Eötvös Loránd University
Faculty of Informatics

Data Analysis on Telecom Data Sets: Normal User Behaviours and Anomaly Detection Models.

Tomas Horvath
Head of the Data Science Department at ELTE

Andrea Galloni
Computer Science

Maurizio Marchese
Professor at UniTn

Máder Miklós Péter
Business Developer at Magyar Telekom

Budapest, 2017

Dedica - Dedication

A tutte le persone che mi sono vicine e lo sono state in questo percorso educativo.
Un ringraziamento particolare ai miei genitori, a mia nonna...
A chi c'é ed a chi c'é stato ma non c'é piú.

Non Literal Translation

To the people who love me..

Contents

1 Introduction **1**

1.1 The Role of Data Science in Modern Telecom Companies 1

1.2 Our Study 2

1.3 Outline 3

2 Data Set and Framework **4**

2.1 General Overview 4

2.1.1 Call Detail Records 5

2.2 The Data-Set 6

2.2.1 Privacy Laws Compliance 6

2.2.2 The Data-Set Structure 7

2.2.3 Cell Towers 7

2.2.4 CRM 10

2.2.5 MSC 12

2.2.6 NGPRS 13

3 The Framework and Data Preprocessing **16**

3.1 The Framework 16

3.1.1 Bash 16

3.1.2 Python 17

3.1.3 Python Pandas 17

3.2 The Data Preprocessing Phase 17

3.2.1 Decompression Phase 17

3.2.2 Inconsistency Analysis 19

3.2.3 Data Cleaning and Aggregation 21

3.2.4 Data Filtering and Reshaping 23

4 The Actual Literature **25**
4.1 Mobile Telecoms and Mobility 26

5 Our Approaches **30**
5.1 General Overview 30
5.1.1 Preliminary Considerations 30
5.1.2 Dividing the Space 31
5.1.3 Time Series 32
5.1.4 The Nature of our DataSet 35
5.2 The Models 40
5.2.1 The Standard Deviation Model 40
5.2.2 The Statistical Approach 43
5.2.3 Evaluation Metrics 48

6 Experimental Results **49**
6.1 NGPRS T60 STD 52
6.1.1 The Pure NGPRS Dataset STD 52
6.1.2 The Augmented NGPRS Dataset STD 54
6.2 NGPRS T1 WMM 56
6.2.1 WMM Not Overlapped 56
6.2.2 WMM Overlapped Symmetric Windows 57
6.2.3 WMM Overlapped Asymmetric Windows 58
6.3 Results Summary and Considerations 59

7 Conclusions and Future Work **60**
7.1 Final Considerations 60
7.2 Future Work 61

Chapter 1

Introduction

1.1 The Role of Data Science in Modern Telecom Companies

Nowadays Big Data is a strategic and vital source of digital innovation and services improvement. Communications service providers that want to be innovative and maximize their revenue potential must have the right solution in place so that they can harness the volume, variety and velocity of data coming into their organization and leverage actionable insight from that data.

Given the premises above, it is a fact that sometimes telecom providers do not have all the required domain knowledge to fully exploit all the the precious information contained on their collected data and this issue leads to a lack of considerably profitable businesses. The evolving scenario shows that data can be seen as a product (in respect of national and EU privacy laws and regulations) that companies and researchers can exploit to archive their goals. Even more data can be exploited to build up telecom-data-driven services creating new opportunities for third-party companies to establish business-to-business deals or even more to improve internal business intelligence processes.

As previously mentioned telecom providers are daily collecting huge amounts of various data related to users and their behaviours, one of those is geo-positioning, this kind of data is continuously collected and trends highlight that the precision of the positioning through Cell Tower "Triangulation" in urban areas is getting more and more accurate; in fact, to provide a better service to many customers in a given area, telecom providers have been improving their networks with many smaller cells, each with its own antenna. The areas (at least in an urban scenario) are served by these new cells are much smaller than the old ones, resulting in a smaller geographic area (leading to an increased location precision).

Companies like ParkNav are exploiting this kind of positioning data to predict the probability to find a free parking spot in a given street, but this is only one application within a huge set.

1.2 Our Study

Our study aims to discover a new possible Telecom data exploitation, its applications, its effectiveness, and its possible impact on the modern society in order to improve human mobility and well-being answering to the following questions:

- Which data do mobile Telecom operators own?
- How can we exploit the "tacit" knowledge contained in it?
- With which effectiveness we can use it to serve the society development?

In order to provide a real and well established scientific answer, given these generic premises, some scientific questions naturally arise:

1. How do we extrapolate knowledge from the big amount of information we collect?
2. What are the most important (relevant) components defining a user behaviour?
3. How do we represent and model this information in such a way that can be described with a mathematical language in order to generate a model understandable and computable by a machine?
4. Is it possible to predict with a certain generalization the future behaviour of a set of users?

The applications of a predictive system would cover a wide range of use cases. In this specific context the usual behaviour of users will be exploited as a baseline to detect extraordinary events such as car accidents or traffic jams.

In order to archive the goal two anomaly detection techniques will be used to spot these unexpected events. The results of this research coupled with an optimization process research could be useful to notify users the best route to follow in order to avoid as much as possible delays caused by unexpected events mentioned above, possibly, avoiding to generate new traffic jams.

1.3 Outline

The structure of the thesis will follow the following schema: in chapter 2 all the details about the data will be provided, in chapter chapter 3 we will describe the pre-processing operations performed over it. Then in chapter 4 an overview of the current status of the research will be provided. Following with chapter 5 some theoretical background will be provided in order to formally define the theoretical concepts useful to formally describe methods used. Only then in the same chapter we will present two different models about anomalous events detection. In chapter 6 we will discuss about experimental results. While finally on chapter 7 conclusions about the research will be provided coupled with the future work of this research.

Chapter 2

Data Set and Framework

2.1 General Overview

As mentioned in the introduction chapter, in this specific research framework, we are trying to understand human mobility through the analysis of mobile phone data-sets. This area of research has emerged about a decade ago, given the availability of anonymized data-sets, this field recently has become a stand-alone research topic. In this scenario work we study the global behaviour of aggregated mobility in order to detect behavioural anomalies.

Mobile phones, especially smart-phones have reshaped people's communication habits in the first the years of the actual century. During the past times the world penetration of mobile phone ownership has raised from 12% of the world population up to 96% in 2014 with a penetration of 128% in the developed world and 90% in developing countries [2] allowing individuals to be connected even in the most remote world's areas. Provided these premises we can assert that nowadays mobile phones are ubiquitous. In most countries of the developed world, the coverage reaches 100% of the population. Due to their ubiquity, these handsets have stimulated the creativity of the scientific community. The essence of mobile phones have revealed them to be a source of rich data.

The first application of a study of phone logs (not mobile) appeared in 1949, with a paper by George Zipf modeling the influence of distance on communication [12]. Since then, phone logs have been studied in order to infer relationships between the volume of communication and other parameters, but, the actual penetration of mobile phone generated data in massive quantities as well. The power of actual computers and methods make them able to handle those data efficiently. This fact has definitely boosted the research interest in that domain. Being personal goods, mobile phones enabled to infer real social networks

from their CDRs, while fixed phones are shared by users of one same geographical space. The communications logs (CDRs) recorded on a mobile phone are thus representative of a part of the social network of one single person, where the records of a fixed phone reaches their limits.

2.1.1 Call Detail Records

A Call Detail Record (CDR) is a data record produced by a telephone exchange or other telecommunications equipment that documents the details of a call or other telecommunications transaction (e.g., Short Message Service also known as SMS) that passes through a telecom provider infrastructures.

The Call Detail Records (CDRs), needed by the mobile phone operators for business purposes, hold a huge amount of information on how, when, and with whom people communicate. The record contains various attributes of the call, such as time, duration, completion status, source number, and destination number. Call detail records are useful for many operational tasks; in fact, for telecom providers, CDRs represent a critical resource for the production of revenue, in fact they provide the basis for the generation of telephone bills while in the field of law enforcement, call detail records provide useful information that can help to identify suspects, call data logs can reveal details such as some individual's relationships with other individuals, communication and/or behavior patterns, and even location data that can establish the estimated location of an individual during the time spent on a mobile phone call. Mobile phone CDRs, contain logs on communications between millions of people at a time, and contain real observations of communications between them. As mentioned before, CDRs also contain estimated localization logs and could be integrated to external profiling data about customers such as age, gender and their environment. Such a combination of heterogeneous data-sets makes mobile phone's CDRs an extremely rich source of data for scientists as the logs hold user profiling information often over a wide period of time.

Along the past few years there have been a growing interest on research based on the analysis of CDRs and has been for a few years the leading topic of NetMob, an international conference on the analysis of mobile phone data-sets. Lately the telecom company Orange has, proposed the D4D challenge, whose concept is to give access to a large number of research teams throughout the world to the same data-set from an African country. Their purpose is to make suggestions for development, on the basis of the observations extracted from the mobile phone data-set. Of course, there are restrictions on the availability of

some types of data and on the projected applications. First,communication's contents are not stored by the operator; second, while mobile phone operators have access to all the information filed by their customers and the CDRs, they may not give the same access to all the information to a third party (researchers included), depending on their own privacy policies and the laws that apply in the country of application.
Anyway, names and phone numbers are never transmitted to external parties. In some countries, location data, such as the base stations at which each call is made, have to remain confidential in some cases operators are even not allowed to use this kind of data for internal private research or business intelligence activities.

For the researchers point of view a mobile phone log has two advantages over fixed phones logs: first, its owner has almost always the possibility to pick up a call, thus the communications are reflecting the temporal patterns of communications in great detail, and second, the positioning data of a mobile phone allows to track the displacements of its owner over the time.

2.2 The Data-Set

In this section we are going to provide all the details and some statistics about the specific data-set on which all the experiments have been performed.

The dataset have been kindly provided by *Magyar Telekom* and it is composed by several *csv* (comma separated values) source files; all the information contained on such files have been organized on a daily basis. The entire information contained within the dataset covers a range of 31 days more precisely between 15th September 2016 and 15th October 2016 included.

2.2.1 Privacy Laws Compliance

In order to be compliant to the actual *Data Protection Directive* (officially *Directive 95/46/EC* on the protection of individuals with regard to the processing of personal data and on the free movement of such data) which regulates the processing of personal data within the European Union, the dataset have been anonymized by the company on a daily basis, only after this process it have been donated to our research group.
Personal details such as full names, phone numbers and billing addresses have been removed so that it is practically impossible to follow and profile a specific user over its

interaction with the network for more than one day. Given the nature of the data it is barely impossible to identify and follow users over the whole time covered by the dataset. As we will see this anonymization process had an impact on some decision processes over our research techniques.

2.2.2 The Data-Set Structure

The dataset is mainly divided in four types of files:

- Cell Towes Global Positioning System (namely: cells.csv)
- CRM Customer Relationship Management Data (namely: crm_YYYYMMdd.csv)
- MSC Mobile Switching Center (namely: msc_YYYYMMdd.csv)
- NGPRS Network General Packet Radio Service (namely: ngprs_YYYYMMdd)

2.2.3 Cell Towers

This file contains details about the cell towers displaced within Hungary. As shown in table 2.1 the file is composed by four columns.

Generation	NGPRS ID	MSC ID	Latitude	Longitude
2G	2163000220BBC	0112F60300220BBC	47.4650098389	19.1169265800
3G	2163010D8E999	0212F60310D8E999	46.8789007026	19.2378720375
4G	2163000429A1D	0212F60300429A1D	47.7321115414	18.8428974285

Table 2.1: cells.csv Example

Follows the description of the file content:

Generation

This column indicates which technology is powering the cell tower signal. This record contains the number of mobile telecommunications technology the domain of this column can be respectively: 2G, 3G or 4G.

MSC ID

This value represent the id of the cells contained on the MSC files logs. In fact for every interaction between a mobile phone and the mobile switching center the identifier string of

the voice call cell where the phone is connected is logged and stored. This value is unique for every record and it resulted useful to detect the geo-localization of the call event.

NGPRS ID

This value represent the id of the cells contained on the NGPRS files logs. In fact for every interaction between a mobile phone and the gprs data network the identifier string of the cell where the phone is connected is collected and stored. This value is unique for every record and it resulted useful to detect the geo-localization of the data event.

Latitude

Latitude is a geographic coordinate that specifies the north–south position of a point on the Earth's surface. Latitude is an angle which ranges from 0° at the Equator to 90° (North or South) at the poles. In this case the information is represented on Decimal degrees (DD). This value represents the exalt latitude of the specific cell tower.

Longitude

Longitude is a geographic coordinate that specifies the east-west position of a point on the Earth's surface. By convention, the Prime Meridian, which passes through the Royal Observatory, Greenwich, England, was allocated the position of zero degrees longitude. Also in this case the information is represented on Decimal degrees (DD). This value represents the exalt longitude of the specific cell tower.

From Figures: 2.1 and 2.2 is clear that (provided the mostly flat characteristics of the Hungarian landscapes) the density of the cells follows almost the population density over the territory. In fact the cell towers placement is more dense within urban areas rather than on country sides. This increases the location resolution in urban areas rather than in the rest of the territory. Important to note is the fact that in a non urban environments cell towers appear to be displaced according to the mobility infrastructure highways included. This latter property of the dataset makes easier to monitor the highway mobility activity from the telecommunication perspective, in fact the activity of the cells displaced far from cities and close to highways can provide data about travellers with a significant lower level of what we define static noise (telecommunication activities performed by non travelling subscribers). When the number of non moving subscribers is low in general the cell log activity result to be lower and well correlated with the highways status making easier to infer the activity of a system from the other.

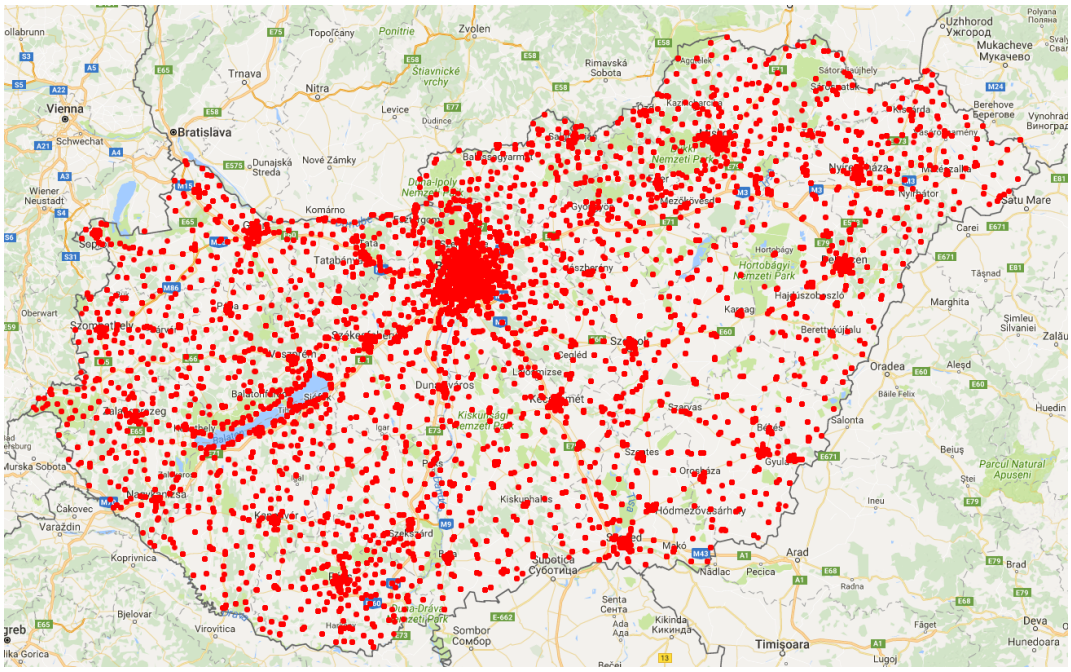


Figure 2.1: Cell Towers Placement in Hungary

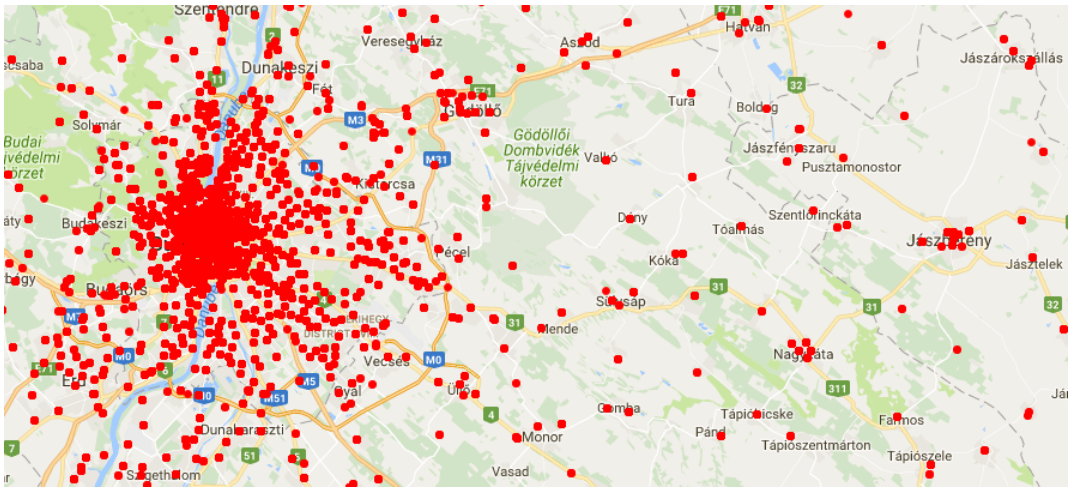


Figure 2.2: Budapest Urban Area and Rural Area Comparison

2.2.4 CRM

The CRM files contain all the details about users and some generic data about profiling information. Provided the daily anonymization of the data the dataset does not contain just a file about subscribers but there is a file for every day covered by the dataset.

IMSI	ZIP	City	Sex	Age	Cat.	SS	ARPU	4G	SED
5804D052	3794	Boldva	F	23	SMB	N	1	Y	201705
584E7114	6783	Ásotthalom	F	29	LAK	N	2	Y	201711
584DB2CB	3530	Miskolc	M	51	LAK	N	2	Y	201709

Table 2.2: cells.csv Example

Follows the description of the file content:

IMSI

The International Mobile Subscriber Identity or IMSI (daily anonymized) is used to identify the user of a mobile network and is a unique identifier associated with the cellular network. It is represented as a 64 bit infromation field and is sent by the phone to the network. It is also used for acquiring other details of the mobile in the home location register (HLR) or as locally copied in the visitor location register. The IMSI identifier is usually composed by 15 digits, but can be shorter.

ZIP

The ZIP code is a postal code, in hungary it is composed by a series of four digits, it is usually included in a postal address for the purpose of sorting mail. The assignment of this code is aimed to divide geographical areas in smaller sections. It gives an idea of the home site of the subscriber. The total number of zip codes contained in the dataset is 4205.

City

The name of the city where the subscriber has its residence. The number of the unique records is exactly 5353. Provided that the number of unique ZIP codes is higher than the number of cities in this case the dataset might have some inconsistency or ZIP codes might be the same for several little villages.

Sex

The sex column contains the gender of the subscriber, it can have two values M for male and F for female, in several records this field have been observed to be missing.

Age

This value contains the age of the subscriber expressed in years. The median value of the age resulted to be 44.

Cat.

This field contains a variable describing the client's contract type basically it helps to define if the subscriptions are for business proposes or not providing details on the size of the business.

The possible values are:

- *LAK* for private individuals
- *SH+* for private individual entrepreneurs
- *SMB* for small and medium business
- *TSS* TSS for T-systems subscribers

SS

This record is used to classify users if they have both a fixed line subscription and a mobile subscription with the same operator.

ARPU

Average revenue per unit, usually abbreviated to ARPU, is a measure used primarily by consumer communications and networking companies, this value defines how the subscriber is used to spend on average. This column has a discrete co-domain, in fact the values are in a range between 1 and 5. In this case the user is classified based on its best fitting category 5 means that the user is very valuable for the company revenues.

4G

This field simply indicates if the SIM card of the customer supports 4G or not.

SED

Namely Subscription end date, this field indicates the year and the month when the customer's subscription will expire.

2.2.5 MSC

The MSC (namely *mobile switching center*) files contain all the details about users interactions with the voice infrastructure such as voice call sent/received or SMS sent/received. This kind of information represents the core of our research as it is coupled with the GPRS data give to us a real perspective of the users activity with the network. This kind of events stored in the MSC files always contain time logs. The number of daily logs contained into each file is about 30 millions.

EID	SIMSI	IMSI	TAC	EID	Date T.	CID
3..1	580A24EE	58753827	35087470	7	2016-09-15 09:46:47	216..963
1..1	580A24EF	5876EE8A	35727105	3	2016-09-15 19:20:42	216..8C0
3..4	580A24EE	58753827	35087470	7	2016-09-15 09:46:44	216..963

Table 2.3: msc.csv Example

Follows the description of the file content:

EID

This column represent the identifier of the event. This value is unique and is an integer.

SIMSI

This value represents the International Mobile Subscriber Identity of the subject performing the action.

IMSI

This value represents the International Mobile Subscriber Identity of the object of the action.

TAC

The TAC namely Type Allocation Code is the initial eight-digit portion of the 15-digit IMEI code used to uniquely identify mobile devices. The Type Allocation Code identifies the specific model of the mobile telephone for use on a GSM, UMTS networks. The TAC number is interesting because can give insight on the actual mobile handsets market. Since the IMEI code is unique for every device it have been truncated for privacy reasons.

EID

This field provides information about the kind of event that triggered the msc log.
The possible values are:

- *1* Outgoing call
- *3* Incoming call
- *7* Incoming SMS
- *9* Outgoing SMS

Date T.

This field in terms of our research is the most valuable one, in fact it provides the a temporal coordinate about the time when the event have been triggered.

CID

This field represents the specific cell tower identifier, even in this case this record is fundamental for our research, in fact if merged with the information contained in the cells file provides an estimation of the geographic location of the described event.

2.2.6 NGPRS

The NGPRS file (namely *Network General Packet Radio Service*) files contain all the details about users interactions with the data services such as data requests sent or received. This kind of information is the most valuable in our research as it is coupled with the MSC data give to us a real perspective of the users activity with the network. This kind of events stored in the NGPRS files always contain time logs. The number of daily logs contained into each file is about 200 millions.

EID	IMSI	TAC	Event T.	Date T.	Cell ID
673279649	580A7812	35945707	84	2016-09-15 13:30:41	1012F6030002F303
673279649	580A7812	35945707	85	2016-09-15 00:27:50	1012F60300097402
673279649	580A7812	35945707	18	2016-09-15 17:44:08	00828CEA

Table 2.4: cells.csv Example

EID

This column represent the identifier of the event. This value is unique and is an integer.

IMSI

This value represents the International Mobile Subscriber Identity of the subject performing the action.

TAC

As for the MSC file the TAC namely Type Allocation Code is the initial eight-digit portion of the 15-digit IMEI code used to uniquely identify mobile devices. The Type Allocation Code identifies the specific model of the mobile telephone for use on a GSM, UMTS networks. The TAC number is interesting because can give insight on the actual mobile handsets market. Since the IMEI code is unique for every device it have been truncated for privacy reasons.

Event Type

This field describes the type of the event, it can contains three values:

- *SGSN* The Serving GPRS Support Node (SGSN) is a main component of the GPRS network, which handles all packet switched data within the network, e.g. the mobility management and authentication of the users. This event is logged every authentication process is triggered.
- *SGW* The SGW routes and forwards user data packets, while also acting as the mobility anchor for the user plane during inter-Node handovers. This event is enabled every time the user equipment changes node.
- *PGW* The PDN Gateway provides connectivity from the UE (User Equipment) to external packet data networks by being the point of exit and entry of traffic for the

UE. This event is logged every time the user performs data requests or receives data packets.

Date T.

This field in terms of our research is the most valuable one, in fact it provides the a temporal coordinate about the time when the event have been triggered.

Cell ID

This field represents the specific cell tower identifier, even in this case this record is fundamental for our research, in fact if merged with the information contained in the cells file provides an estimation of the geographic location of the logged event.

Comments

In this section we described how the dataset is composed. The CRM contains anonymized information about customers. The localization data is contained on the cells file. Finally the communication events are stored in two separate files the MSC for voice data (and SMS) and NGPRS for the internet connection traffic. Given the the presented characteristics of the data the location obtained by merging the information contained in the MSC and NGPRS logs and the cell towers location does not provide the exact location of the subscriber but simply the location of the cell tower from where the event was registered, this characteristic of the information has an impact on the precision of the entire system because the area covered by cell towers (in rural areas) can be about several square kilometers. The range covered by a specific cell tower depends on several factors such as: the average *crowdedness* of the specific area and its geological properties. The dataset contained exactly 39898 different cell towers locations placed all over Hungary (Figure 2.1).

Chapter 3

The Framework and Data Preprocessing

In this section we will give a brief overview of the technologies used to handle the experiments on the data and then we will describe in detail how the data preprocessing operations have been performed.

3.1 The Framework

3.1.1 Bash

In order to reshape and clean the database all the power and simplicity of the Linux *Burning Shell* and *Python* have been used. In fact tools like *awk*, *sed*, and other text manipulation commands have been used. All the files have been decompressed using the GNU *tar* utility. During the preprocessing phase all the inconsistent logs have been removed, only the strict necessary data have been kept such as event log time and cell position. All the customer data have been filtered out reshaping the files based on the cell towers activity. So given this fact our perspective is not centered on the single user activity but rather on the aggregated activity around the cells.

The temporal resolution of the time series is in seconds, but, as we will see further in our research a more aggregated time resolution provided a better understanding of the data both during the visualization process and the experimental one.

3.1.2 Python

The simplicity and the "programmer-friendly characteristics" of Python made it the best choice to develop the code to validate our techniques in short time and without many language related issues. Thank to this we gained time and resources to study and to develop the proposed solutions.
Python is a free and open-source general-purpose, high-level interpreted programming language. The Python's dynamic type system, the automatic memory management and the software design in general make it simple to learn and certainly a good instrument for prototyping in short time terms. Python interpreter is available for many platforms, this make it widely portable. As any general-purpose programming language it does not come with specific built in modules to operate with time series itself.

3.1.3 Python Pandas

In order to get a time series oriented framework the *Python Pandas* package have been used. We choose Pandas because is fast, and provides flexible data structures, it is designed to make working with labeled data both easy and intuitive. It aims to be the fundamental high-level, language independent tool for real world data analysis.
Pandas is built on top of *NumPy* and is intended to integrate well within a scientific computing environment with many other 3rd party libraries, in fact in order to plot graphs the *PyPlot* package have been used with a 100% compatibility.

3.2 The Data Preprocessing Phase

I this section we are going to provide technical details about the preprocessing phase.

3.2.1 Decompression Phase

Since the original dataset files have been delivered in a compressed *.tgz* format a bash script have been wrote for the decompressing phase (Code Snippet 3.1). The script is subdivided in two main parts: a for loop and a *parallel()* function. For every file except CRM ones (regular expressions have been used to filter file names) a new process that executes the *parallel()* function is spawned.

The *tar -xvf <compressed_file_path>* command decompresses the file (passed as parameter) in the script execution folder. Finally after the decompression phase all the decompressed files are moved and renamed from the temporary decompression folder in the proper

directory based on the nature of the file (MSC or NGPRS).

The compressed MSC files average dimension was around 500MB while decompressed about 2.5GB with a compressing ratio around 500% containing on average about 30million logs per day, while NGPRS files originally before decompression on average occupy 2.5GB, and after about 15/16GB with a compression ratio close to 1000% with an average of 200million logs per file.

```
#!/bin/bash
# FILENAME: decompressor.sh


RES_FOLDER=/[path_to_the_archive]/;
TMP_FOLDER=/[path_to_tmp_working_directory]/;
DEST_FOLDER=/[main_path_of_destination_folder]/;

parallel(){

    # the file name to decompress is passed as parameter
    file=$1

    # decompressing command -x: extract, -v: verbose, -f folder name
    tar -xvf $file

    # building the name of the file
    fname_0=${file#$RES_FOLDER}
    fname=${fname_0%.tgz}

    # creating the name of the final file
    s_tmp=${file#$RES_FOLDER'motionl_MT_'}
    new_f_name=${s_tmp%.tgz}

    # Different destination folders depending on the type of file #RegEx
    if [[ $file =~ .*msc.* ]]; then
        # Move the file to the right folder
        mv -v  $TMP_FOLDER$fname $DEST_FOLDER"msc_temp/"$new_f_name

    elif [[ $file =~ .*ngprs.* ]]; then
        # Move the file to the right folder
        mv -v $TMP_FOLDER$fname $DEST_FOLDER"ngprs_temp/"$new_f_name

    else
        # Debugging log
        echo "NO MATCH! Skipping $file"

    fi

}

# for every file in the data resource folder
for f in $RES_FOLDER*
    do
        # if not a CRM file then (#RegEx)
        if !  [[ $f =~ .*crm.* ]]; then
            # parallel decompresses the file;
            parallel $f >> out.log &

        else
            # Debugging log
            echo "SKIPPED: $f"

        fi
    done
```

Code Snippet 3.1: Parallel Decompressing Script

3.2.2 Inconsistency Analysis

After the decompression phase we started to clean the data, but unfortunately during this process we realized that our dataset contained some inconsistencies, in fact, some records (both for MSC and NGPRS files) contained unique cells identifiers that did not match with our list of cell towers identifiers.

In order to better understand and get a quantitative measure of this inconsistencies and the number of affected records we decided to write an ad hoc test. This analysis have been performed to understand if these inconsistencies would have an relevant impact in our research or not. On Code Snippet 3.2 we scan all the dataset files and try to match the identifiers, in case this does not happen then we collect some information about it generating some statistics. The test scanned the whole dataset involving billion of records taking more than five hours (exactly 5h 31m 17s) to run in a single threaded python process.

Table 3.1 shows the statistics reported by our test; the table reports the total number of processed files, the total number of records contained in our dataset, the total number of the cells found on MSC and NGPRS files but not found in the cells file, the number of cells listed in the cells file, the total number of all the cells found in the CRM and NGPRS files and finally the number of the so called *phanthom cells* that is the number of cells contained in the cells file but not referenced at all from both the MSC or NGPRS files.

Value Name	**N**
Number of Processed Files	62
Total Number of Records	6,584,008,885
Number of Affected Records	1,467,264,300
Number of Listed Cells	75,502
Total Number of Discovered Cells	1,651,060
Number of Missing Cells	1,592,811
Number of Phantom Cells	17,253

Table 3.1: Cell Inconsistency Report

The number of affected records is considerably high, anyway we decided to go further in the research, the missing values are not randomly distributed but they involve precise locations.

```
#!/usr/bin/python3
# FILENAME: cell_consistency.py

import os, csv

h_cells_f  =  '/[base_path]/cells.csv'
WORKING_DIR = '/[base_directory_containing_decompressed_files]'
# specific file type subfolders
MSC_TMP = '/msc_temp'; NGPRS_TMP = '/ngprs_temp'
# set Python's structure automatically handles duplicates
c_set = set(); desd_c = set(); miss_cs = set()

def main():

    # Values initialization
    n_proc_f = 0; tot_n_of_rec = 0;tot_n_of_inc_rec = 0

    # loads the list of phone cells from the datast
    with open(h_cells_f,'r') as hcells:

        cells = list(csv.reader(hcells,delimiter=";"))

        #creates the cell_set from the .csv file
        for cell in cells:
            c_set.add(cell[1])

    # file list initalization
    file_list = []

    # Generates the list of all files to be tested
    for d in [MSC_TMP,NGPRS_TMP]:
        for f in os.listdir(WORKING_DIR+d):
            file_list.append(WORKING_DIR + d + "/" + f)

    # For every file
    for f in file_list:

        with open(f,'r') as fil:
            # initialize the file reader
            reader = csv.reader(fil, delimiter=';')

            # for every entry of the file
            for record in reader:
                #increase the counters
                tot_n_of_rec+=1

                #if the encountered cell is not listed in our dataset
                if record[8] not in c_set:
                    # adds it to the proper set()
                    miss_cs.add(record[8])
                    tot_n_of_inc_rec+=1

                # set of all descvered cells
                desd_c.add(record[8])
        del(f)
        n_proc_f+=1

    # prints the statistics
    print ("Number of file processed: " + str(n_proc_f))
    print ("Number of records:" + str(tot_n_of_rec))
    print ("Number of Inconsistant Records: "+ str(tot_n_of_inc_rec))
    print ("Number of listed cells:" + str(len(c_set)))
    print ("Number of missing cells:" + str(len(miss_cs)))
    print ("Number of descovered cells:" + str(len(desd_c)))
    print ("Number of phantom cells:" + str(len(c_set)-len(c_set & desd_c)))
    # save the result
    np.save('c_test_res.npz',[n_proc_f,tot_n_of_rec,miss_cs,c_set,desd_c,
    tot_n_of_inc_rec])

# main function
if __name__=='__main__':
    main()

# EOF
```

Code Snippet 3.2: Cells Consistency Test

3.2.3 Data Cleaning and Aggregation

After the decompression and the inconsistency test phase we started working on removing useless records, aggregating and finally reshaping the data (both for the MSC and NGPRS data logs) trying to reduce the information at the minimum in order to make it properly fit with our specific project-related needs. We decided to do so both for matters of commodity and for performances reasons.

After this phase the files contained not only the unique identifier of the cell tower but also the global position coordinates, this procedure avoided us to consult the cell.csv file every time we analyze a record, thus, increasing time performances when working with time series.

To do so for utility reasons we split this task in several steps. First of all a python module called *cleaner_plus_gps* have been wrote (Code Snippet 3.3). It contains a function called *csv_handler* the scope of the function is to filter the file attributes based on the needs and then add the gps coordinates of the cell tower based on the tower id attribute. *csv_handler(...)* takes exactly four parameters as input:

1. The path of the input file
2. The separator char of the .csv input file (optional)
3. An array containing the csv columns to keep on the output file
4. The name of the output file (optional)

The function *csv_handler* is then used on *parallelize.py* (Code Snippet 3.4) which spawn parallel executors to perform the cleaning operation in a parallel fashion.
At the end of the procedure we got files organized still on a daily basis but containing just four records: the IMSI of the subscriber, the date-time of the event (without details about its nature) and finally the longitude and latitude of the cell tower related to the event. Given the inconsistency of the dataset analyzed with the consistency test, in case the cell tower identifier of the event log is not contained into the cells file then the log is just discarded.

```
1 #!/usr/bin/python3
2 # FILENAME: cleaner_plus_gps.py
3
4 import sys, csv
5
6 h_cells_f="/[path_to_file]/cells.csv"
```

```
7
8  def csv_handler(filepath, separator=",", attrs, out_file_path=None):
9      # python dictionary to fill with tower data
10     tower_dic = {}
11
12     # open the cell file in read-only mode (allows concurrency)
13     with open(h_cells_f,'r') as hcells:
14         # reads thw full cells.csv file
15         cells = list(csv.reader(hcells,delimiter=";"))
16         # fills the dictionary with the cell towers data
17         for cell in cells:
18             tower_dic[cell[1]] = [cell[2],cell[3]]
19
20     # open the input file file in read-only mode
21     with open(filepath,'r') as in_csv_file:
22         # csv reader initilization
23         reader = csv.reader(in_csv_file, delimiter=separator)
24
25         # handle the missing output file parameter
26         if not out_file_path:
27             out_file_path = filepath + "_clean"
28
29         # open the output file in write mode
30         with open(out_file_path,'w') as out_csv_file:
31             # csv writer initialization
32             writer = csv.writer(out_csv_file, delimiter=';')
33
34             # For every line add the cell tower coordinates and
35             # finally keeps only the arguments passed as "attrs" parameter
36             for record in reader:
37                 try:
38                     writer.writerow([record[i-1] for i in attrs] +
39                         tower_dic[record[-1]])
40                 except KeyError:
41                     # Discards inconsistent cells...
42                     # Consisitency test provides details about the statistics
43                     pass
44 # EOF
```

Code Snippet 3.3: CSV Filtering plus GPS

```
1  #!/usr/bin/python3
2  # FILENAME: parallelize.py
3  import os
4  # import the function to be parallelized
5  from cleaner_plus_gps import csv_handler
6  # import python's multiprocessing thread pool handler
7  from multiprocessing import Pool
8
9  # some string variables initialization (paths and suffix)
10 WORKING_DIR = '/[path_to_the_root_working_dir]/cutted'
11 MSC_TMP = '/msc_temp'; NGPRS_TMP = '/ngprs_temp'; MSC = '/msc'
12 NGPRS = '/ngprs'; SUFFIX = '_cutted_gps.csv'
13 DIR_LIST = [[MSC_TMP,MSC],[NGPRS_TMP,NGPRS]]
14 def main():
15     # creates a pool composed of 10 executors
16     pool = Pool(10)
17     # file list initialization
18     file_list = []
19     # creates the list of files to be processed
20     for inner in DIR_LIST:
21         for f in os.listdir(WORKING_DIR+inner[0]):
22             #for both in inner:
23             file_list.append([WORKING_DIR+inner[0]+'/'+f,WORKING_DIR+inner[1]+'/'+f
       +SUFFIX])
24     # creates the list of parameters for the executors
25     parameters = [[f[0],';',[2,8,9],f[1]] for f in file_list]
26     # passes the function to be executed by the 10 executors
27     # and the list of parameters that the csv_handler takes
28     pool.starmap(csv_handler, parameters)
29
30 if __name__ == "__main__":
31     main()
32
33 # EOF
```

Code Snippet 3.4: Parallelization Python Handler

3.2.4 Data Filtering and Reshaping

At this point we have minimized the dataset removing all the irrelevant information from it. But since we are going to analyze the activity from the cells perspective we decided to aggregate the data still on a daily basis but divided by cells locations. After this phase every cell will have a file describing its daily activity. In fact, our files before the filtering task are just divided in a daily basis but containing all the logs of all the cell towers displaced in Hungary. Performing this reshaping we will get many more files (one for each cell for each day of the month) but smaller and better organized, leading to a more agile analysis.

In Code Snippet 3.5 we select the cells locations and the interested dates then the script will filter and sort the event logs according to our needs. Only after this the *ts_ min_ aggregator.sh* bash script (Code Snippet 3.6) aggregates and counts the number of logs with a resolution of seconds. Here by we used many powerful features of bash such as the *pipe function* (|) that connects the standard output of a command with the standard input of another, the *grep* utility is used to filter the logs based on the cell, the *sort* command is used to sort the logs in ascending order based on time while *uniq -c* have been used to count the number of logs for each second. Here the *awk* utility is used to filter and format inputs and outputs. Also in this case the NGPRS and MSC data have been kept separated for utility reasons.

```
#!/bin/bash
# FILENAME: logs_cells_filetering_plus_ts.sh

# base path initialization
BASE_DIR='/[working_directory_path]/cutted'
DEST_DIR="/[output_dir_path]/cutted/20170927_M1"
# array of interested dates
DATES=('20160920' '20160927' '20161004' '20161010')
# array of interested locations
cells_coord_1=(
    '47.6511212317;17.9824818709' '47.6763226127;18.0341952315'
    '47.6734745202;18.0960044913' '47.6728137899;18.1777190948'
)

for day in ${DATES[@]}; do

  SUB_DIRS=(
      '/msc/msc_T.out.'$day'_cutted_gps.csv'
      '/ngprs/ngprs_T.out.'$day'_cutted_gps.csv'
  )
  SUFFIX='_'$day'.csv'

  for sdir in "${SUB_DIRS[@]}"; do
    for suff in "${cells_coord_1[@]}"; do
    # Filtering Phase
    cat $BASE_DIR$sdir | grep $suff >> "$DEST_DIR/$suff$SUFFIX"
    # Aggregation Phase
    ./ts_min_aggregator.sh "$DEST_DIR/$suff$SUFFIX" "$DEST_DIR/ts/ts_$suff$SUFFIX"

    done
  done
done

# EOF
```

Code Snippet 3.5: Cell and Date Filtering

```
#!/bin/bash
# FILE: ts_min_aggregator.sh

# awk select columns | sorts | counts the number of unique occurrences
awk -F';' '{print substr($2,0,20)}' $1 | sort | uniq -c |
        awk '{printf("%s %s%s\n",$2,$3";",$1)}END{print}' | head -n -1 >> "$2"
        # output formatting

#EOF
```

Code Snippet 3.6: Time Series Generation

Finally we used the *sed* command to add an header to our csv files (Code Snippet 3.7).

```
#!/bin/bash
# FILE: sed_is_sad.sh

# -i: in place
# 1s: apply only to the first line
# * : Bash wild card (applies for every file in the folder)

sed -i '1s/^/DateTime;Value\n/' *

#EOF
```

Code Snippet 3.7: Adding a proper CSV header

At the end of the process the data looks like the following sample:

```
# FILE: ts_47.6460880566;17.9184636369_20160915.csv
# NAME CONVENTION: ts_{LATITUDE};{LONGITUDE}_{YYYYMMDD}.csv


DateTime;Value
2016-09-15 00:00:00;9
2016-09-15 00:00:01;27
2016-09-15 00:00:02;1
2016-09-15 00:00:06;2
2016-09-15 00:00:38;1
2016-09-15 00:05:02;2
```

As we can infer from the file example if the number of logs for a given time-instant is equal to zero then no entries at all will appear whithin the file. Since time series are defined for continuous values and fixed sampling rates to overcome this issue we use a specific Python's Pandas function at loading time called *fillna(val)*, in this specific case if any time related value is missing we fill it with an entry containing zero (Code Snippet 3.8).

```
df = pd.read_csv(path,sep=';', parse_dates=['DateTime'],
                         date_parser=dateparser, index_col=[0])
time_serie = df.fillna(0.0)
#EOF
```

Code Snippet 3.8: Filling Missing Values

Chapter 4

The Actual Literature

Lately the interest of the academic research over mobile telecom logs datasets has reached a lot of hipe and it became a topic itself. Some scientists focused their research studying the topological properties of the social network generated by the analysis of the mobile calls between users. When the localization information of each node were available, these network became geographical social networks, in this case the relationship between distance and the structure of the social network have been analyzed [10]. Finally given the dynamic aspect of the human relationships this networks have been studied as evolving temporal networks, in this case scientists tried to discover the relation between frequency of calls, with the distance and the time lasting of the relationship [8].

Leaving back the sociological nature of the research, this data also opens a huge number of other potential applications, which gives to this kind information an inestimable value, for instance telecom datasets could be used to extract valuable insights on geo-localized advertising campaigns. Even more mobile handsets are a way to analyze the pulse of cities, the trend shows that more and more cities are make development plans making use of information gathered from mobile telecom operators. In this framework, recent research has shown that mobile phone data could estimate where people are and where people use to travel, sometimes also the purpose of the trip can be estimated [6].

Provided this brief introduction to the actual wide literature material, in this chapter we are going to focus on the specific topic related to telecom dataset and mobility deepening in the mobility anomaly detection literature following a pure chronological approach.

4.1 Mobile Telecoms and Mobility

Back in 2001, the research was already well close to solve the problem of mobility flaw real-time monitoring on highways, in fact experimental results were promising. As described on the research paper [9] the STRIP project experiments were based on data in collaboration with a french telecom operator. The positioning information were collected exploiting the GSM protocol specifications without impacting on the actual network infrastructure. In fact, among the several standardized messages of the protocol carried by the signalling, some contain measurement reports; exploiting the information about these measurements the STRIP project were able to estimate the position of the 20% of the total car number in a 120km highway slice with a location estimation carrying a small error, namely, less than 500m. The aim of the paper was to demonstrate that with the STRIP project locating mobiles and thus cars can be realized without extra or specific devices other than normal mobile phones leading to a possible huge money saving in infrastructures, thus, opening new challenges to this research field.

In 2001 [4] highlights the issues of active monitoring versus passive motoring making a comparison between GSM and UMTS networks. The research pointed out that the newer UMTS technology could be better exploited to estimate the location and the speed of individuals carrying handsets. The paper also highlighted the fact that passive techniques are more interesting over active techniques for real applications scenarios. The main reasons in support of the passive localization techniques are: they do not produce any extra payload on the telecommunication network thus without having an impact on the quality of service and last but not least on the other side they do not have any impact on the energy consumption of the mobile handsets. This contribution also highlighted that newer technologies could help to increase the resolution of future mobility traffic monitoring systems. In [1] the authors tried to gather two different data sources in order to get a more accurate snapshot of the highways traffic statistics. Specifically, two discrete data sources involved; cellular network traffic data and road traffic data. These datasets were properly combined in order to create an unique dataset which correlates the operational conditions of the cellular network with the conditions on the road network. In this research the key variables of each data source are cell-to-cell handovers and average road speed. The cellular data used in this paper originate from the real-world network of a major mobile service provider in Greece. The selection of the cells was made so that antennas pointed as accurately as possible towards the adjacent highways. This way, the vast majority of cellular network subscribers associated with these particular cells at any time

are most probably driving on the motorway. After this pre-processing phase two different neural networks techniques were applied. The study showed that general regression neural network were the best approach over Multy-Layer Perceptron Neural Networks. The research concluded with interesting findings. Despite the good results the proposed solution has some real-word application-related issues, in fact, still, it needs fixed monitoring system to be trained, thus leading to the need investments on the highways infrastructures.

In [7] the authors highlight that the main limitation of the existing approaches to traffic estimation is the lack of a model taking explicitly into account of the mobility and transportation infrastructures. The estimates are often based on purely statistical correlation approaches which usually assume users movement directions following a uniform probability distribution. On the other hand, physical and normative constraints to user mobility inside a cell (e.g. as roads topology, mandatory directions etc.) are usually not taken into account in those models, with few exceptions. While relationships with traffic domain and external events, such as social events and social processes (e.g. work/home commuting, shopping periods etc.) are completely ignored. Moreover the authors claim that some issues such as privacy and scalability are also problematic in real deployment scenarios. For instance, techniques to infer Origin/Destination matrices use information about the location areas over the time, where a location area is a set of cells where the mobile terminal is assumed to be located. In few words the algorithm needs to identify time, origin and destination areas of the whole trip made by each single telephone, thus representing a remarkable privacy infringement issue. Mobile device localization detect the spatial position of the single user, by using techniques based on distance from the cell antenna, or assuming the placement of special detector-antennas for enhancing the accuracy of the localization. Although the remarkable precision is obtained at the expenses of investments, in both cases there are relevant problems of privacy and scalability. In fact, due to the huge amount of data generated by monitoring, each single terminal position in a cell would require an enormous bandwidth, storage and computational costs. The authors in their work propose a model which integrates spatial networks with mobile phone networks, in order to monitor, analyze and predict the user traffic on the mobility infrastructure and to make detection and inference about social events and processes in place, on the basis of anonymous aggregated data. The aim of the research in this case is that by integrating mobility constraints (e.g. available roads), it is possible to improve the accuracy of predictions coming from the cellular network based on the mobility network and vice versa. Moreover they highlighted that social event or processes which take place can also be detected, and conversely the knowledge of those dynamics can improve the predictive

model in the mobility domain. The dataset was mostly based on data series describing the ”handover” of anonymous users, i.e. the number of users which traverses any of the six boundaries of an hexagonal cell in a mobile phone network. The authors justify the choice of handover data usage over individual location estimation mostly for two different reasons: firstly privacy issues, anonymized handovers can easily be made available and can be securely and effectively transmitted while the continuous position tracking of a single subscriber's terminal would represent sensitive data about the individual user behavior and secondly has the drawback of big impact on performances and scalability. While in the proposed solution the size of the information to process remains mostly constant as the number of users increase.

The paper concluded with encouraging results demonstrating that traffic flaw control and the anomalous events detection can be performed preserving the total respect of user's privacy and in the mean time keeping at minimum the computational resources needed.

In 2013 in [5] a passive model composed by two main components, location areas updates and cells handover. Their approach leverages the signaling data observable in the mobile cellular network to infer road traffic status in real-time. While almost all previous studies on the topic have considered only mobile phone data related to “active” terminals, such as terminals with an ongoing voice call or data transfer, they contribute with a method that exploits the more complete signaling data captured from the network links near the Radio Access Network (RAN) of the cellular network. In this way, also the position of “idle” terminals can be observed, though leading to an increased spatial granularity, i.e., Location Area (LA) level (smaller) instead of cell level (wider). Since “idle” terminals are the majority of the mobile terminal population, this approach allows them to reach much better coverage. To benefit both from the large set of idle terminals and the more accurate positions of active users, they propose a two-stage approach. The fist one based on location areas updates tracks moving individuals estimating the average travel speed. While the second stage aims to detect active users having a phone call to better localize the position of the event. Experiments on a major traffic route in Austria showed that the presented system is able to detect road incidents reliably and in a timely manner.

Conclusions

This short literature analysis the technology has wide potential although more research is needed in order to use it as efficiently as possible. Researchers have shown good results on highways while promising results have been shown in arterial and even urban environ-

ments, but, the results in these environments are quite few and very varying.

Effective systems are using multiple type of information from data communications and heterogeneous sources, rather than just telephone calls because the effectiveness of the methods result considerably increased. The shift towards UMTS affected the performance in these systems, in fact using also UMTS data (given the smaller signal coverage areas) for travel time estimation increased the number of "sensors" in the system, which is an important factor when it comes to both travel time accuracy and accidents detection. Increasing the number of data samples in a travel time reporting interval is fundamental for accuracy, especially in this kind of scenarios where the system generates noisy measurements. The higher data rate in UMTS makes the network react much faster to changes helping the detection of the anomalies. This, in combination with the soft handover principle that makes a radio link addition or removal, might be the reason to the much better UMTS location accuracy. This is independent of whether handovers or something else is used to determine specific locations of the mobile terminal on the road. The higher location accuracy of the UMTS network can be used to make the travel time accuracy more precise. Shorter travel time segments are necessary in urban environments and are also useful when detecting accidents.
It is likely that hybrid approach solutions like in [5] will outperform the other methods if a good way of selecting the right cell towers to monitor is developed. The UMTS network utilizing event triggered measurement gives a great improvement to the hybrid monitoring approaches, anyway, the location data available from a cellular network depends extremely on the configuration of the network. Most vehicles today have at least one mobile phone on board and a lot of rich data is possible to extract from the cellular networks. Passive approaches are the always favourite in this scenario because of their non-invasive nature and their high easy to scale nature.

Anyway the perspective of these approaches are always creating a model based on the speed and flow of the cars and not in the load of the cells.

Chapter 5

Our Approaches

5.1 General Overview

5.1.1 Preliminary Considerations

Since our aim is to spot anomalies on highways we decided to address this research issue with a pretty different approach. Given the fact that usually highways infrastructure are deployed mostly in rural areas (at least in Hungary it is an evidence), as a consequence of this, the cells tower infrastructures are deployed mostly close by highways and the activity of the cells is mostly determined just by travellers and their interaction with handsets. Given the low noise level generated by static users our assumption is that it is possible to infer anomalous events just spotting relevant fluctuations on the activity of these specific cells without inferring the number of travellers or creating a model based on individual trajectories, but just monitoring the aggregated number of events over the time.

Assuming that services used by people in daily life (e.g.: the cellular and road networks in this specific scenario) are not completely uncorrelated, but rather they are affected one by another, these assumption can drive us to the conclusion that information about one system (highways in this case) can be deduced from data provided by other systems (cellular network activity). The study of the exact relationships among these *"closely related"* systems can provide a valuable information about how to make use of data to make a reasonable analysis regarding one system by using data produced from the other system.

One trivial example that better describe this correlation concept is suddenly noticing a meaningful increase in outgoing calls or data traffic from a specific cellular tower that is near an highway and leading to the conclusion that most probably there is some sort of

traffic related anomaly in progress, as that would be a major reason for the drivers to stop driving and start using their mobile phones to communicate delays or just to kill the time.

5.1.2 Dividing the Space

The spatial localization of the cells activity can be inferred by generating graphs sub-divided in Voronoi cells, which is a good approximation model to delimit the area of influence of each cell tower or antenna. The Voronoi diagram sub-divides the area into polygonal regions, associating each region with one tower.

The partitioning method is such that all points within a given Voronoi cell are closer to its corresponding tower than to any other tower in that area. This model helped us in the task to select the cell towers that most probably will reflect the relation between the two related systems.

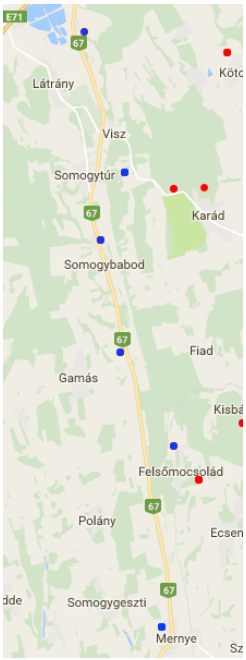


Figure 5.1: The cells displacement reflects the highway infrastructure

Figure 5.1 confirms with evidence our preliminary considerations that in rural areas, or at least on low population density areas, cell towers are strategically displaced following the infrastructure of the transportation system. In this case the blue points represents the cells on which their activity is expected to be highly correlated with the activity of the highway; while the red dots represent cells that in this case are considered not relevant for

our research.

On the other hand Figure 5.2 illustrate the Voronoy diagram related to the map of Figure 5.1, as we can see the polygonal structure of the diagram helps us to better spot the highway-related cells. As we can observe in this case the blue dots are representing the cell towers whose Voronoy polygons contain the highway within their area. The red dots represents the mobile cells whose activity are no expected to be affected by the activity of the highway thus their polygons do not cross with the drawn highway in black.

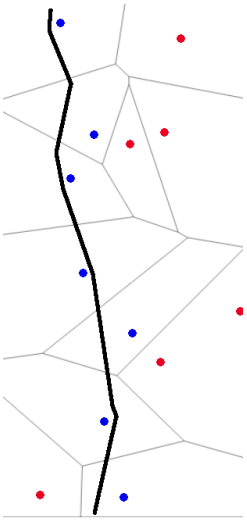


Figure 5.2: The related Voronoy diagram used to define which cells to select

5.1.3 Time Series

Given the nature of the data the best representation to spot anomalies resulted to be time series. Before describing our solution there is the need to introduce some formal notations and theoretical findings about time series.

A time series is often the result of the observation of a process in the course of which values are collected from measurements made at uniformly spaced time instants and according to a given sampling rate. A time series can thus be defined as a set of contiguous time instants. Time series can be univariate or multivariate when several series simultaneously contain multiple measurements within the same time range. Time series in data analysis stems from the desire to emulate the human ability to visualize the shape of data. Humans rely on complex schemes in order to perform such tasks. We can actually avoid focusing on small fluctuations in order to derive a notion of shape and identify almost instantly similarities

between patterns on various time scales. Major time-series-related tasks include query by content, anomaly detection [11], motif discovery, prediction, clustering, classification, and segmentation.
A series is said **continuous** when observations are made continuously in time (the term continuous is used also when the measured variable takes only a discrete set of values).

On the other hand a time series is said **discrete** when observations are taken only at specific times, in general equally spaced (the term discrete is used for this kind of series even when the measured variable is a continuous one). In this specific scenario we will use just discrete time series.

Given a continuous time series we can read off values in (equally distributed) intervals of time to produce a discrete time series that is a sampled time series (for example, humidity in the air measured every 7 minutes). In this case the time series are called also flows. Suppose a variable does not have an instantaneous values, but the values can be aggregated over equal intervals of time. In this case the series (especially in the economic field) are called stocks.

Another important feature of time series should be mentioned. Time series analysis has the characteristic of not assuming that the sample consists of independent and identically distributed samples (typical of statistical theory). In fact, temporal dependence is probably the most important intrinsic feature of time series data and the analysis of time series have to be made taking into account the time order of the observations. Indeed, when successive observations are dependent, future values can be somehow predicted by past observations.

Time Series Analysis

The main steps over time series analysis are:

1. **Description:** the first step in the analysis is the plot of the data and obtain simple measures to have a look at the main properties of the series. Although much more sophisticated techniques are often used, the analysis of the graph must not be neglected since it can reveal the existence of trend, seasonality, anomalies, motifs or turning points.

2. **Explanation:** this means identifying the random mechanism that generates the phe-

nomenon of which a sequence of observations are available. In case observations are taken on two or more variables, it is possible to explain the variation in one time series on the basis of the variation of another time series or just compute their correlation.

3. **Prediction:** given observed time series a typical task is to predict future values. This is based on the principle that the behaviour of the phenomenon in the past is maintained in the future with a certain regularity.

Time Series Components

Every time series can be subdivided in components:

1. **Trend:** can be defined as long term change in the mean of the entire series. The meaning of "long term" is quite ambiguous definition. A definition can be that a trend in mean could be defined as comprising all cyclic components whose wave length exceeds the length of the observed time series.

2. **Seasonality:** These are short term movements occurring in a data due to seasonal factors. The short term is generally considered as a period in which changes occur in a time series with variations in weather or festivities. A seasonal behavior is strictly regular, meaning there is a precise amount of time between the peaks and troughs of the data. For instance temperature observations over the year would have a seasonal behavior.

3. **Cycles:** these are variations exhibited by the time series at fixed period that cannot be considered as seasonality. The length of the cycle might be not fixed but may vary in terms of duration over time.

4. *Motifs:* are subsequences of a longer time series, which are very similar to each other

5. **Irregular fluctuations:** after trend, cycle and seasonal variation have been modelled during the time series analysis something might be left in the residuals. irregular

fluctuations can not be inferred from the past history. Making sure that the residual component is random is an index that the decomposition of the series into the described above components is correct.

Useful Definitions

5.1.1. Definiton. *A time-series TS is an ordered sequence of n real-valued variables:*

$$TS = (t_0, \ldots, t_n) \text{ where } t_i \in \mathbb{R} \text{ and } i \in \mathbb{N}$$

5.1.2. Definiton. *Given a time series $TS = (t_0, \ldots, t_n)$ of length n, a subsequence S of TS is a series of length $m \leq n$ consisting of contiguous time instants from a fixed time instant T whose index is denoted by k.*

$$S = (t_k, t_{k+1}, \ldots, t_{k+m}), \text{ where } 0 \leq k \leq n - m$$

5.1.3. Definiton. *Given a time series TS of lenght n, and a fixed value r with $r \leq n$, an r sum aggregated time series of TS denoted as $\overline{TS}$ will be:*

given $TS = (t_0, \ldots, t_n)$ where $t_i \in \mathbb{R}$ and $i \in \mathbb{N}$, then $\overline{TS} = (t'_0, \ldots, t'_m)$

$$\text{where } t'_i = \sum_{j=ri}^{r(i+1)-1} t_j \text{ with } m = \left\lfloor \frac{n}{r} \right\rfloor \text{ and } 0 \leq i \leq m$$

Definition 5.1.3 is important to understand the nature of the data. In fact, as mentioned, the dataset logs were stored with a resolution in time of seconds, given the high level of fluctuations, during the visualization process the resolution of the time series have been reduced using the *r sum aggregation* with a variable *r*.

5.1.4 The Nature of our DataSet

In this section we are going to dive into the visualization of the data, from this fundamental process we observed the characteristics of the generated time series. The result of this analysis justifies the roots of both our solutions.

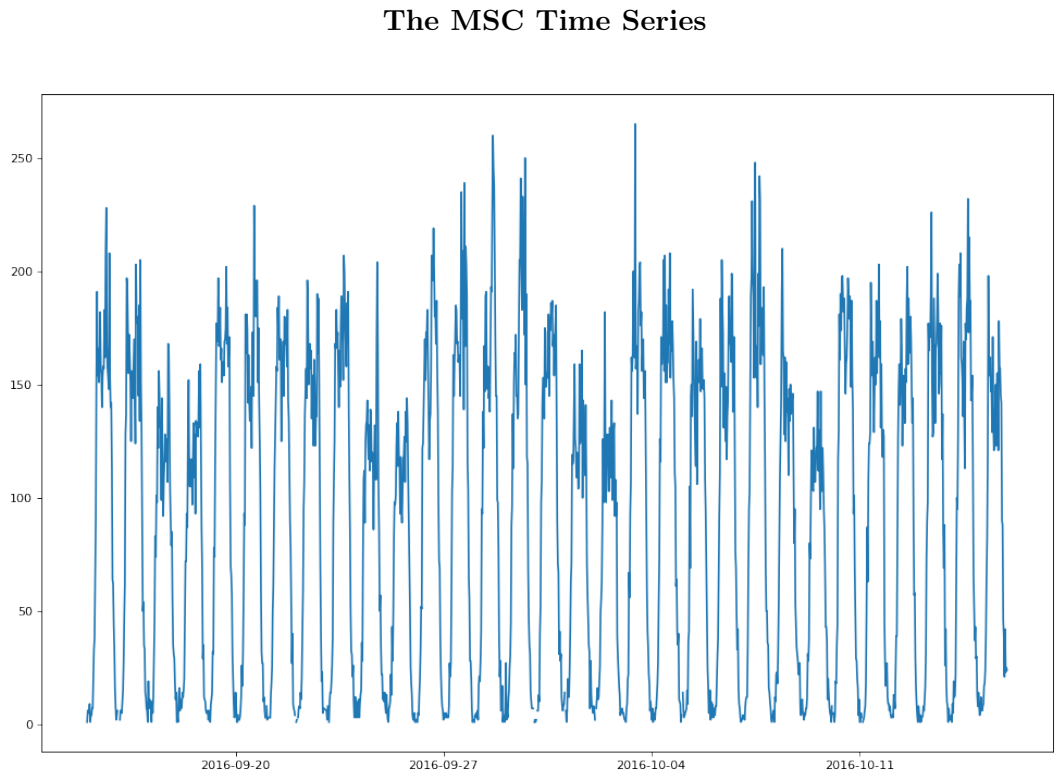


Figure 5.3: Montly activity of a sample cell based on msc data

Figure 5.3 shows the time series generated by the activity of a sample cell based on msc data files. The cell activity represented here in this specific context refers to a specific cell positioned close by highway M1 in Hungary. The time window of the observations is one month wide, exactly between the 15th of September and the 15th of October 2017, as far as we now at the moment we are writing, no anomalies should be happened on that area in that specific time window.

As we can infer from the graph no trend or seasonality have been spotted since the period of the observations is just one month however daily motifs are present.
The aggregated series is derived by starting from the original series deriving an r sum aggregated series with r equal to 1800, in few words the new sampling is equal to 30 minutes instead of 1 second resolution.

For the aggregated data corresponding to a 30 minutes interval on this sample cell we can observe that the graph has a certain regularity and it kind of reflects the normal human activity: low level during the night and early morning and peaks close to nine am in the

morning and between two or three pm in the afternoon. From this graph we can even infer working days and weekends days since the holidays (or weekends) are characterized by having a eye-noticeable lower activity on the graph.

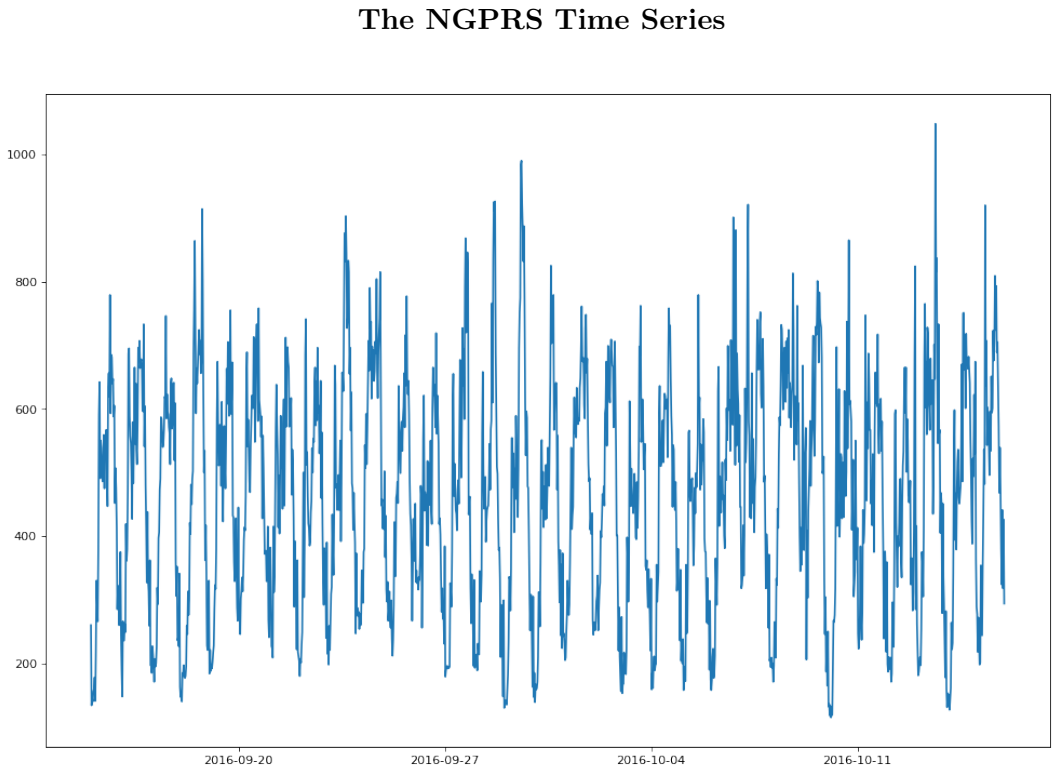


Figure 5.4: Montly activity of a sample cell based on ngprs data

On the other hand, similarly to Figure 5.3, Figure 5.4 shows the time series generated by the activity of the same sample cell based on ngprs data files.
Also here the time windows width is the same (one month) with the same dates.
Also in this case no trends or seasonality are evident and motifs are harder to spot. Even tough the same sampling rate and the aggregation parameters are the same in this case the time series looks pretty different.

In this case the number of events per sampling rate if compared with Figure 5.3 is pretty higher with peaks around a thousand of event (opposed to 250 on the other graph), thus it looks like the different nature and usage of the gprs data is reflected to this graph.
In fact, as opposed to the conventional calls or SMS the data usage is not enabled just

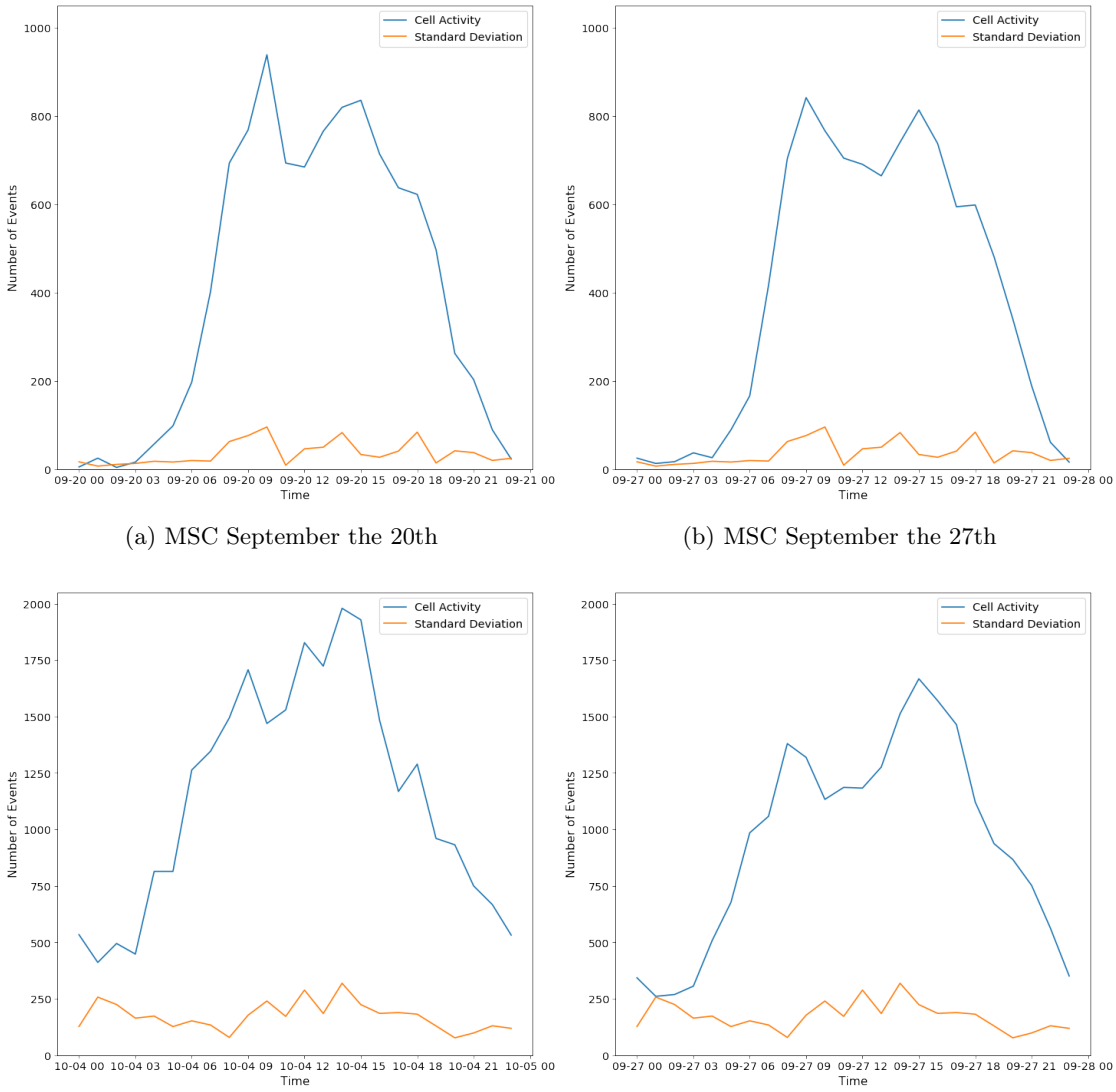
while a person is communicating with another individual (thus actively communicating information).

Data, besides texting through over the top applications, can be used to check the news (one to one activity) or even more several automated processes can act as triggers enabling network data events such as notifications or application specific logs, thus, sometimes individuals have their handset interacting with the network through data even tough they are not interacting with the device. *Besides the apparently noisy nature of the NGPRS activity this characteristic can be exploited for monitoring both the active and passive network events enriching our sensor network and providing a more continuous and rich overview of state of the highways.*

Looking further to the time series we can see that the normal human calling behaviours that respects human biological rhythms can not be clearly spotted, in fact, peaks are not so regular and the daily activity can not be well distinguished, at first glance every day looks similar and at the same time different from all the others.

In this specific time series random noise or irregular fluctuations appear to be dominant. Here days are not similar the one to the other, but an exception can be made in the case the day of the week is the same. Weekends or holidays appear to be not clearly distinguishable if compared to MSC data.

Despite all this noise we believe that anomalous activities can be notices also from this kind of data with an increased overall usage of the network.



(a) MSC September the 20th

(b) MSC September the 27th

(c) NGPRS September the 20th

(d) NGPRS September the 27th

Figure 5.5: Detailed NPRS and MSC comparison

In Figure 5.5 we give a more detailed comparison among MSC daily data (a and b) and NGPRS daily data (c and d) for the same days of the week. As we can note, the fluctuations of the MSC data are less noticeable if compared to the NGPRS ones. In this case to smooth the graph the sampling rate have been reduced to 60 minutes. The NGPRS data has an higher variance in every point of the time series.

The (a) and (b) graphs so reflect the intrinsic rhythm and pulse of the highway cupled with the normal human calling behaviours. In a more detailed fashion: we can expect call

patterns during peak hours to be dominated by the traffic activity. Besides different spatial cell activities depending on the selected area, on graphs (a) and (b) an evidence inferred from the dataset is that each particular time of the day, as well as each day of the week, is dominated by a different level of activity, but this level of activity (as shown in the graph) is similar for the same day along different weeks, in fact, Mondays are more similar to other Mondays if compared with the other days of the week. Unfortunately this evidence can not be inferred on the NGPRS dataset.

At the expense of some loss of time resolution, aggregating data into larger temporal bins, thus taking the sampling temporal ratio lower, allows for better statistics and for a more stable activity pattern. In fact the number of calls made from a specific cell at a certain time and day of the week is expected to be mostly constant with a bit of shifting. This means that made exceptions for small statistical fluctuations. Usually, activity patterns are strongly correlated with the daily pulse of a specific area and, at a wider spatial area, to variations in population density between different regions within the country.

Provided these observations about the dataset, observed activities values far from the mean, are in general not just simply correlated with the number of travellers. Instead they could reflect anomalous, or even better, unexpected events strictly correlated with the status of the mobility infrastructure.

Measurements of statistical fluctuations around the mean of the expected activity looks relevant, since it enables the possibility to perform a quantitative analysis of anomalous events and, ultimately, of possible emergency situations such as car accidents. This indeed constitutes the roots of a real-time monitoring framework to help infrastructure mainteiners and institutions to keep track of the current situation based on the normal activity and its fluctuations.

5.2 The Models

5.2.1 The Standard Deviation Model

Given the characteristics of the dataset we found in literature a good model able to dynamically estimate anomalous activities based on the mean of the normal activity of the cells and its fluctuations. The method is able to spot anomalies making use of the standard deviation and a threshold value to be fine tuned [3].

5.2.1. Definiton. *To get the weekly period of the observed events, we can define $n_i(l,t,T)$ as the number of calls recorded at location l covered by a specific cell tower during the i_{th} week between times t and $t+T$, where time is defined modulo 1 week where t and T depends on the observations time ratio. Assuming we have information records for N weeks, the* ***mean cell activity is defined as****:*

$$\bar{n}(l,t,T) = \frac{1}{N}\sum_{i=1}^{N} n_i(l,t,T)$$

The value of T depends on which aggregation values preserves a stable behaviours.In our case we will test with different values in order to get the lowest value possible.
In order to model the normal fluctuations in normal conditions the sample standard deviation represents a good instrument.

5.2.2. Definiton. *Given the previous definition the* ***sample standard deviation*** *can be defined as follows:*

$$\sigma(l,t,T) = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(n_i(l,t,T) - \bar{n}(l,t,T))^2}$$

This way, using recorded data for a long period of time,it is possible to determine the expected traffic levels and corresponding deviations for all times and locations. Once the normal activity is learned coupled with its standard deviation. Given a fixed threshold A_{thr}, an anomalous activity between the time instant t and $t+T$ is detected when the following condition is satisfied:

$$n_i(l,t,T) > A_{thr} \times (\sigma(l,t,T) + \bar{n}(l,t,T))$$

On Code Snipped 5.1 Follows the Python implementation code. The script first loads the time series and aggregates the activity in hourly basis using panda, then the mean activity is computed. Using the mean the standard deviation is computed as well, only then the

time series to be tested is iterated and the evaluation process takes place.

```
#!/usr/bin/python3
from operator import add
import pandas as pd

files = [
    'ts_47.6763226127;18.0341952315_20160920.csv',
    'ts_47.6763226127;18.0341952315_20160927.csv', # day of the accident
    'ts_47.6763226127;18.0341952315_20161004.csv',
    'ts_47.6763226127;18.0341952315_20161011.csv',
]

# threshold value
A_tr = 1.09
# aggregation parameter
aggregation_val = '60T'
# List initialization
past_activity = []

for i in range(0,len(paths)):
    # time series loading
    df = pd.read_csv(paths[i],sep=';', parse_dates=['DateTime'],
                                    date_parser=dateparser, index_col=[0]).fillna(0.0)
    # time series resampling / aggregation 1h
    series[i] = df.resample(aggregation_val).sum()
    # data reshaping
    past_activity.append([item for sublist in series[i].values for item in sublist
    ])

# removes from the training data the ts to be tested
ts_to_test = past_activity.pop(1)

#Given a list of lists compute the mean point by point
def average_vec(arr):
    # created an array of the same length of the values array
    res = [0]*(len(arr[0]))
    # iterates over the values
    for i in range(len(arr[0])):
        sm = 0 # sum initialized to zero
        # iterates over lists
        for j in arr:
            sm += j[i]
        res[i] = sm
        # divide by the number of elements and returns
    return [i/len(arr) for i in res]

#Given a list of lists compute std point by point
def std(mean_list, val_list):
    # created an array of the same length of the mean array
    res = [0]*(len(mean_list))
    # iterates over the values
    for i in range(len(mean_list)):
        sm = 0 # sum initialized to zero
        # sum, iterates over the lists
        for j in val_list:
            sm += (j[i] - mean_list[i])**2
        # the std is computed and then returned
        res[i] = (sm/(len(val_list)-1))**0.5
    return (res)


if __name__=="__main__":

mean_list = average_vec(past_activity)
std_list  = std(mean_list, past_activity)
upperbound = list(map(int,list(map(add,mean_list,std_list))))

for i in range(0,len(mean_list)):
    if ts_to_test[i] >= upperbound[i] * (A_tr):
        #anomaly detected prints out the index of the time series point
        print(i)
# EOF
```

Code Snippet 5.1: The Python Code of the Standard Deviation Model

5.2.2 The Statistical Approach

Since we wanted to get evaluation metrics to compare the results of the standard deviation approach with another technique of different nature, we opted for a totally different type of comparison. In this case the statistical approach we will introduce (as opposed to the A_{thr} of the previous model) has a big advantage: it does not need any parameter to tune, thus, this solution is expected to be easier to scale.

The statistical approach we decided to use is the so called Mann–Whitney–Wilcoxon (MWW) rank test, since it does not need any assumption on the nature of the distribution of the data to be compared it seems to be a good starting point for a statistical analysis of the time series.

In statistics, the Mann–Whitney U test is a non-parametric test of the null hypothesis that it is equally likely that a randomly selected value from one sample will be less than or greater than a randomly selected value from another sample. The MWW test does not need any previous assumption about the distribution of the values. From this test can be generated the Wilcoxon rank sum test that is a non-parametric test as well and it can be used to determine whether two independent samples were selected from populations having the same probability distribution.

In statistical theory, a U statistic is a class of statistics relevant in estimation theory where the letter "U" stands for unbiased.

Generally for the Wilcoxon rank sum test the two following steps are executed:

1. The *U statistics* is calculated from the two sets.
2. The resulting value is compared to the theoretical distribution of such U statistics, producing a probability that can be interpreted as a p-value.

A general formulation of the test needs some hypothesis:

- All the observations from every group are independent the one from the other
- The responses are ordinal (is possible to state of any two observations, which is the greater)

- Under the null hypothesis H_0, the probability that an observation from population A exceeding an observation from a second population B equals the probability of an observation from B exceeding an observation from A: $P(A > B) = P(B > A)$ or $P(A > B) + 0.5 \cdot P(A = B) = 0.5$. A stronger null hypothesis (the one we will use) is: "The distributions of the two populations are the same" which implies the previous hypothesis.
- The alternative hypothesis H_1 is the probability of an observation from population A exceeding an observation from the another population B is different from the probability of an observation from B exceeding an observation from A: $P(X > Y) \neq P(B > A)$. It can be viewed also as : $P(X > Y) > P(Y > X)$.

The computation of such a test can be performed as follows:

1. Assign a numeric rank to all the observations considering both sets, where the smallest observation has rank 1, the second smallest has rank 2, and so on.
2. Adding the ranks for the observations from each sample obtaining R_1 and R_2.
3. At this stage the U_i values can be computed as:

$$U_1 = R_2 - \frac{n_1(n_1 + 1)}{2}$$

   where n_1 is the size of the first sample, and R_2 is the total sum of the ranks of the second sample. The U_2 value can be computed analogously.
4. At this point the U value can be computed as: $min(U_1, U_2)$.
5. Finally the $p - value$ can be estimated from U by means of statistical tables for the Mann-Whitney U test to find the probability of observing a value of U or lower.

In our specific scenario we will compare two "sub-series" trying to infer if they come or not from the same probability distribution. The smaller the value of the *p-value* the higher probability that the two time series sub-series are not following the same probability distribution. Standard delimiter values of the *p-value* are 0.05 or 0.01

From our perspective if the compared sub-series will lead to a *p-value* lower than a specific value then the sub-series will result to be too different. Doing so with several of the observations in the past and calculating the average *p-value* can help us to determine if an

anomaly is occurring or not, or even better, we can determine if the activity recorded in a certain period is compliant with the previously observed normal activity of the cell or it is an outlier representing a probable anomaly.

Windows Overlapping

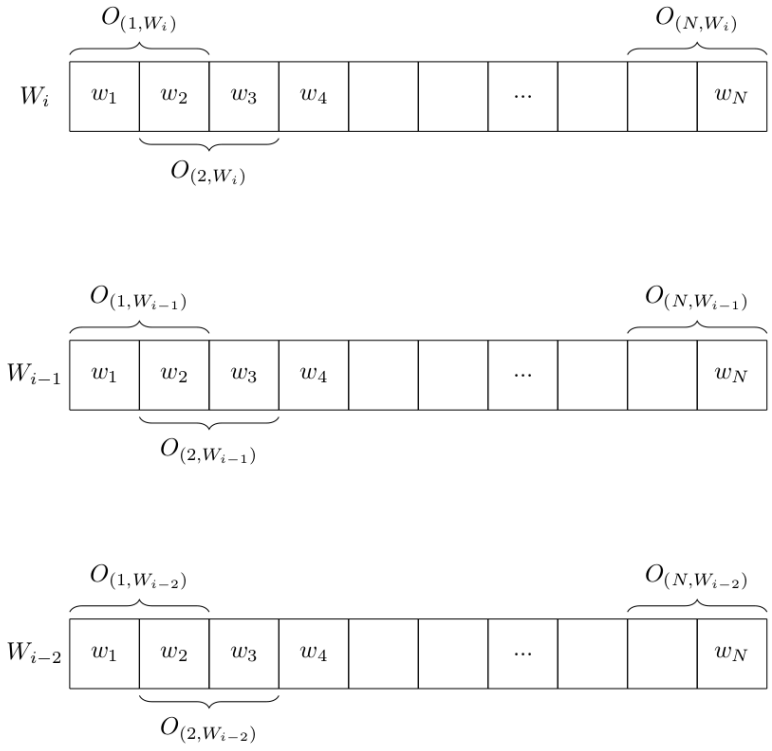


Figure 5.6: An Illustation of Overlapping Windows Sampling

Since our aim is to reduce at the minimum the time between an observation and the other keeping the stability and precision of the model we found that calculating the *p-value* for overlapping windows samples lead to better results. This improvement have been tested for two reasons: firstly the MWW test is more precise with a considerable number observations, secondly the overlapping windows sampling can help us to have a "more continuous" consideration of the time avoiding net cuts on the time series, thus, avoiding to cut any anomalous event in half of its escalation.

Figure 5.6 shows the concept expressed above. W_i represents a given day of the week to be tested, while W_{i-1} represents the same day one week before and so on...

Since we divided the day in smaller samples the w_i notation represents a time window, where the number of samples contained on a time window depend on the sampling rate and the width of the window. We decided to keep the sampling rate fixed to a resolution of one minute testing the algorithm with different windows sizes (15 and 30 min).
$O_{(k,W_j)}$ represents the observations on the samples contained in two windows, namely w_k and w_{k+1} for the j_{th} week. Finally we denote the Wilcoxon rank sum test providing the *p-value* as: $WMM(O_1, O_2)$.

5.2.3. Definiton. *Given a daily observation in the past i and a generic function f we define the* ***control function*** *denoted by θ, where $\theta : (\mathbb{N}, f(...)) \to \{0, f(...)\}$ such that:*

$$\theta(i, f(x)) = \begin{cases} 0 & \textit{if an anomaly happened in } i \\ f(x) & \textit{if no anomaly happened in } i \end{cases}$$

where $f(x)$ can be any weighting function.

Supposing we have data for M weeks in the past, denoting as x the day to be tested, $f(k)$ as a generic weighting function and a constant normalizing function and another $\gamma(i)$ function, the weighted global mean of the *p-value* for each O at the k time instant is calculated as follows:

$$\bar{p}_{val}(x, k) = \frac{1}{\sum_{i=1}^{M} \theta(i, \gamma(i))} \sum_{i=1}^{M} \theta(i, f(i)) \cdot WMM(O_{(k,W_i)}, O_{(k,W_x)})$$

If $\bar{p}_{val}(x, k)$ is smaller than 0.01 then our observation looks to be part of a different probability distribution with an high confidence, thus, the activity for that observation can be denoted as anomalous. The control function θ is used to avoid to consider past observations when anomalies occurred. In all our experiments we used $f(x) = 1$ and $\gamma = f$, however other weighting functions can be used e.g.: $f(x) = log(x)$.

In this specific context we decided to use the *SciPy* library, namely the *scipy.stats.mannwhitneyu(O_1, O_2)* function have been used.

Follows the Python implementation:

```
#!/usr/bin/python3
from matplotlib import pyplot as plt
import scipy.stats as sps
import pandas as pd
import numpy as np

base_path = "/mnt/disk2/agalloni/BigData/cutted/20170927_M1/ngprs_only/ts/"

files = [

    "47.6763226127;18.0341952315_20160920.csv",
    "47.6763226127;18.0341952315_20161004.csv",
    "47.6763226127;18.0341952315_20161011.csv",
    "47.6763226127;18.0341952315_20160927.csv", # the day of the accident

]
dateparser = lambda x: pd.datetime.strptime(x, '%Y-%m-%d %H:%M:%S')
paths = [ base_path +"ts_"+ x for x in files]

# lists initialization
series = [0]*(len(paths)); grouped = [0]*(len(paths))

#Alpha threshold value
alpha = 0.01,

# Parameter settings
window_w = 15; offset = 30

for i in range(0,len(paths)):
    # time series load from csv
    df = pd.read_csv(paths[i],sep=';', parse_dates=['DateTime'], date_parser=
    dateparser, index_col=[0])
    # resamplint 1m and null filling wuth 0.0
    series[i] = df.resample('1T').sum().fillna(0.0)
    # take raw values
    vals = [item for sublist in series[i].values for item in sublist]
    #reshapes the series based on offset and window width
    grouped[i] = [vals[x-offset:x+window_w] for x in range(offset, len(vals),
    window_w)]  #ok 2


# select the day to be tested
tbt = grouped.pop(3)

for i in range(0,len(grouped[0])):
    #mean accumulator initialization
    acc = 0

    for j in range(0,len(grouped)-1):
        # inidividual p-value
        p_val = sps.mannwhitneyu(grouped[j][i],tbt[i])[1]
        # mean accumulator
        acc += p_val
    # mean p-value
    avg = acc/(len(grouped))
    # condition test
    if (avg < alpha):
        print ('Anomaly Found {} AVG: {}'.format(i,avg))

# EOF
```

Code Snippet 5.2: WMM Statistical Test

5.2.3 Evaluation Metrics

Since we need some metrics to evaluate and compare the results provided by the models previously described in this section we will define the concepts of *precision* and *recall*.

In binary classification, precision is the number of positive results over all the generated results, while recall is the number of relevant results that have been presented over total set of relevant results. Both precision and recall are based on an a measure of relevance.

For classification, the terms true positives, true negatives, false positives, and false negatives compare the results of the classifier to be tested with external trusted evaluations. The terms positive and negative refer to the classifier's output, while the terms true and false refer to whether that classification is compliant with the one performed from the trusted external evaluation.

- **True Positive (TP):** positive observation classified as positive by the evaluation process to be tested.
- **False Positive (FP):** positive observation wrongly classified as negative by the evaluation process to be tested.
- **True Negative (TN):** negative observation classified as negative by the evaluation process to be tested.
- **False Negative (FN):** positive observations wrongly classified as negative by the evaluation process to be tested.

Provided these preliminary definitions, now we can formally define *precision* and *recall* as follows:

$$precision = \frac{TP}{TP + FP}$$

$$recall = \frac{TP}{TP + FN}$$

Chapter 6

Experimental Results

In this chapter we are going to provide experimental results regarding the proposed solution. All the experiments have been run on a selected day and location where we have the proof that a accident happened.

kisalfold.hu | 24 ÓRA | SZÓRAKOZÁS | SPORT | PROGRAMOK | ÁLLÁS | INGATLAN | EXTRA

Két súlyos baleset - Meghalt egy utas, mentőhelikopter vitt egy sofőrt a győri kórházba

KISALFOLD.HU 2016.09.27. 18:58

Kedden délután, kora este súlyos közlekedési baleset is történt az 1-es főúton, illetve az M1-es autópályán.

Meghalt egy ácsi férfi a kedden délután Nagyigmándnál történt balesetben. A Suzukiban utazott, amellyel a sofőrje Kócs irányából kanyarodott ki a 13-as számú útra, a 13-as kilométerszelvénynél. A rendőrség tájékoztatása szerint az autó vezetője súlyos sérüléseket szenvedett, amikor összeütközött járműve a Komárom felől Kisbér felé haladó személyautóval. Egy ráfutásos baleset is történt kedden délután, az M1-es autópálya 94-es kilométerszelvényénél a Budapest felé vezető oldalon. A Concó pihenőnél, Bábolna térségében a külső sávban állt egy teherautó, jobb hátuljának ment neki a Ford bal eleje. Aszfaltozást végeznek a közelben, de arról még nem tudtunk információkat szerezni, ezzel összefügg-e az eset. Vélhetően a feltorlódott járművek miatt rántotta el a kormányt a személyautó sofőrje, mert nem tudott időben megállni. A rendőrség tájékoztatása szerint egy időre - a helyszínelés és a mentés miatt - az autópálya forgalmát egy szakaszon az 1-es útra terelték. Mentőhelikoptert is hívtak a helyszínre, a személyautó vezetőjét életveszélyes sérülésekkel szállították a győri kórházba.Dr. Oláh Attila professzor tájékoztatása szerint arckoponyasérülést szenvedett, több komoly törése, sorozatbordatörése is van, tüdősérülése és alkartörése. Állapota súlyos, de stabil, az intenzív osztályon lélegeztetőgépen tartották, készültek megműtésére.

19. 00 - Ismét zavartalan a forgalom az M1-es autópályán, ahol korábban a Budapest felé vezető oldalon a 94-es kilométerszelvénynél személyi sérüléses közúti közlekedési baleset történt. A rendőrök a helyszínelést befejezték az érintett pályaszakasz ismét teljes szélességben járható. Személyi sérüléses közúti közlekedési baleset történt az M1-es autópálya Budapest felé vezető oldalán 2016. szeptember 27-én 16 óra 30 perc körüli a 94-es kilométerszelvénynél. Az elsődleges adatok alapján egy személygépkocsi az előtte haladó teherautónak ütközött. A balesetben a személygépkocsi vezetője életveszélyesen megsérült. A rendőrök a helyszínelés és a műszaki mentés idejére a pályaszakaszt lezárták. A forgalmat a 100-as kilométerszelvénynél az 1-es számú főútra terelik. Az autópályára visszatérni a 85-ös kilométerszelvénynél lehetséges. Kérjük óvatosan vezessenek - közölte a rendőrség.

Figure 6.1: An article from kisalfold.hu Reporting the Accident

The highway car accident we are talking about happened on *September the 27th, 2016 around 4:30PM* the highway interested was the M1, the street have been out of usage for three hours before the car removal operations, that have been accomplished around three hours later since the accident happened.

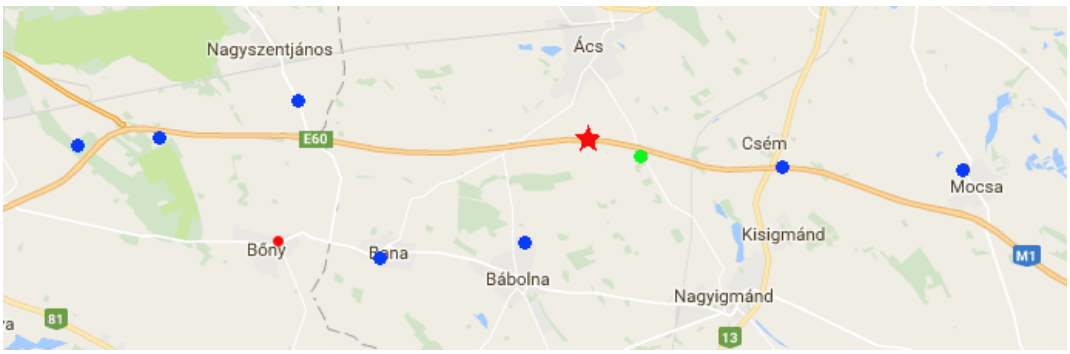


Figure 6.2: The location of the highway car accident.
47 40’58.9"N 18 00’36.0"E

Figure 6.2 illustrates the area of the accident, the red star represent the estimated location of the anomalous event, blue points are the cell towers estimated to keep track of the activity of the highway while the light-green dot represent the closest cell to the accident location, the activity of this cell tower is expected to be highly influenced by the specific anomalous event more than the others.

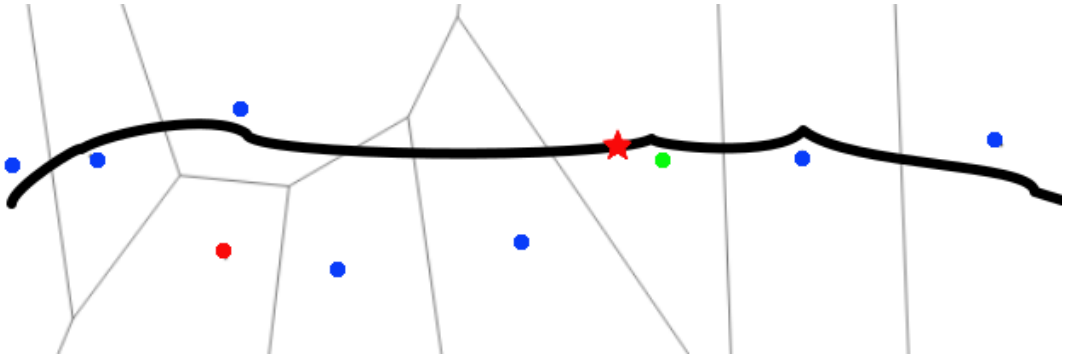
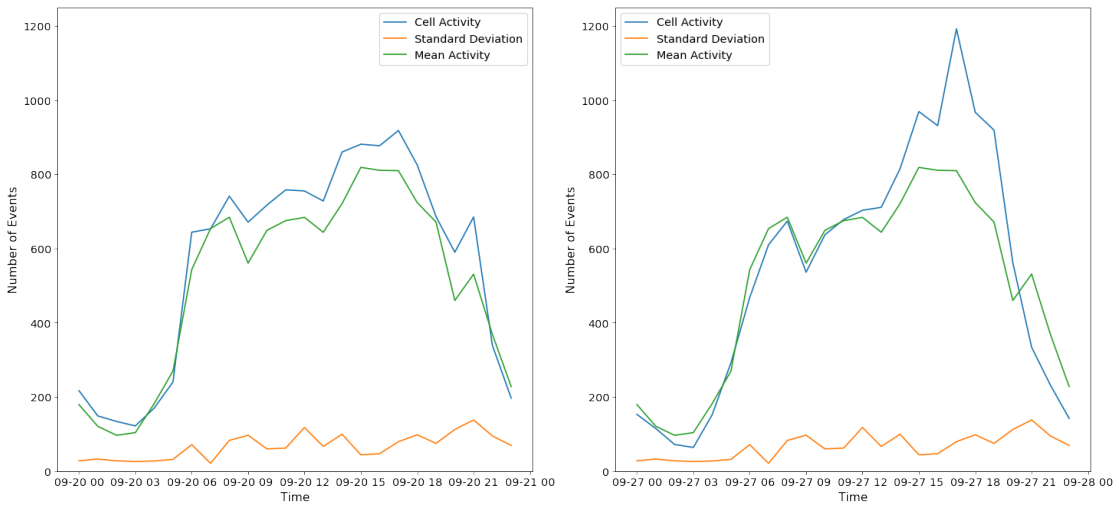


Figure 6.3: The location of the highway car accident.
47 40’58.9"N 18 00’36.0"E

Figure 6.3 represents the estimated Voronoy cells for the interested area, the Voronoy diagram helped us to better understand which cell towers were interested by the specific anomalous event. The dot notation is the same of Figure 6.2.

Given the lack of msc data regarding the interested cell tower all the experiment were performed making exclusively use of ngprs data. As observed on section 5.1.4 the nature of ngsprs logs (if compared with the msc data) is dominated by random noise or irregular fluctuations, thus, following a logical reasoning the performances of the solution applied to msc data are expected to be higher in respect of the ngprs data. In conclusion we performed experiments on the worst scenario.

In the following part of this chapter we are going to analyze the results of the experiment following a progressive schema: first of all we will present the result with a lower sampling resolution then we will discuss results of experiments with higher sampling rates.



(a) NGPRS September the 20th - Normal Activity (b) NGPRS September the 27th - Anomaly

Figure 6.4: Detailed Average Activity vs Anomaly Activity

Figure 6.4 wants to provide a graphical representation (with an hourly aggregation) of the normal activity of the cell Figure (a) compared with the anomaly recorded one week later on the same cell on Figure (b). Is evident that between the two graphs there is a notable difference on their characteristics especially between 4PM and 20PM. In this specific context case we consider as anomaly points the whole time slots between the accident happening time (4:30PM) and the end of the car removal operations (7:30PM).

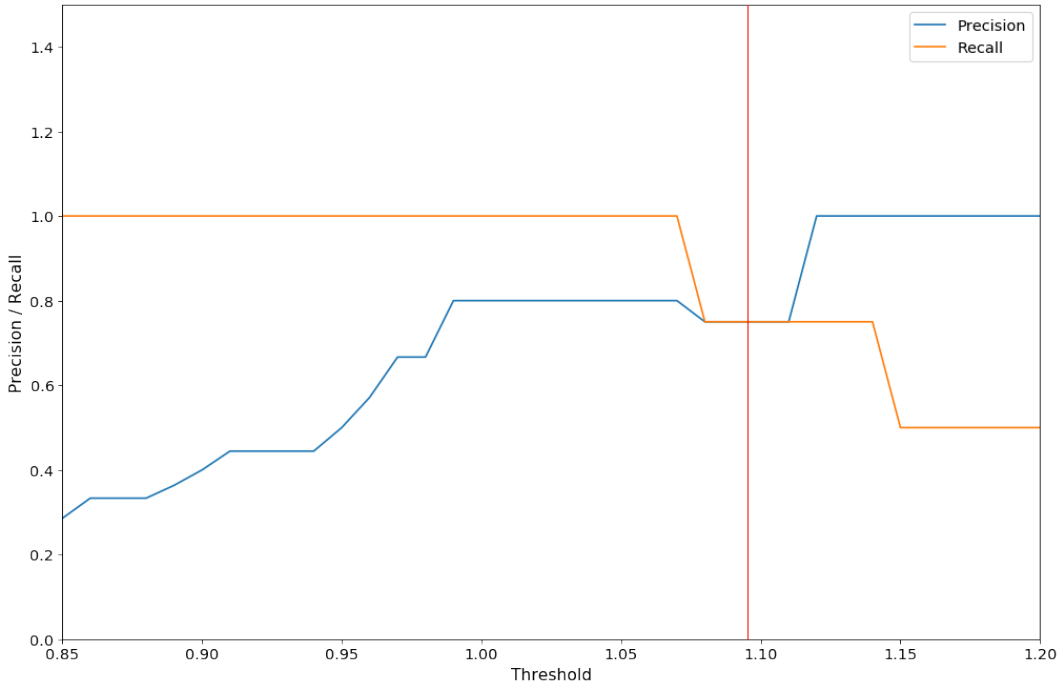


Figure 6.5: Precision and Recall for Hourly Aggregation at Variation of A_{thr}

6.1 NGPRS T60 STD

Here by following experiment represents the activity of the interested cell tower with an hourly sum aggregation (Definition 5.1.3) thus with an $r = 3600$.

6.1.1 The Pure NGPRS Dataset STD

In figure 6.5 the graph represents the variation of the precision and recall at variation of the threshold variable (A_{thr}). Observing the graph is possible to note that the optimal solution is not present (with optimal solution we intend when both precision and recall are equal to one).

This issue can be addressed to just one fact, given the training set of three weeks (too few) the sample standard deviation of the normal activity in some points of the function is too low. This leads our technique to spot false positives as soon as light (but not observed in the training set) statistical fluctuation are observed on the test data.

The cause of the miss classification can be found looking at the graph of the standard

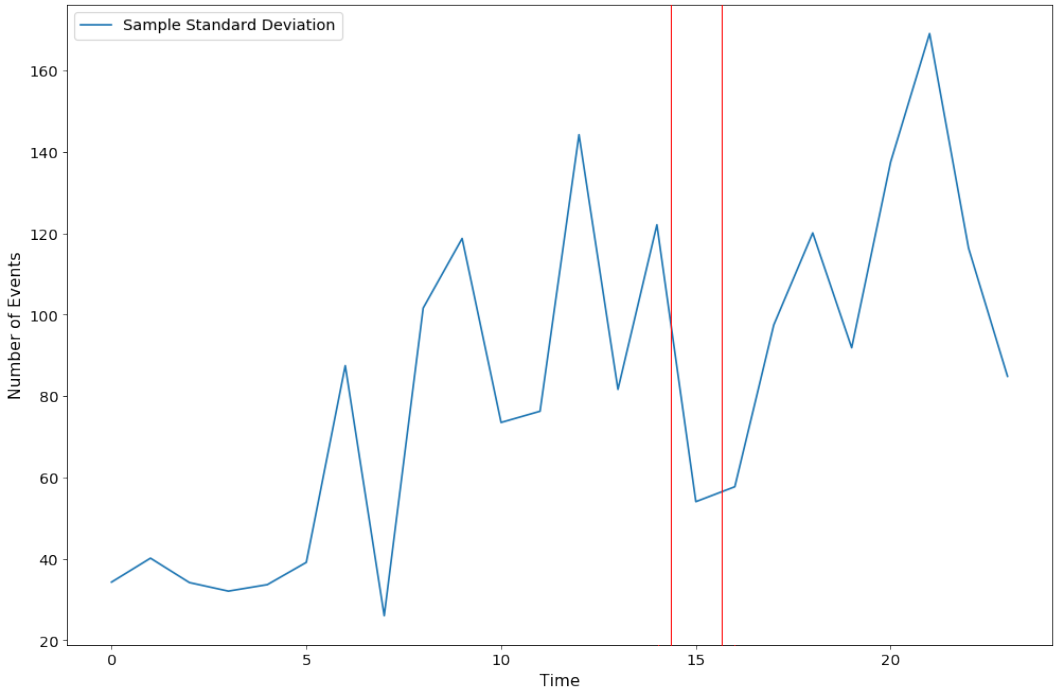


Figure 6.6: Standard deviation of the training data

deviation (Figure 6.6), the value between the red lines represents the standard deviation of the time slot between 3PM and 4PM, in that point the standard deviation reaches one of the minimal values (others low values in the graph resulted acceptable because they refer to nightly hours where a minimal activity is expected thus a lower values of the standard deviation are expected) while in the surrounding time slots of the time series values are pretty higher.

Finally Figure 6.7 represent the actual observations of the anomalous event (the blue function) over the estimated critical values (in red). In this case the A_{thr} value are 1.09 (denoted by the red line oi Figure 6.5). As we can note the false positive outlier point is again between the red lines, as we can see the difference between the estimated critical point and the observed value is minimal. This lead us to believe that with anther meaningful observation on the dataset this issue can be overcomed. We will give the proof of this assumption on the next described experiment.

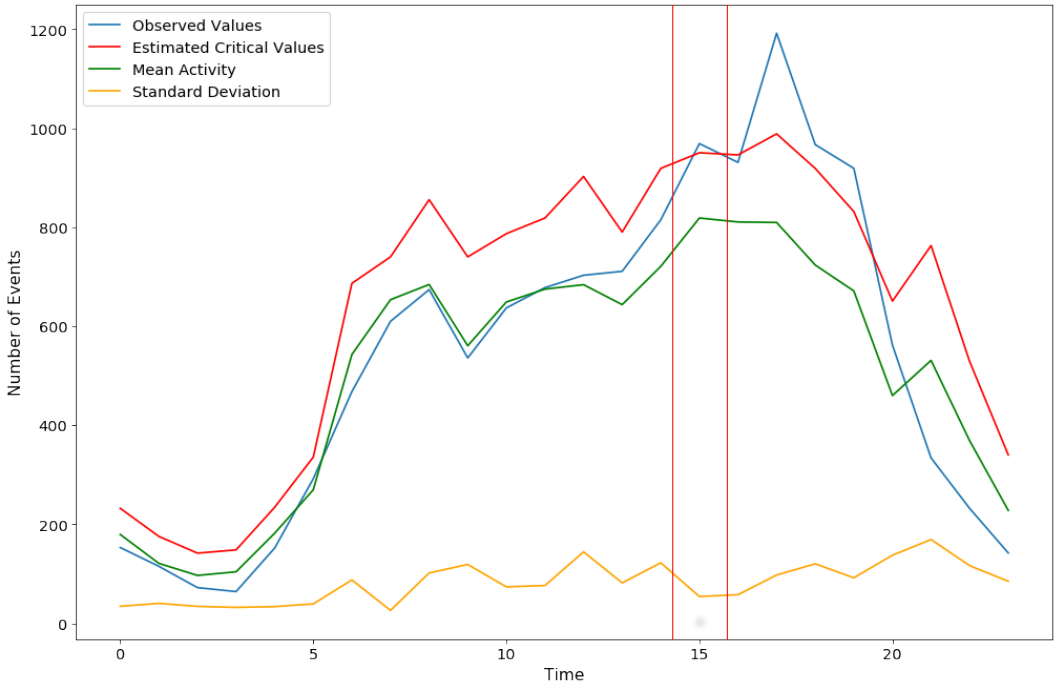


Figure 6.7: Observed Values Over the Estimated Critical Values

6.1.2 The Augmented NGPRS Dataset STD

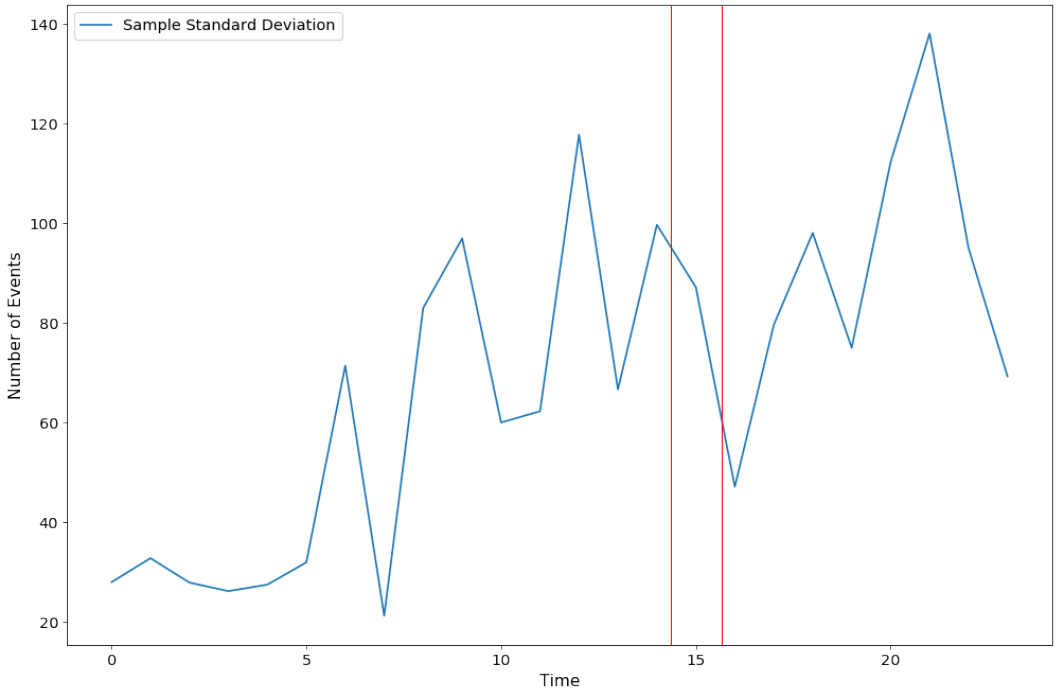


Figure 6.8: Standard deviation of the new training data

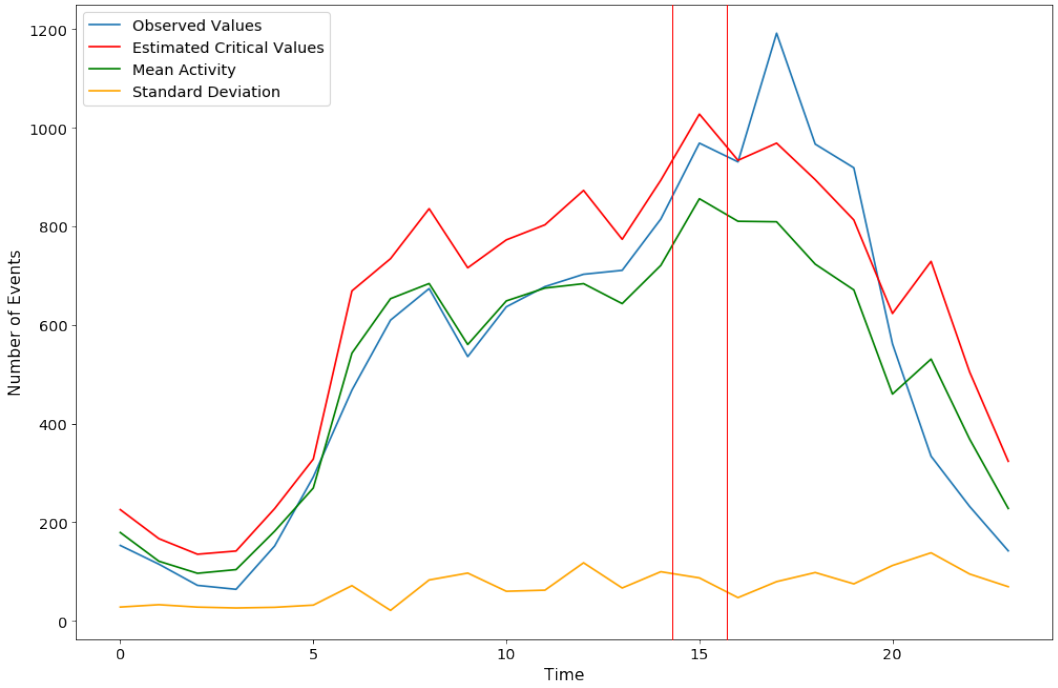


Figure 6.9: Observed Values Over the new Estimated Critical Values

In order to test if the number of observations had an impact or not on the result of our experiment we generated an artificial observation: the generated time series is composed by the mean activity of the cell computed from the observation in the dataset, except for the false positive point, there we substituted the value with the activity of the cell from the real observation, namely we took the same value of the false positive spotted during the experiment. We did so that the new observation did not have an impact on the shape (but just on the scale) of the overall standard deviation but just in the targeted point (Figure 6.8 illustrate the new standard deviation). Doing so, as expected, with a new artificial observation this time the estimated critical values changed with a positive impact. The threshold value have been affected by the new standard deviation, in fact, with a new observation we needed to re-test and re-calibrate the value but still the size range of optimal values is the same. As shown by Figure 6.10 now the best threshold values are between 1.02 and 1.08. With this set up we detected the anomaly within half hour since the event happened, in the worst case the delay can reach one hour.
Finally we tried to push this solution with a time resolution of 30 minutes, but the experiments showed that in this specific case there existed just one optimal point. Since the threshold value could change depending on the specific cell and its activity we concluded that this model can not apply on smaller time resolutions, or at least can not be generalized.

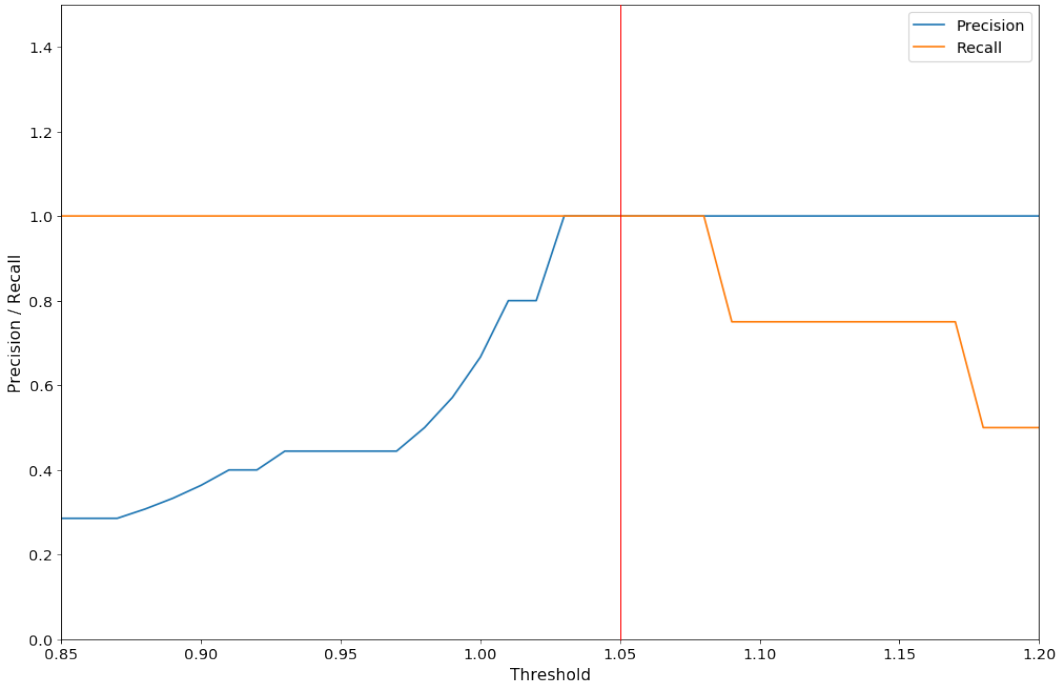


Figure 6.10: New Precision and Recall for Hourly Aggregation at Variation of A_{thr}

6.2 NGPRS T1 WMM

Whithin this section we are going to present results about the statistical approach. Here by the following experiment represents the activity of the interested cell tower with a one minute sum aggregation (Definition 5.1.3) thus with an $r = 60$.

Since the WMM test do not need any parameter tuning we are going to show the result mainly in a tabular format. The threshold value selected for the *p-value* mean is 0.01, it have been selected as the best value after few experiments.

6.2.1 WMM Not Overlapped

As we can see from Table 6.1 results from the statistical approach seems to be more stable. The table contains the size of the time windows the precision and the recall of the classification and finally the amount of time needed to the system to spot the anomaly.

We started the tests with bigger windows and then gradually decreasing the width to measure how stable is the solution while computing the *p-value* with less samples.
For time windows of one hour the WMM provided an excellent result in precision (1.0)

with no false positive at all, however the beginning of the anomaly have been spotted in 90 minutes, so this result appear to be unacceptable for real word applications, the recall appear to be quite good, only the first time window after the accident have been classified as a normal behaviour (false negative). With a 30 minutes interval the system managed to spot the anomaly with 60 minutes delay, both recall and precision are acceptable; similar results have been obtained with a time window of twenty minutes, however surprisingly, in this case the anomaly have been spotted with a 20 minutes delay. While with a fifteen minutes windows width the system did not react properly, despite the precision of the result is high, a lot of positive points have been classified as normal behaviour (many false negatives) spotting the anomaly within 30 minutes.

Windows Size	**Precision**	**Recall**	**Delay**
60 m	1.000	0.750	90m
30 m	0.800	0.666	60m
20 m	0.833	0.625	20m
15 m	1.000	0.416	30m

Table 6.1: WMM Results without Windows Overlapping

6.2.2 WMM Overlapped Symmetric Windows

In this part follow the result obtained making use of overlapping windows described on Section 5.2.2 in this case the size of the overlapping is equal to the size of the window width (symmetric overlapping).

Windows Size	**Precision**	**Recall**	**Delay**
60 m	0.800	1.000	30m
30 m	1.000	1.000	30m
20 m	0.555	0.625	20m
15 m	0.777	0.583	30m

Table 6.2: WMM Results with Windows Overlapping

Whith the windows overlapping the whole set of results have been improved. Looking at the first line of the table we can see that the time needed to spot the anomaly have been decreased with one hour in advance (if compared to Table 5.2.2) but in this case the solution clearly have been affected by a false positive, this can be inferred by the value of the

precision. The best results have been achieved with a window's width of 30 minutes, here, in this case both recall and precision are equal to one (optimal result) and the anomaly have been spotted with thirty minutes delay, this is not a bad result since the accident happened at 16:30 exactly when the the computation for the next half-hour started. The twenty minutes width did prodice the same results as non overlapping while the fifteen minutes width decreased in precision but gained few points in recall.

As we can infer from the table the best compromise in this case is represented by the 30 minutes width with an exalt classification and at the same moment with an acceptable delay on the anomaly spotting.

6.2.3 WMM Overlapped Asymmetric Windows

Finally since the results of the 20 and 15 minutes window's width did not provide better results even with the overlapping technique we tried to understand if the nature of the problem was in the number of sampling contained in such windows or not. To do so we tried to perform experiments with bigger overlapping values.

Windows Size	**Overlapping Size**	**Precision**	**Recall**	**Delay**
20 m	40 m	0.633	0.875	20m
20 m	60 m	0.714	0.769	20m
15 m	30 m	0.846	0.846	15m
15 m	45 m	0.916	0.846	15m

Table 6.3: WMM Results with Asymmetric Windows Overlapping

As we can infer from table 6.3 the overall performances are improved. In this case the best result have been archived with windows of fifteen minutes width with an overlapping of 45 minutes. The assumption we made was true, apparently at least with this kind of data-set the WMM test lead to better performances when the size of the samples is close to 60 points, bigger samples lead to loose in granularity while smaller ones are conditioned from the nature of the data.

A final note on the false positive is needed. During our experiments with the WMM approach mostly all the false positive having a bad impact on the omputed precision of the method were points in between 21PM and 22PM (and few between 12AM and 13 AM). Looking at Figure 6.6 we can see that in that intervals the standard deviation reaches the absolute top values in the chart. From this we can infer that the sampling in that interval

are difficult to model since the points apparently do not follow a similar probability distribution. In this case further investigations to understand the real nature of the problem are needed.

6.3 Results Summary and Considerations

Comparing the two techniques having two different theoretical foundations, lead us to state that the statistical approach is both more stable and reliable. Despite the standard deviation approach provided acceptable results for intervals of one hour, decreasing the intervals width lead to a less accurate precision of the results. In fact, the standard deviation approach with smaller time windows have been demonstrated to be highly sensitive to outliers and random fluctuation. So, at least, we can deduce that the model seems to not properly fit with the nature of the NGPRS data. In addition this model depends totally on the calibration of parameters based on the standard activity of the area covered by a cell tower. The A_{thr} value to get the most accurate performances should be fine tuned depending both on the base normal activity of the area and the number of past observations involved during the evaluation process. The calibration phase for several areas would need time and resources making it hard to scale in a real scenario.

On the other side, with this data, the statistical approach with overlapping windows lead to way better results and it have been demonstrated to be stronger to false positives even with smaller time windows. The best result on the worst case is able to spot anomalies within 30 minutes delay while with the first method within one hour. Even with smaller overlapping time windows the result seem to produce results with an acceptable level of accuracy. As opposed to the first method the statistical analysis we made use of does not have any parameter to be tuned, thus, given this intrinsic nature of the model, potentially, it would be easier to scale within a real scenario.

Interesting to note is that the weak point of the standard deviation approach is the when in some points the computed standard deviation is low leading to false positives, while the statistical approach, in some cases (with small time windows) appeared to be sensitive to false positives exactly when the standard deviation of past observations reaches the highest values.

Chapter 7

Conclusions and Future Work

7.1 Final Considerations

Nowadays modern companies are overwhelmed of data. Big Data collection and exploitation is an evident trend on which companies bet to ensure their presence in the market of the future. Telecom providers are the ones among many.

In this thesis work we started analyzing the potential applications of the data collected by a mobile telecom operator. Then we decided to model and understand the relation of the data logs with the mobility infrastructures. The question we had was: how can the status of the mobility infrastructure have an impact on the telecommunication activity? More precisely we tried to focus on highways and how to infer anomalies on the mobility from the telecom point of view. In order to answer to this question an analysis of the literature have been performed both to understand how further the research went on this direction and to get known on what are the main limitation and new challenges at the moment of writing.

The outcome from the literature was positive, telecom infrastructures and mobility infrastructure are correlated, and the status of the one can influence the activity of the other, thus, providing to us the basis to start a new research. After this we performed a deep analysis study on the data we received from *Magyar Telekom*. Once we gained a deep knowledge about the main characteristics and features of the data-set we tried, with success, to model the behaviours of users related with mobility and its anomalies. Moreover we provided two original solutions apparently never applied on this specific problem set.

As stated in [7] and remarked in subsection 2.2.1 one of the limitation of the usage of telecom data for mobility is the individual right of privacy. So, since the early stages of the

research we avoided to use Origin Destination matrices or models based on the analysis and modelling of the individual mobility of subscribers. Then as pointed out from [4] and [7] passive methods are preferred for many reasons in addition they hold considerable advantages, we decided to follow this direction as well for several reasons: scalability issues, real scenario applicability and data constraints.

So we decided to tackle the problem from the perspective of the cell towers and their activity over time. Analyzing the dataset we found out that the daily activity of a cell (determined by its geographical location and the behaviours of the users "connected" to it) has a certain regularity. To represent the information we opted for a model that takes into consideration values over time so we decided to make use of time series. Once we selected the right representation of the data further researches on anomaly detection have been performed.

Finally we tested and presented two models based on different theoretical roots, the first based on the mean activity of a cell and its standard deviation and the second based on statistical analysis. Both models gave positive encouraging results. The statistical approach revealed to be more stable and easier to scale while the one based on the standard deviation needs to be fine tuned for each cell. The outcome of this research have been positive, accidents (and anomalies in general) happening on highways can be spotted analyzing the mobile telecom activity on totally respect of subscriber's privacy and with a reasonable amount of time.

As opposed to other solutions already present in the literature, both the proposed models in case of real scenario deployment have no impact on the infrastructures involved, no extra hardware on the highways is needed, nor the quality of service of telecommunication serviced would be affected. Both the presented solutions have minimal computational costs making easy the deployment in a real scenario with a low financial investment, this latter statement especially apply for the statistical approach given its intrinsic non-parametric easy to scale nature.

7.2 Future Work

Since the dataset we had did not cover a wide time range (just one month) our first objective is to perform more experiments on real data to better validate the models. Then

a future objective is to infer the direction of the anomaly, this can be done at least in two ways: the first one could be based on the correlation of two neighbour cells, in fact, if a car accident happened, and the road is blocked, we expect that the activity of a cell will increase (the one covering the accident location) while the next cell displaced further on the road should be affected by a lower activity. Analyzing the correlation of the cells, more precisely, if there is a negative correlation between the activity of two cells the direction can be easily inferred. Another possible technique computationally more expensive involves the utilization of the previous location of subscribers. The basic idea is to store the last location of each user and once an anomaly is spotted a model based on vectors can be utilized to infer the direction of the unexpected event. Then the next step further would be to project a real-time solution and test its performances and costs on a real scenario.

Acknowledgement

Few words must be spent to tank all the EIT institution and its amazing staff, thanks to whom proposed this project in the European Union and their efforts to support education.

A special thank to the UniTN and ELTE staff.
Especially I want to explicitly thank Tomas Horvath for his patience and his helpful hints.

Bibliography

[1] V. Asthenopoulos, I. Loumiotis, P. Kosmides, E. Adamopoulou, and K. Demestichas. Traffic estimation through mobile network performance data processing. *WSEAS Transactions on Communications*, 2016.

[2] V. D. Blondel, A. Decuyper, and G. Krings. A survey of results on mobile phone datasets analysis. *EPJ Data Science*, 4(1):10, 2015.

[3] J. Candia, M. C. González, P. Wang, T. Schoenharl, G. Madey, and A.-L. Barabási. Uncovering individual and collective human dynamics from mobile phone records. *Journal of Physics A: Mathematical and Theoretical*, 41(22):224015, 2008.

[4] D. Gundlegard and J. M. Karlsson. Generating road traffic information from cellular networks-new possibilities in umts. In *ITS Telecommunications Proceedings, 2006 6th International Conference on*, pages 1128–1133. IEEE, 2006.

[5] A. Janecek, K. A. Hummel, D. Valerio, F. Ricciato, and H. Hlavacs. Cellular data meet vehicular traffic theory: location area updates and cell transitions for travel time estimation. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, pages 361–370. ACM, 2012.

[6] S. Jiang, G. A. Fiore, Y. Yang, J. Ferreira Jr, E. Frazzoli, and M. C. González. A review of urban computing for mobile phone traces: current methods, challenges and opportunities. In *Proceedings of the 2nd ACM SIGKDD international workshop on Urban Computing*, page 2. ACM, 2013.

[7] A. Milani, E. Gentili, and V. Poggioni. Cellular flow in mobility networks. *IEEE Intelligent Informatics Bulletin*, 10(1):17–23, 2009.

[8] T. Raeder, O. Lizardo, D. Hachen, and N. V. Chawla. Predictors of short-term decay of cell phone contacts in a large scale communication network. *Social Networks*, 33(4):245–257, 2011.

[9] J.-G. Remy. Computing travel time estimates from gsm signalling messages: the strip project. In *Intelligent Transportation Systems, 2001. Proceedings. 2001 IEEE*, pages 6–9. IEEE, 2001.

[10] G. Tibély, L. Kovanen, M. Karsai, K. Kaski, J. Kertész, and J. Saramäki. Communities and beyond: mesoscopic analysis of a large social network with complementary methods. *Physical Review E*, 83(5):056125, 2011.

[11] M. Weiss, M. Elsner, F. Kartberg, and T. Nilsson. Anomalous subdiffusion is a measure for cytoplasmic crowding in living cells. *Biophysical journal*, 87(5):3518–3524, 2004.

[12] G. K. Zipf. *Human behavior and the principle of least effort: An introduction to human ecology.* Ravenio Books, 2016.