81969-81985, 2021, doi: 10.1109/ACCESS.2021.3086476.

[108] Y. Lin, E. Giacoumidis, S. O'Duill and L. P. Barry, "DBSCAN-Based Clustering for Nonlinearity Induced Penalty Reduction in Wavelength Conversion Systems," in IEEE Photonics Technology Letters, vol. 31, no. 21, pp. 1709-1712, 1 Nov.1, 2019, doi: 10.1109/LPT.2019.2942961.

[109] F. Wang, J. Chen and F. Liu, "Keyframe Generation Method via Improved Clustering and Silhouette Coefficient for Video Summarization," in Journal of Web Engineering, vol. 20, no. 1, pp. 147-170, January 2021, doi: 10.13052/jwe1540-9589.2018.

[110] Y. Wang, X. Du, Z. Lu, Q. Duan and J. Wu (2022). Improved LSTM-Based Time-Series Anomaly Detection in Rail Transit Operation Environments. In IEEE Transactions on Industrial Informatics, vol. 18, no. 12, pp. 9027-9036. DOI: 10.1109/TII.2022.3164087.

[111] X. Zhang, Y. Lei, H. Chen, L. Zhang and Y. Zhou (2021). Multivariate Time-Series Modeling for Forecasting Sintering Temperature in Rotary Kilns Using DCGNet. In IEEE Transactions on Industrial Informatics, vol. 17, no. 7, pp. 4635-4645. DOI: 10.1109/TII.2020.3022019.

[112] M. Ni, Q. He and J. Gao (2017). Forecasting the Subway Passenger Flow Under Event Occurrences With Social Media. In IEEE Transactions on Intelligent Transportation Systems, vol. 18, no. 6, pp. 1623-1632. DOI: 10.1109/TITS.2016.2611644.

[113] U. A. Bhatti et al. (2021). Time Series Analysis and Forecasting of Air Pollution Particulate Matter (PM2.5): An SARIMA and Factor Analysis Approach. In IEEE Access, vol. 9, pp. 41019-41031. DOI: 10.1109/ACCESS.2021.3060744.

[114] H. Kaur and S. K. Sood (2020). Energy-Efficient IoT-Fog-Cloud Architectural Paradigm for Real-Time Wildfire Prediction and Forecasting. In IEEE Systems Journal, vol. 14, no. 2, pp. 2003-2011. DOI: 10.1109/JSYST.2019.2923635.

[115] Samudre, P., Shende, P., Jaiswal, V. (2019). Optimizing Performance of Convolutional Neural Network Using Computing Technique. E-ISBN: 978-1-5386-8075-9. DOI: 10.1109/I2CT45611.2019.9033876.

[116] M. Pan et al. (2020). Water Level Prediction Model Based on GRU and CNN. In IEEE Access, vol. 8, pp. 60090-60100. DOI: 10.1109/ACCESS.2020.2982433.

[117] Madhuri, Ch., R., Chinta, M., Kumar, V., Ph. (2020). Stock Market Prediction for Time-series Forecasting using Prophet upon ARIMA. 7th International Conference on Smart Structures and Systems (ICSSS). E-ISBN: 978-1-7281-7223-1. DOI: 10.1109/ICSSS49621.2020.9202042.

[118] A. N. Alkawaz, A. Abdellatif, J. Kanesan, A. S. M. Khairuddin and H. M. Gheni (2022). Day-Ahead Electricity Price Forecasting Based on

Hybrid Regression Model. In IEEE Access, vol. 10, pp. 108021-108033. DOI: 10.1109/ACCESS.2022.3213081.

[119] H. Liu, W. Liang, L. Rai, K. Teng and S. Wang (2019). A Real-Time Queue Length Estimation Method Based on Probe Vehicles in CV Environment. In IEEE Access, vol. 7, pp. 20825-20839. DOI: 10.1109/ACCESS.2019.2898424.

[120] Bensalma, A. (2015). New fractional Dickey Fuller test. 6th International Conference on Modeling, Simulation, and Applied Optimization (ICMSAO). E-ISBN: 978-1-4673-6601-4. DOI: 10.1109/ICMSAO.2015.7152263.

[121] S. Yang et al. (2019). Long-Term Prediction of Significant Wave Height Based on SARIMA Model in the South China Sea and Adjacent Waters. In IEEE Access, vol. 7, pp. 88082-88092. DOI: 10.1109/ACCESS.2019.2925107.

[122] U. M. Sirisha, M. C. Belavagi and G. Attigeri (2022). Profit Prediction Using ARIMA, SARIMA and LSTM Models in Time Series Forecasting: A Comparison. In IEEE Access, vol. 10, pp. 124715-124727. DOI: 10.1109/ACCESS.2022.3224938.

[123] Y. Li, Y. Yang, K. Zhu and J. Zhang (2021). Clothing Sale Forecasting by a Composite GRU–Prophet Model With an Attention Mechanism. In IEEE Transactions on Industrial Informatics, vol. 17, no. 12, pp. 8335-8344. DOI: 10.1109/TII.2021.3057922.