BRNO UNIVERSITY OF TECHNOLOGY
VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ

DEPARTMENT OF RADIO ELECTRONICS
ÚSTAV RADIOELEKTRONIKY

# LOCATION AWARE ANALYTICS IN THE CONTEXT OF MOBILE NETWORK PERFORMANCE OPTIMIZATION

POLOHOVĚ ORIENTOVANÁ ANALÝZA DAT V KONTEXTU OPTIMALIZACE MOBILNÍCH SÍTÍ

MASTER'S THESIS
DIPLOMOVÁ PRÁCE

AUTHOR
AUTOR PRÁCE

Bc. Lucie Urbanová

SUPERVISOR
VEDOUCÍ PRÁCE

doc. Ing. Martin Slanina, Ph.D.

BRNO 2019

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

# Diplomová práce

magisterský navazující studijní obor **Elektronika a sdělovací technika**
Ústav radioelektroniky

***Studentka:*** Bc. Lucie Urbanová ***ID:*** 151753
***Ročník:*** 2 ***Akademický rok:*** 2018/19

**NÁZEV TÉMATU:**

## Polohově orientovaná analýza dat v kontextu optimalizace mobilních sítí

**POKYNY PRO VYPRACOVÁNÍ:**

Seznamte se se základními technikami zpracování velkých souborů dat a hledání souvislostí v těchto souborech. Seznamte se s koncepcí, funkcionalitou a obsluhou mobilní aplikace "netztest", kterou využívá rakouský telekomunikační regulátor RTR. Zjistěte podíl a typ geografických oblastí, pro které zatím měření prostřednictvím netztest neexistují. Vytvořte koncepci odhadu měřených bitových rychlostí dosažitelných v libovolné poloze prostřednictvím nejnovějších metod regrese.

Vytvořte softwarový nástroj, který bude schopen vytvořit odhad měřené bitové rychlosti na místech, kde skutečná měření neexistují. Vyhodnoťte alespoň dvě metody regrese s ohledem na jejich přesnost a složitost. V experimentální části práce proveďte referenční měření aplikací netztest a porovnejte výsledky s vytvořeným modelem.

**DOPORUČENÁ LITERATURA:**

[1] RASMUSSEN, C. E., WILLIAMS, C. Gaussian processes for machine learning. Cambridge, Mass.: MIT Press, c2006. Adaptive computation and machine learning. ISBN 0-262-18253-X. [Online] Dostupný z: http://www.gaussianprocess.org

[2] RTR Open Data [Online]. Dostupný z WWW: www.rtr.at/en/inf/RTROpenData

***Termín zadání:*** 4.2.2019 ***Termín odevzdání:*** 16.5.2019

***Vedoucí práce:*** doc. Ing. Martin Slanina, Ph.D.
***Konzultant:***

**prof. Ing. Tomáš Kratochvíl, Ph.D.**
*předseda oborové rady*

BRNO UNIVERSITY OF TECHNOLOGY
FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

# Master's Thesis

Master's study field **Electronics and Communication**
Department of Radio Electronics

***Student:*** Bc. Lucie Urbanová ***ID:*** 151753

***Year of study:*** 2 ***Academic year:*** 2018/19

**TITLE OF THESIS:**

## Location Aware Analytics in the Context of Mobile Network Performance Optimization

**INSTRUCTION:**

Get acquainted to the techniques of processing big sets of data and searching for relationships within such data. Create an overview of the concept, functionality and the interface of a mobile app “netztest”, which is used by the Austrian telecommunication regulator RTR. Identify the share and type of geographical areas, which are not covered by measurements of netztest so far. Create a concept of estimating the data rate measured at an arbitrary location using state of the art regression methods.

Create a software tool capable of estimating the measurement data rate at locations where measurements are missing, Evaluate at least two regression methods with respect to complexity and accuracy. In the experimental part of the work, perform reference measurements with netztest to compare the results with the modeled data.

**RECOMMENDED LITERATURE:**

[1] RASMUSSEN, C. E., WILLIAMS, C. Gaussian processes for machine learning. Cambridge, Mass.: MIT Press, c2006. Adaptive computation and machine learning. ISBN 0-262-18253-X. [Online] Dostupný z: http://www.gaussianprocess.org

[2] RTR Open Data [Online]. Dostupný z WWW: www.rtr.at/en/inf/RTROpenData

***Date of project specification:*** 4.2.2019 ***Deadline for submission:*** 16.5.2019

***Supervisor:*** doc. Ing. Martin Slanina, Ph.D.
***Consultant:***

**prof. Ing. Tomáš Kratochvíl, Ph.D.**
*Subject Council chairman*

**WARNING:**

The author of the Master's Thesis claims that by creating this thesis he/she did not infringe the rights of third persons and the personal and/or property rights of third persons were not subjected to derogatory treatment. The author is fully aware of the legal consequences of an infringement of provisions as per Section 11 and following of Act No 121/2000 Coll. on copyright and rights related to copyright and on amendments to some other laws (the Copyright Act) in the wording of subsequent directives including the possible criminal consequences as resulting from provisions of Part 2, Chapter VI, Article 4 of Criminal Code 40/2009 Coll.

# ABSTRAKT

Předmětem této práce je polohově orientovaná analýza v kontextu optimalizace mobilních sítí. Popisuje nástroj pro odhadování základních parametrů sítě na místech s neznámými parametry sítě na základě databáze RTR NetTest. Je zde stručně představena oblast velkých dat, strojového učení a shrnutí o konceptu a funkcionalitě aplikace NetTest. Práce ukazuje a porovnává skupinu regresních metod na základě jejich komplexnosti a vhodnosti pro vytvoření map odhadovaných parametrů sítě. Po jejich důkladné 1D analýze je IDW a GPR analyzováno ve 2D a využito pro vytvoření skupiny map odhadu parametrů sítě. Je posouzena i jejich přesnost na základě referenčního měřeni aplikací NetTest.

# KLÍČOVÁ SLOVA

GPR, IDW, Mapa pokrytí, Regrese, RTR-NetTest

# ABSTRACT

This thesis deals with the location aware analytics in the context of mobile network performance optimization. A tool which estimates initial network parameters in the location with unknown network performance based on RTR NetTest measurements database is presented. The thesis briefly introduces the topic of big data and machine learning and gives an overview of NetTest application concept and functionality. A set of regression methods is presented and their complexity and suitability for the purposes of coverage maps creation is compared. After their thorough 1D analysis, IDW and GPR are analysed in 2D and used to create a set of estimation maps of network parameters. Evaluation of their accuracy is made based on reference measurements using NetTest application.

# KEYWORDS

Coverage Map, GPR, IDW, Regression, RTR-NetTest

# ROZŠÍŘENÝ ABSTRAKT

S narůstajícími nároky na bezdrátové sítě, jejich spolehlivost, schopnost reagovat a pracovat, je pro operátory sítí velmi důležité rychle a spolehlivě navázat spojení mezi uživatelským zařízením a sítí bez nežádoucího přetěžování sítě s vhodným přidělením zdrojů. Tato práce prezentuje základ pro řešení, které odhaduje základní parametry sítě uživatelského zařízení a vytváří mapy odhadovaného chování parametrů sítě na základě strojového učení s využitím regresních metod.

Práce stručně shrnuje téma velkých dat, historii umělé inteligence a strojového učení. Jsou zde vysvětleny základní parametry posuzované při popisu souborů velkých dat a techniky používané při práci s velkými databázemi.

Je shrnuta celá procedura vedoucí k vytvoření map odhadovaného pokrytí parametry mobilních sítí a zároveň jsou uvedeny hlavní výzvy vyvstávající při práci s dostupnými daty, zahrnující hustotu proměření oblasti Rakouska, změny v zatížení sítě nebo limitace tarifu uživatele.

Práce rovněž obsahuje popis a funkcionalitu aplikace RTR NetTest, která slouží jako hlavní zdroj vstupních dat pro odhad parametrů sítě. Po popisu obsluhy aplikace a uživatelského rozhraní je popsána procedura RMBT testování a jsou detailně vizualizovány možnosti zobrazení výsledků měření na stránce databáze RTR. Na základě předchozích informací je oblast zahrnující měření rozdělena do několika kategorií. Výsledky měření jsou jako veřejně přístupná data dostupná na zmíněné stránce a jsou detailně popsána. Následně je představen další zdroj dat.

Důkladná analýza několika regresních metod je uvedena a doplněna o jejich teoretické pozadí, dále o skupinu vyhodnocovacích metod, včetně střední průměrné chyby, která je použita pro posouzení přesnosti a komplexnosti každé z regresních metod. Po jejich důkladné 1D analýze jsou inversní váhování vzdálenosti a Gaussova směsná regrese analyzovány ve 2D a jsou použity pro vytvoření skupiny map odhadů parametrů sítě v místech, pro která nejsou dostupná referenční měření. Jejich přesnost je posouzena na základě referenčních měření využívajících NetTest aplikaci a na základě statistické distribuce dat.

V závěru jsou posouzeny informace prezentované v této práci a jsou shrnuty nejdůležitější poznatky a závěry. Rovněž je zde poskytnut základ pro možnosti rozšíření této práce.

BIBLIOGRAPHIC CITATION

URBANOVÁ, L. *Location Aware Analytics in the Context of Mobile Network Performance Optimization*. Brno University of Technology, The Faculty of Electrical Engineering and Communication, Department of Radio Electronics, 2018. Diploma thesis supervisor Dr. Philipp Svoboda and doc. Ing. Martin Slanina, Ph.D. Available at: https://www.vutbr.cz/studenti/zav-prace/detail/118463.

# PROHLÁŠENÍ

Prohlašuji, že svoji diplomovou práci na téma “Polohově orientovaná analýza dat v kontextu optimalizace mobilních sítí” jsem vypracovala samostatně pod vedením vedoucího semestrální práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušila autorská práva třetích osob, zejména jsem nezasáhla nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědoma následků porušení ustanovení § 11 a následujících zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

V Brně dne ............................. ..................................

(podpis autorky)

# PODĚKOVÁNÍ

Ráda bych poděkovala Dr. Philippovi Svobodovi za cenné rady, věcné připomínky a vstřícnost při konzultacích. Mé poděkování dále patří Dipl. Václavu Raidovi, Ing. za pomoc při práci a za všechny cenné rady, které mi poskytl. Rovněž děkuji doc. Ing. Martinu Slaninovi, Ph.D. za pomoc s realizací projektu a stipendijnímu programu Aktion pod záštitou OeAD za finanční podporu, která umožnila realizaci tohoto projektu. Závěrem bych ráda poděkovala Mgr. Agatě Klusové za pomoc s kontrolou anglického pravopisu.

V Brně dne: …………………………

podpis autorky

# DECLARATION

I declare that I have written my diploma thesis on the theme of "Location Aware Analytics in the Context of Mobile Network Performance Optimization" independently, under the guidance of the diploma project supervisor and using the technical literature and other sources of information which are all cited in the project and detailed in the list of literature at the end of the project.

As the author of the diploma thesis I furthermore declare that, as regards the creation of this diploma thesis, I have not infringed any copyright. In particular, I have not unlawfully encroached on anyone's personal and/or ownership rights and I am fully aware of the consequences in the case of breaking Regulation S11 and the following of the Copyright Act No 121/2000 Sb. of the Czech Republic, and of the rights related to intellectual property right and changes in some Acts (Intellectual Property Act) and formulated in later regulations, inclusive of the possible consequences resulting from the provisions of Criminal Act No 40/2009 Sb. of the Czech Republic, Section 2, Head VI, Part 4.

Brno ............................. ...................................

(author's signature)

# ACKNOWLEDGMENT

I would like to thank Dr. Philipp Svoboda for valuable counsel, objective remarks and helpfulness during consultations. My thanks also belong to Dipl. Vaclav Raida, Ing. for his help and valuable insights. Also, I would like to thank doc. Ing. Martin Slanina, Ph.D. for his help with the realisation of the project and the scholarship programme Aktion under OeAD for financial support that allowed for this thesis to happen. Finally, I would like to thank Mgr. Agata Klusová for her help with English grammar correction.

V Brně dne: ..............................

author's signature

# TABLE OF CONTENT

# LIST OF SYMBOLS AND ABBREVIATIONS

**ABBREVIATIONS:**

| | | |
|---|---|---|
| AI | … | Artificial Intelligence |
| AKOS | … | Agency for communication networks and services of the Republic of Slovenia |
| ASCII | … | American Standard Code for Information Interchange |
| AT | … | Austria |
| BSSID | … | Basic Service Set Identifier |
| CI | … | Carrier per Interference |
| CHI | … | Channel Information |
| CQI | … | Channel Quality Indicator |
| CSV | … | Comma-Separated Values |
| DL | … | Downlink |
| DNN | … | Deep Neural Network |
| DNS | … | Domain Name System |
| DRATE | … | Download Rate |
| EDGE | … | Enhanced Data-Rate for GSM Evolution |
| ES | … | Exponential Smoothing |
| FTP | … | File Transfer Protocol |
| GPR | … | Gaussian Process Regression |
| GPS | … | Global Positioning System |
| GS | … | Gaussian smoothing |
| GSM | … | Global System for Mobile Communication |
| HMAC | … | Hash-Based Message Authentication Code |
| HTML | … | Hypertext Markup Language |
| http | … | HyperText Transfer Protocol |
| ID | … | Identifier |
| IDW | … | Inverse Distance Weighting |
| ILR | … | Luxembourg Institute of Regulation |
| iOS | … | iPhone Operating System |
| IP | … | Internet Protocol |
| JSON | … | JavaScript Object Notation |

| | | |
|---|---|---|
| LAN | … | Local Area Network |
| LI | … | Linear Interpolation |
| LTE | … | Long-Term Evolution |
| MAC | … | Media Access Control |
| MAE | … | Mean Absolute Error |
| MIMO | … | Multiple Input Multiple Output |
| MMS | … | Multimedia Messaging Service |
| MS | … | Microsoft |
| MSE | … | Mean Square Error |
| QoS | … | Quality of Service |
| RATEL | … | Regulatory Agency for Electronic Communications and Postal Services from Serbia |
| RF | … | Random Forest |
| RMBT | … | RTR Multithreaded Broadband Test |
| RMSE | …. | Root Mean Square Error |
| RSRP | … | Reference Signal Received Power |
| RSSI | … | Received Signal Strength Indicator |
| RTR | … | Austrian Regulator Authority for Broadcasting and Telecommunications |
| PCHI | … | Packet Channel Information |
| pdf | … | Probability Density Function |
| SIM | … | Subscriber Identity Module |
| SMS | … | Short Message Service |
| SSID | … | Service Set Identifier |
| TCP | … | Transmission Control Protocol |
| TLS | … | Transport Layer Security |
| TX | … | Transmitter |
| UDP | … | User Datagram Protocol |
| UL | … | Uplink |
| UMTS | … | Universal Mobile Telecommunications System |
| USB | … | Universal Serial Bus |
| UTC | … | Coordinated Universal Time |
| VIX | … | Vienna Internet eXchange |
| VoIP | … | Voice over IP |

| | | |
|---|---|---|
| WLAN | … | Wireless LAN |
| XML | … | Extensible Markup Language |
| 1D, 2D, 3D | … | 1, 2, 3 Dimensions |
| 2G, 3G, 4G | … | 2nd, 3rd, 4th Generation |

**SYMBOLS:**

| | | |
|---|---|---|
| $\alpha$, $\beta$, $\sigma$ | … | Alpha, Beta, Sigma (Parameter) |
| $cd$ | … | Cumulative Distance |
| dB | … | DeciBel |
| dBm | … | DeciBels below 1 Milliwatt |
| Ɛ | … | Statistically Independent Error |
| $f(x)$ | … | Function of $x$ |
| km | … | Kilometre |
| Mbit | … | Mega bit |
| Mbps | … | Megabits per Second |
| $m(x)$ | … | Mean of the Function |
| m | … | Meter |
| $m^2$ | … | Meter Squared |
| $p$ | … | Power (Parameter) |
| s | … | Second |
| $s_i$ | … | Predicted Value |
| $w$ | … | Weight |
| $x$ | … | Variable |

# LIST OF FIGURES

# LIST OF TABLES

# INTRODUCTION

With the increasing demands on the wireless networks, aspecially their reliability, responsiveness and performance, it is crucial for the network operators to quickly and reliably establish the connection of the user's device to the network, without needlessly overloading it. This Master's Thesis presents the basics for the machine learning solution utilizing several regression methods, which estimate the initial network parameters of the user's device and create maps of the estimated behaviour for the network performance parameters.

The thesis briefly introduces the topic of big data, history of artificial intelligence and machine learning. The basic parameters evaluated when describing big sets of data are explained and techniques utilized while working with large databases are listed.

The overview of the whole procedure leading to the creation of maps of the estimated mobile network's parameters is described as well as the main challenges arisen from the available data, including coverage of measurements, changes in network traffic or user's tariff limitation.

Also, the thesis includes the description of the interface, functionality and concept of the mobile application RTR NetTest (netztest), which serves as the main source of input data for the network's parameters estimation. First, the application's operability and user interface are described. Then the RMBT testing process is presented and the visualization possibilities on the RTR's website are depicted in detail. Based on the previous information, the area of measurements is divided into several categories. The open-data available on the website from all measurements are being presented in detail and the second source of input data is being explained.

A thorough analysis of several regression methods is complemented by their theoretical background. A set of evaluation methods utilizing several metrics such as mean average error are used to assess the accuracy of each regression as well as their complexity. After their thorough 1D analysis, inverse distance weighting and Gaussian process regression are analysed in 2D and are used to create a set of estimation maps of network parameters in the locations for which there are no reference measurements. Their accuracy is evaluated from the point of view of the performed reference measurements using NetTest application and the analysis of the statistical distribution of the data.

The summary evaluates the information presented in the thesis, pinpoints the findings and sums up the objective. Also, it offers a foundation for the possible extension of the work.

# 1 MACHINE LEARNING & BIG DATA

In this chapter, a theoretical framework for the used data science methods is discussed. Differences between different machine learning algorithms are outlined and their suitability for the tasks involved is presented complemented by introduction to the topic of big data.

## AI & MACHINE LEARNING

One of the most discussed and developed topics within scientific community as well as among public is artificial intelligence (AI) and the ways to utilize it for our benefit. The topic fascinates and scares everyone who starts to think about it, whether humanity is capable of controlling the machines more intelligent than humans or whether the machines will rise against us, as is the main topic of many popular movies. Luckily, there is no need to create a complex-thinking AI to utilize the benefits it offers. Machine learning, a sub-field of AI, presents itself as a safe, reliable and tested approach for problem-solving in wide variety of industrial fields. The concept of machine learning has been developed in the 1970's, but the utilization lacked computer power [1]. Over the past decades, the machine learning was enabled because the performance and computation capabilities of computers advanced to the state, where complex calculations are being done quickly, reliably and without investing large sums of money into hardware. In the 1990's the first applications were created, that not only analysed the data, but learned from them by drawing conclusions [2]. In 1997, the AI called Deep Blue from IBM company astonished the public, when it managed to defeat Garry Kasparov, the world champion, in six-game match of chess [3]. Today, countless facial recognition software tools, autonomous search engines, sorting algorithms and bots playing computer games better than any human [4] are being developed to gain knowledge and skills, that for human alone would be impossible.

Machine learning is a special kind of artificial intelligence, which processes a set of data to acquire results and then uses these results to improve its own algorithm. Its main benefits include relatively low computation complexity and direct problem assignment, and therefore is utilized for optimization purposes across all industries. Machine learning techniques enable performance prediction, failure pre-detection when utilized as a diagnostic tool and offer improvement of workload distribution. For this thesis, some machine learning algorithms are considered as a tool for performance prediction of user device in the location with unknown network parameters.

### SUPERVISED LEARNING

Algorithms which utilize data with their corresponding labels, also called truth values, are categorized as supervised learning methods. A supervised learning algorithm accepts a set of data called the *training* data as input and produces a model which can be used for classifying new examples as output. After the training process, data without labels can be input into the algorithm which returns the labels as output. Ideally, the algorithm would be able to generalize and interpolate reasonably even unseen data. The problem of finding a compromise between adjusting the data too specifically based on trained data and predicting very generally is called the bias-variance trade-off. [5]

An algorithm has high bias when it perfected its accuracy for the data it was trained on, yet when given different data set, it scores poorly. An opposite, equally undesired phenomena is high variance. It occurs when the algorithm makes too general guesses. The balance between variance and bias can be solved by providing an adjustable parameter or choosing an algorithm which finds the balance itself. Complexity of the classification function is also to be considered. Simple function may be more suitable for usages with lower amounts of data whereas a more complex function requires a sizeable data set while delivering lower variance.

For example, the authors of [6] created a methodology to label locations using location-based social media photos. The algorithm utilizes several machine learning techniques combined with neural network system to choose representative photos for each point on the map. The classification is then realized by chosen group of people, combining crowdsourcing with machine learning algorithms with human-factor decision making.

Neural networks have seen a rapid development in the recent years and have enabled whole new economies to emerge. Their subset, Deep Neural Networks (DNN) can be used to extract multiple layers of information from raw input and are commonly used by companies like Tesla in autonomous cars software. Although autonomous cars are seeing fast advancements and are predicted to become common soon, even a massive dataset of approximately billion hours of video footage taken from Tesla cars cannot make it universally safe and reliable in unknown roads. An initiative named DeepTest has been successful to systematically test DNN-driven vehicle software detecting erroneous behaviours that could potentially lead to fatalities using automatically generated test cases simulating real world conditions like rain, fog and different lighting conditions. [7]

For this thesis, some machine learning algorithms are considered as a tool for performance prediction of user device in the location with unknown network parameters.

### UNSUPERVISED LEARNING

Also called clustering or cluster analysis is a group of techniques which aim to find the relations within the data without the training part discussed above. Real world data often come without labels, but machine learning is still able to extract valuable information from it. The aim of clustering is to process the data while finding natural clusters. Clusters should be sets of points which are similar to each point within their cluster and dissimilar to points from different clusters. Their similarity is determined by a similarity measure which has to be chosen appropriately to the problem at hand. An important feature of cluster analysis is its capability to discover patterns in data without providing explicitly what the algorithm should distinguish.

Anomaly detection is an exemplary application of such method. A density-based clustering method has been found effective in network intrusion detection. It’s main advantage over other machine learning approaches such as neural networks and support vector machine was the ability to discover new, previously unknown attacks on the network. [8] The team utilized a combination of density-based and grid-based high dimensional clustering called fpMAFIA and was found to be 95% successful in covering data with appropriate values of parameters and was proven to scale linearly with the size of the input data.

# BIG DATA

Discussed machine learning have been successfully used performing complex tasks. It wouldn’t be the case if the input data sets were not appropriately large and diverse. Big data together with advancement in computational processing power are the reason machine learning is universally successful.

The world’s capacity to store information is increasing by 23% every year while it’s general-purpose computing capacity is growing by 58% annually. The amount of data stored in 2007 was $2.9 \times 10^{20}$ bytes [9]. It is predicted, that by 2020, 1.7 megabytes of new information will be created every second per person [10] and that around a third of all data will be processed through the cloud. Sadly, nowadays less than 0.5 % of available data is being analysed. This fact is caused by insufficiently effective and inaccessible computational methods for large-scale data processing, unwillingness of companies to invest into often costly data-analysis tools and lack of experts in the field. Yet, the situation is starting to change.

The storable data capacity should come in pair with the ability to process it and extract valuable information from it. Companies like Netflix, Amazon, Google and Facebook were the pioneers in analysing such immense amounts of data which naturally meant they had to develop their own technologies to process it. Google originally came with the MapReduce approach which allowed it to process data in distributed and parallel manner, which later was reimplemented and open-sourced by Apache [11]. In order to filter the noise and take advantage of it fully, the desired characteristics for the datasets have been clearly defined.

## BIG DATA - CHARACTERISTICS

Each set of big data can be reassured by the three basic “V” characteristics [12]. First one is “Volume”, which represents the size of the data, which presents itself as one of the leading challenges for storage and processing. “Variety” refers to the spread of distinct formats of the data, which may come from many different sources in many different forms. The presented challenge is to structure or organize the data in a logical way. “Velocity” describes the speed, at which the data can be analysed and reviewed. This also addresses the rate of change of the data, whether by acquiring the new data or by dumping the outdated ones. Additionally, another “V’s” can be added depending on the requirements. “Value” represents the usefulness of the data corresponding to decision making of the algorithm, “Veracity” refers to the trustworthiness of the data sources, “Volatility” refers to the time period for which the data is valid, “Validity” refers to the accuracy of the data for reaching the desired goal and “Variability” refers to the differences in data structures caused by different origins of the data. The last measure for big data is “Complexity” referring to the volume of effect on the system due to the small change in data due to inner linking in the data [13] [14].

There is a number of commonly used techniques to reduce the complexity of working with large databases of data by reducing the computation times or speeding filtering algorithms [15]. The first technique is “Association”. By finding links and correlations within the data, it is possible to predict whether and how that data will be used. It may also be valuable to store the associated data together. “Classification” technique sorts the data into categories or groups depending on the desired parameter (type, size, time) to speed up the searching process in the future. “Generic algorithms” are non-specific applications, that pre-process the data or dump the corrupted files. “Machine learning” software allows to draw conclusions from own results and optimize own decision-making capabilities. “Regression analysis” analyses the data depending on some of its aspects, draws conclusions and creates predictions based on the difference in variables based on the aspect’s change. “Sentiment analysis” optimizes the system based on the feedback form the user. “Social network” analysis sorts the results into nodes and creates ties between the correlated ones. There are many platforms available for the big-data storage, mining, processing and analysis, yet for the purpose of this thesis the data are acquired from open-data applications (RTR NetTest) and are being stored locally.

## BIG DATA - STORAGE OPTIONS

For data storage and processing, most companies have to choose between two primary options that are widely available. These are own servers and cloud computing option. The choice between them is depending on the specific application. The most impactful aspects are operation and acquisition costs, level to which the company is willing to invest into the new technology and the value obtained.

In case of the hardware solution, the choice of platform, system and storage/processing data format are fully up to the project crew. The costs of this solution mainly include power consumption and maintenance. The scalability depends on the chosen system and architecture. Horizontal scaling distributes the workload across many devices (servers, computers, etc.) and has almost infinite scalability, whereas vertical scaling focuses all computation power on a single device by investing into more powerful components [16].

The Cloud solution on the other hand presents a zero-maintenance system on the remote server, from which the user can buy storage space and computing power. Such solutions consist of a storage (Google cloud services, Microsoft Azure, Amazon S3) and several supporting software or frameworks. The most widely available programming framework for Big Data is HADOOP [17]. It is available as an open-source, it supports distributed computing environment which is robust against node failures, as well as it is supported by Big Data Cloud Solutions such as Microsoft Azure or Amazon S3. MapReduce is a programming tool of HADOOP capable of processing large numbers of datasets [11]. There are several MapReduce-based tools, which further improve computing capabilities of the Big Data system. MapReduce is widely applicated in cloud systems since it is highly scalable, open-source solution.

There are several data store types, which are optimized for Big Data storage and analytics, each excelling in different aspect. Document-oriented store is designed to work with larger groups of documents in different formats such as JSON, XML or MS Word. Document oriented stores are also called row formats. Column-oriented formats store the data in columns, sorted in the same fashion, depending on their properties. Graph database stores data with relation to each other in graph-like fashion with nodes, peaks etc. Key-value databases store the data based on the chosen key values. These systems are designed to operate large databases.

# 2 FUNDATION OF THE THESIS

In this chapter the basic overview of this thesis is presented. It includes an extended introduction to the process of determining the coverage maps, followed by the impacting factors of crowdsourced data, that are affecting the final results. Following these initial remarks, the basic theory of depicted regression methods is presented and their applications on the (noisy) sine function are displayed. The definition of the mean average error is provided, since it serves as a basic metric for the evaluation and comparison of the methods. This serves as a baseline for the subsequent chapters.

## 2.1 INITIAL REMARKS

This chapter deals with the basic principals that are presented in this thesis. It gives an overview of the project and presents the most impactful factors of the crowdsourcing data.

### 2.1.1 DIPLOMA THESIS OUTLINE

The main goal of the proposed thesis is to create a tool, that is able to predict the parameters of the network and therefore be used for the network optimisation purposes. Based on previous measurements taken at the locations with known coordinates, the realized tool should establish an accurate prediction of network parameters in the location for which the measurements do not exist in the form of performance map. The objective of this thesis is to predict the network performance parameters of the user's mobile LTE device (the majority of analysis of the utilized methods is carried out on the Reference Signal Received Power (RSRP) measurements due to its independency of tariff limitation).

For this purpose, several methods of regression are used and compared against each other to obtain the most reliable result. The basic theory of the chosen methods is briefly described in the following Chapter 2.2. A number of algorithms, that are based on these various regressions are all implemented in MATLAB.

There are two main sources of input data for the thesis (see Chapter 3). First one is a crowdsource based database containing the reports of RTR-NetTest measurements [18]. It is an Internet connection speed test provided by RTR with open data results. The database with 4G measurement results for the year of 2018 is trimmed to the Vienna city area (to reduce the computation power demands and to ensure as densely measured area as possible) and divided based on the operator. This data set serves as a base for the creation of estimated coverage maps of various network parameters. Also, the RTR-NetTest measurement tool is used to generate self-measured database, which serves as an evaluation matric for the resulting map and additionally as the comparison of regression's behaviour in the open-air area vs. the city area.

The second source of data is a set of 9 measurements of network parameters taken on the route through the centre of Vienna as an example of drive test measurements. These measurements were acquired by using the NEMO system and were provided by TU Vienna. The main utilization of these data is 1D analysis of the various regressions and therefore reducing the regression to 1D problem. The data were acquired on foot, with specialized measurement equipment (NEMO Keysight). The measurement data are in a series, specified by time as well as by GPS coordinates compared to the crowdsource-based database from RTR, which does not have the consistently spaced time intervals between the measurements.

The 1D analysis (see Chapter 4) of chosen regressions, including linear interpolation, exponential smoothing (extended by other smoothing techniques), inverse distance weighting (IDW), random forest and Gaussian process regression (GPR), serves as a baseline to determine the most suitable methods for 2D utilization of the given data regression. The parameters of each method are described and optimized for objectives of this thesis. The methods are compared against each other in terms of their mean average error (MAE) metric (see Chapter 2.3), complexity and computational-demand characteristics.

The further regression evaluation and parameter optimization is realized as the 2D analysis of chosen methods (see Chapter 5) on NEMO measurement data, as well as self-made RTR database. To create the unbiased regressions, the latitude and longitude coordinates of every point were transformed into metres using the adjusted Heaviside formula, since the same difference in latitude and longitude does not create the same difference in metres (see Chapter 5.1). The regression methods chosen for 2D analysis (GPR and IDW) are then shown and compared based on the different parameters.

The distributions of RSRP, downlink throughput and uplink throughput within the considered measurements were shown and compared to the corresponding Gaussian distribution to assess, whether the data distributions are similar to the Gaussian, since GPR is optimized to predict normally distributed data.

Using the error metric, the minimum evenly distributed measurements per area, which ensure minimal margin of error, was derived. The chosen prediction methods are utilized to create performance maps using the data sources utilized in this thesis. The validity of the performance maps is discussed.

The filtered RTR measurement samples cover the area of approximately 8.8524 × 18.6676 km over the central part of Vienna. The database consists of 17322 measurements. 4315 were realized in H3 network, 8176 measurements were realized in T-Mobile network and 4831 measurements in A1 network. All considered measurements are within LTE technology.

The network parameters considered in this thesis are:

Download speed (DL), which is the rate of data volume over time flowing in the direction towards the end device. Downlink throughput, defined as the maximum achievable download speed, is the metric describing the current network state at the given coordinates.

Upload speed (UL) and uplink throughput are defined in the same way as download, only referring to the flow of data out of the end device.

Reference Signal Received Power (RSRP) is the average power measurement value in dBm of the reference signal received by the device, measured over the whole bandwidth.

### 2.1.2 CROWSOURCED DATA IMPACTING FACTORS

Several challenges arise during the data gathering, processing and postprocessing, which may infect the crowdsourced data that are used as a base for creating network's parameters coverage maps. These may come from several sources which may include the measurement of input data, different locations of these measurements and therefore the changing signal strength connected to the diversity in the network's operator coverage and many others. Some of these problems and factors impacting the results together with their brief discussion are presented in this chapter.

**Coverage of measurements** – There is only a limited amount of input data based on the number of tests in the available database. These tests do not cover every location that may be of interest within the scope of this project since the measurements are not evenly distributed in space. Creating a spatial regression requires several reliable measurement results per area to assure correctness of the result, regardless of the regression's type.

**GPS accuracy** – The accuracy of GPS coordinates varies based on both user's location and user's device. The user's location affects GPS accuracy especially at the locations with signal dampening objects, which create shadowing, e.g. high building surroundings, underground or thick walls (indoor positioning problem). The user's device may affect the accuracy in terms of settings (high/low accuracy). Also, the older device models may have less precise positioning systems implemented. The GPS accuracy of the modern devices is approximately 4 m.

**User's Tariff limitation** – The tariff provided by the Internet provider is limited based on what the user wants to pay for. This results into a different data rate speeds across the spectrum of obtained test and needs to be taken into consideration while working with crowdsource data. Since we do not have the information about the user's tariff's terms, we may only assume if the measured speed is lower/higher due to the tariff limitation or due to other conditions.

**User's device limitation** – The device used for the test measurement may be just as limiting as the tariff factor. The device type, age, its operation system and configuration may resolve into limiting the performance and therefore it has to be taken into consideration when working with crowdsource data.

**Changing traffic over time** – The Internet traffic changes depending on the time of a day. To demonstrate the big variety and high slope of changes the Figure 2-1 is presented. It shows the changes of the sum of total data rate of VIX (Vienna Internet eXchange [19]) during the day (Figure 2-1 top) and during a week (Figure 2-1 middle). Therefore, the data (e.g. connection speed) acquired during the busy hours may be devaluated in comparison to the data acquired during the quiet hours. Another factor is the change over one-year intervals. Every year the network load increases (see Figure 2-1 bottom). Therefore, the data acquired long time ago should not be randomly mixed with the newest data without taking this fact into consideration. Also, this factor may be a problem when evaluating the precision of the created coverage map, especially in case the data used for it are remote in time (e.g. 1-2 years old) and are being compared to newly measured data. (Within the Figure 2-1, all times are in UTC, dated for year 2018.)

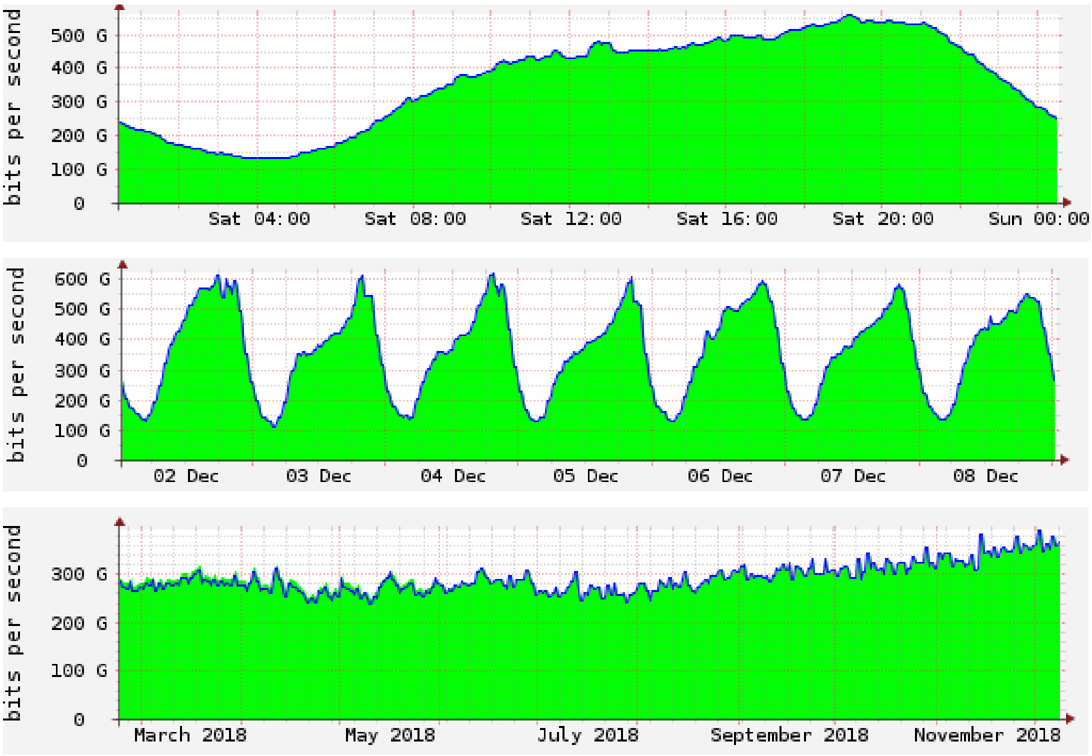


Figure 2-1 Traffic Over Time (Top - Daily Traffic, Middle - Weekly Traffic, Bottom - Yearly Traffic) [19]

**Bandwidth utilisation of each Internet provider** – Every Internet provider uses its spectrum in a different way depending on their priorities. Therefore, the measurements may be affected and may vary from provider to provider. To assure the accuracy of the coverage map, the providers should be treated separately, the input data should be divided based on the operator instead of merging all operators into one data input.

**Signal coverage across different geographic locations** – As explained in Chapter 3, the geographical structure of Austria is more complicated in comparison with some other countries (e.g. Czech Republic), therefore it is not covered entirely. For those locations, where the signal strength may be weak or not available at all, there are only a few measurements, or no measurements exist there.

**User Location while taking the test** – may be yet another factor devaluating the result of a measurement. If the user is in a space, where the signal strength is blocked (e.g. underground, lift) the result of the test's measurement may be less reliable due to this factor. To get the best test results, the open space with no signal blocking is needed, but this is not a demand that is always possible to keep.

**The technology used** (EDGE/GSM, UMTS, LTE) – With a different technology comes a different speed of transmission. This factor can be taken into account quite easily as the statistics of how many devices use what technology are available and as each test carries an information about the used technology. To make the process simpler only the LTE measurements are taken into account. On the other hand, using only LTE measurements reduces the number of available input data.

**Connection type** - In case of using LAN during the performance of test, which is used to obtain the input data, the quality of the LAN router needs to be taken into consideration as well as when using WLAN. In case of WLAN the distance between the measuring device and the router needs to be considered as well. Since there is no information about this within the test results, this becomes to be one of the reasons why only the mobile LTE based measurements are used for finding the solution to coverage maps.

**Reason why the test was taken** – RTR NetTest is a tool for evaluation network parameters which is not connected to any other application and has to be accessed individually or must be running on loop, which utilizes large amounts of data (e.g. Tutela implements their network parameters measuring software on background of various applications and games to gain more samples over time without user participation, their test takes less amount of data than RTR [20]). Therefore, the users utilizing the RTR test are doing so after receiving an impulse for such action. There are various reasons why a user utilizes the connectivity test, sometimes it is to gain information, but most frequently to confirm the bad or decaying connectivity. Another reason to take the test may be to evaluate the network or new devices on the side of the operator (e.g. NetTest is used by operators as a tool for drive tests, which may have a better connectivity etc. than common user's connectivity services). All these tests are saved within one database in the same fashion.

## 2.2 REGRESIONS – BASIC THEORY

In this chapter, the regression methods used within the scope of this thesis are introduced and the basic theory used is explained. For better visualization and clarification of various parameters, the implementation of these regression methods is presented on a sine function.

### 2.2.1 LINEAR INTERPOLATION

The linear interpolation (LI) is one of the simplest interpolation methods in terms of its implementation difficulty and complexity. On the other hand, it may derive worse results in terms of its accuracy in comparison to some more sophisticated methods, in cases where the trend is non-linear. The results of this interpolation method may be sufficient in some implementations, especially in cases where precise function values are of less importance than the computation demands.

It is based on establishing unknown value points from the set of known points. The new points are derived geometrically by connecting two adjacent points in plane by the straight line. All points, but the original ones, realizing this connection line are therefore the interpolated points. The interpolation of the set of data is then considered as a linear interpolation of each two nearest points in plane, assuming the function connecting the neighbouring points is purely linear. See Figure 2-2, presenting a simple example of linear interpolation applied to one period of sinus function (the range from 0 to $2\pi$ (approximately 6.283)). The interpolation is realized by selecting a value (sample) at every whole number and applying the interpolation function to them.

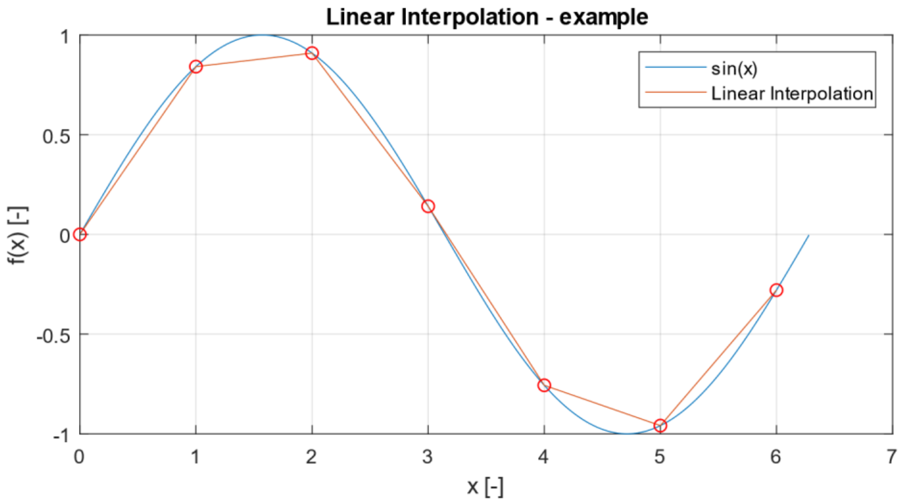


Figure 2-2 Linear Interpolation of Sine Function

This method it therefore suitable only for application with low demands on accuracy, especially in cases where the slope of input function is changing rapidly (faster than its “sampling”), for example in the cases where the input data is changing dynamically around its mean or for the functions where the peak values are important. The basic formula for deriving the value *f(x)* of linear interpolation at desired point *x* is shown in Equation 2.1 [21]:

$$f(x) = \frac{x - x_1}{x_2 - x_1} \cdot f(x_2) + \frac{x_2 - x}{x_2 - x_1} \cdot f(x_1) \tag{2.1}$$

where $f(x)$ is the function value at point $x$, and $x_1$ and $x_2$ are the previously known points.

Based on the definition of linear interpolation, it is clear that this method is meant for one dimensional application. In case of two spatial dimensions, linear interpolation becomes a bilinear interpolation. The base of this method is to apply the linear (one dimensional) interpolation first in one direction and after that to apply it in the other direction. Therefore, the outcome of the method is not linear, but the product of two linear interpolations [22].

The use of this method was first documented in 300 BC and was used over the whole known history e.g. in astronomy and mathematics. Later, in 20th century, the method was implemented and used in computer graphics [23].

### 2.2.2 EXPONENTIAL SMOOTHING

Exponential smoothing (ES) [24] is a technique for smoothing time series sample-by-sample, which was first developed in 1957 by C. E. Holt and widely used ever since. Compared to the other regressions, exponential smoothing does not create new data points. The algorithm weights past observations with decreasing weight over time and, based on the chosen parameters, assigns the new point as a combination of new value and the value from the past. The name exponential refers to the exponentially decreasing weight of the sample points from the past on the current sample. This technique is utilized to neglect high-frequency noise, meaning it can be used as a first-order impulse response low-pass filter. There are several implementations of exponential smoothing in practical use.

The simplest implementation of exponential smoothing is called exponential moving average and can be expressed as:

$$s_0 = x_0$$
$$s_i = \alpha \cdot x_i + (1-\alpha) \cdot s_{i-1} \tag{2.2}$$

Where $s_i$ is a predicted value and $x_i$ is the measured value at time $i$, $\alpha$ is the smoothing parameter from the interval $\langle 0; 1 \rangle$. The smaller the $\alpha$, the smaller is the weight of the new sample. In Figure 2-3, simple exponential smoothing of noisy sinus function with two different values of alpha are shown. The figure shows, that for $\alpha = 0.2$ the trend is smoothed, and the noise is supressed, but the signal is delayed. With $\alpha = 0.8$, the smoothing and the noise cancelation is visibly smaller (signal copies the original more reliably), but so is the delay. Therefore, the utilization of this method is possible in applications, where the delay is acceptable, otherwise it is necessary to find similar method of noise cancelation, which does not cause the delay.

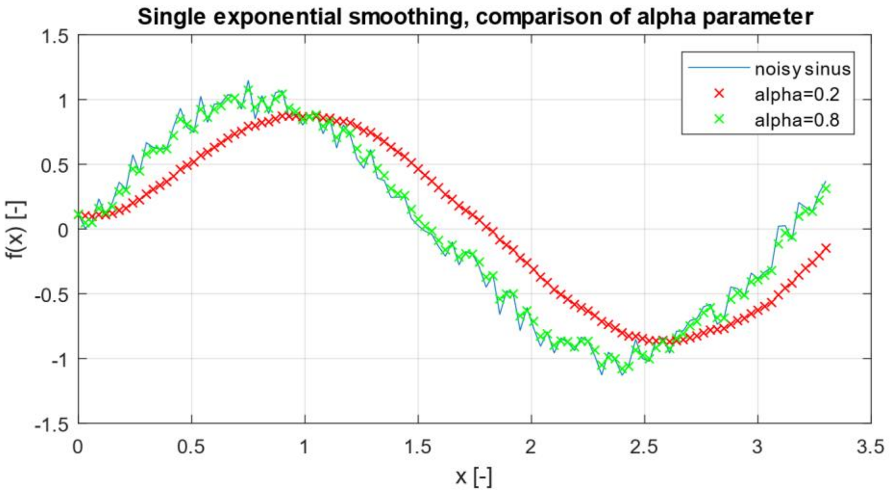


Figure 2-3 Single Exponential Smoothing of Sine Function

More advanced exponential smoothening method, called double exponential smoothening is used to neglect the undesired trend in data. This technique applies the filter on the data and then once again on itself with secondary parameter $\beta$. The previous equation changes to Equation 2.3 and 2.4:

$$s_0 = x_0,$$
$$s_1 = x_1, \qquad b_1 = x_1 - x_0$$
$$s_i = \alpha \cdot x_i + (1-\alpha) \cdot (s_{i-1} + b_{i-1}) \tag{2.3}$$
$$b_i = \beta \cdot (s_i - s_{i-1}) + (1-\beta) \cdot b_{i-1} \tag{2.4}$$

where $s_i$ estimates the new value and $b_i$ estimates the trend of the data. $\alpha$ and $\beta$ are the weighting parameters of the regression, both from the interval $\langle 0; 1 \rangle$. Figure 2-4 shows the comparison of single exponential smoothing with parameter $\alpha = 0.2$ and double exponential smoothing with parameters $\alpha = 0.2$, $\beta = 0.8$ applied to the noisy sine signal. The figure shows, that the time delay is neglected thanks to the secondary smoothening application. On the other hand, the resulting signal is less smooth.

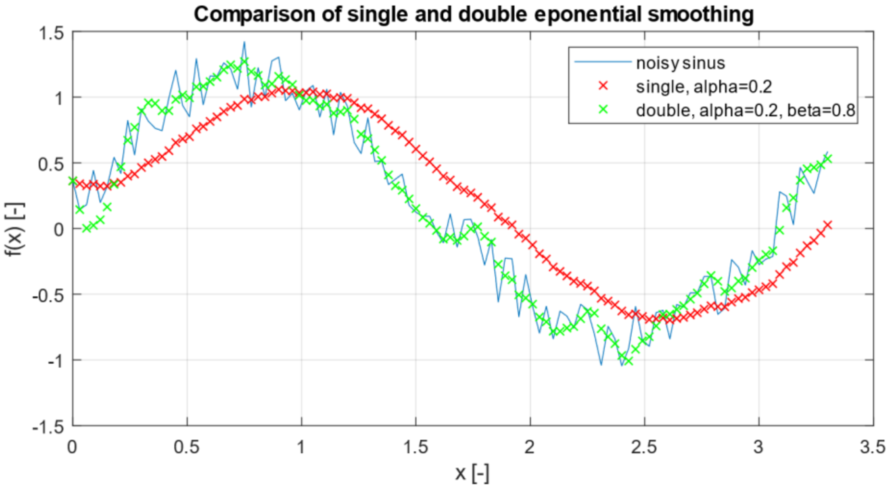


Figure 2-4 Comparison of Single and Double Exponential Smoothing

### 2.2.3 SYMMETRIC SMOOTHING METHODS

There is a number of other methods to smooth the data, which compensate for the exponential smoothening's downside of considering only the left side of the data (past results). Such methods operate with smoothing kernel moving across the data, calculating weighted mean based on the kernel shape and window size. It is important to remember, that the window is centred around the considered sample only if its size is odd.

The basic and frequently utilized solution is windowed moving average, for which the kernel function has constant height over the whole window, resulting in calculated mean over several neighbouring samples for the considered sample. The simple formula for odd window sized moving average per sample is:

$$s_i = \frac{1}{N}\sum_{k=0}^{N} x_{i-\frac{N-1}{2}+k} \tag{2.5}$$

Where $N$ is the window size, $s_i$ is the prediction sample and $x_i$ is the current original sample. Another implementation for smoothing is triangular kernel smoothing utilizing triangle-shaped window, Gaussian kernel smoothing or smoothing using Savitzky-Golay filter. The effect of window size on the resulting smoothing for moving average is shown in Figure 2-5 (left), where it is visible that with the increased window size, the estimation is smoother, yet on the other hand the peaks are significantly reduced. By comparing the graph to exponential smoothing results, there is no shift of samples within those methods. The comparison of moving average, smoothing using Gaussian kernel and smoothing using Savitzky-Golay filter is shown in Figure 2-5 (right), where the window size was set to 19 samples to minimize both noise and peak reduction (see Chapter 2.3). Savitzky-Golay approximation reacts more dynamically to the changes within the data (when the noise affects several subsequent samples in the same way) than both Gaussian and moving average.

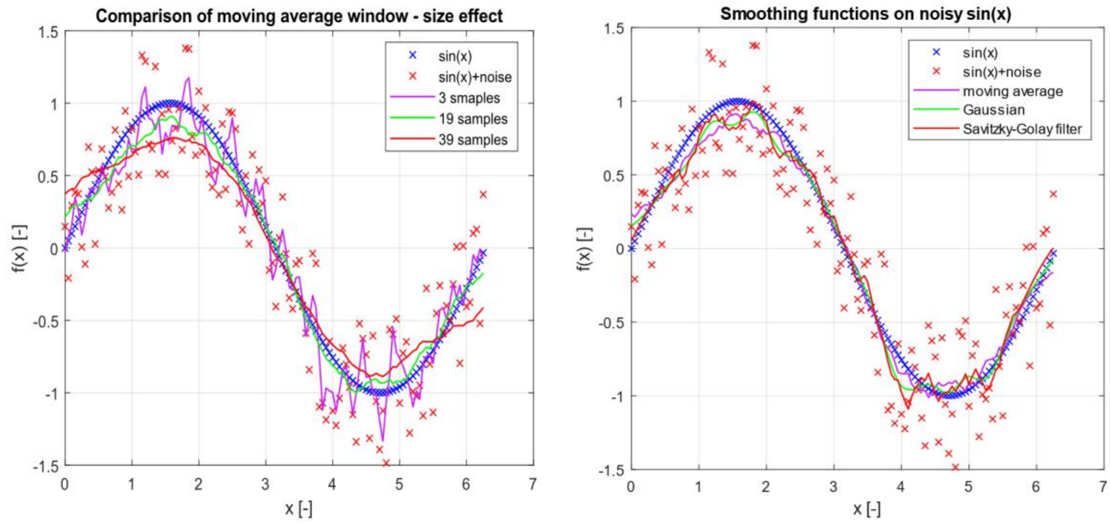


Figure 2-5 Comparison of Smoothing functions - Sine Example

### 2.2.4 INVERSE DISTANCE WEIGHTING

Inverse distance weighted (IDW) interpolation is a regression method applicable in multidimensional space, which determines cell values using an inversely weighted combination of a set of sample points. The surface being interpolated should be that of a location dependent variable. The weight is a function of inverse distance raised to the mathematical power, thus reference data closest to the considered point in grid have the highest impact on the resulting value. The algorithm includes option of choosing the maximum radius above which it stops considering the neighbouring cells. [25] [26] The basic formula for IDW is shown in Equation 2.6:

$$z_i = \frac{\sum_{i=1}^{n} z_k \cdot \left(\frac{1}{w_{i,k}}\right)^p}{\sum_{i=1}^{n} \left(\frac{1}{w_{i,k}}\right)^p} \tag{2.6}$$

$$w_{i,k} = |x_i - x_k|$$

Where $z_i$ is the desired quantity approximation at coordinates $x_i$, $z_k$ is the quantity of k$^{\text{th}}$ reference symbol at coordinates $x_k$, $w_{i,k}$ is the Euclidian distance between $x_i$ and $x_k$ and $p$ is the power parameter.

Determining the optimal values for each parameter is depending on the application IDW is used for, since it is not based on any physical principle. Later in this thesis, radius and power parameter are optimized in 1D to match the behaviour of the trend. The optimization algorithm is chosen in a way to minimize the mean absolute error of the regression.

One downside of this technique is that at reference points, the regression returns the value equal to the value of that reference point (zero distance equals infinite weight), therefore in case of having several reference points affected by noise in close proximity, the regression will fluctuate between those points and will not return reliable output. Therefore, before applying IDW into spatial coordinates, the data will first be filtered for noise suppression using a different technique.

Choosing the appropriate power parameter is bread and butter of this technique, from which the two extreme situations can arise. When power parameter is too high, only the closest point "attracts" the regression curve while all other values are significantly smaller due to inverse weight to the power of "high number". When power parameter is too small, the further points "drag" the regression to the mean value between the closest data points. Figure 2-6 presents two such cases. When approximating the sinus function, the high-power parameter value (*p* = 20) causes the regression to hold its value around every reference point. If the parameter is small (*p* = 1), the value drops between the reference points heavily towards the mean. The balanced choice of power parameter (*p* = 2) keeps the approximation between the reference points smoother and less drawn to the function's mean.

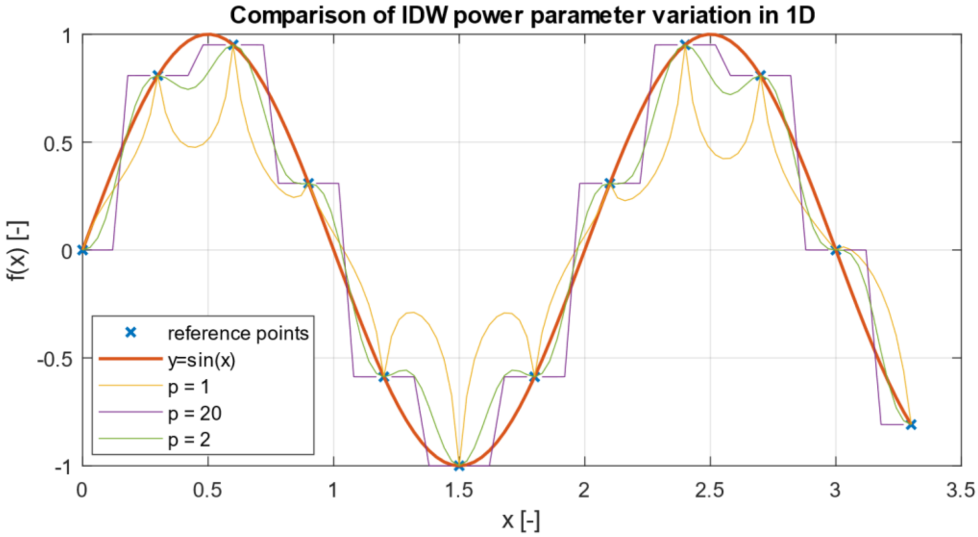


Figure 2-6 IDW of Sine Function

If the radius parameter is chosen to be infinite, in relatively distant locations from closest reference points (based on power parameter chosen), the value of each point will approach the mean value of all reference points.

In practice, IDW is utilized in a variety of scientific fields varying from geostatic predictions to applications in computer science. [27] utilizes IDW as a rainfall distribution prediction method. The method has been evaluated as suitable with high (0.95) correlation coefficient values. [28] utilizes IDW for the array of 3D imaging sensors with adaptive power parameter, significantly improving the interpolation accuracy of the system.

### 2.2.5 RANDOM FOREST

Random forest (RF) is a supervised machine learning algorithm utilized as both classification and regression problems, which builds a "forest" of decision trees and from such forest chooses the most probable outcome. Each decision tree works as a classifier, independently solving tasks for each regression point and then from the set of results, the method chooses the most probable one [29] [30] [31].

The decision tree works in such fashion, that it creates a tree-like mesh with the best attributes at the root of the tree and by analysing all training questions (reference point decisions), it creates leaf nodes in all branches of the tree. With the increasing number of the trees the regression gets more accurate, but also more demanding in terms of computation power. On the other hand, it is proven that the accuracy stops increasing after a certain threshold of trees is implemented, varying for each specific problem that the regression is applied to. See Figure 2-7, where an example of random forest algorithm is applied to a sine function, while changing the number of decision trees (126 samples are used as a training data, grid divided into 126 equidistantly distributed points). The difference between the regression based on 1 and 10 trees is significantly higher than the difference between 10 and 100 trees. (For more complex functions the number of trees would need to be higher than for sine function which serves as a simple example.)

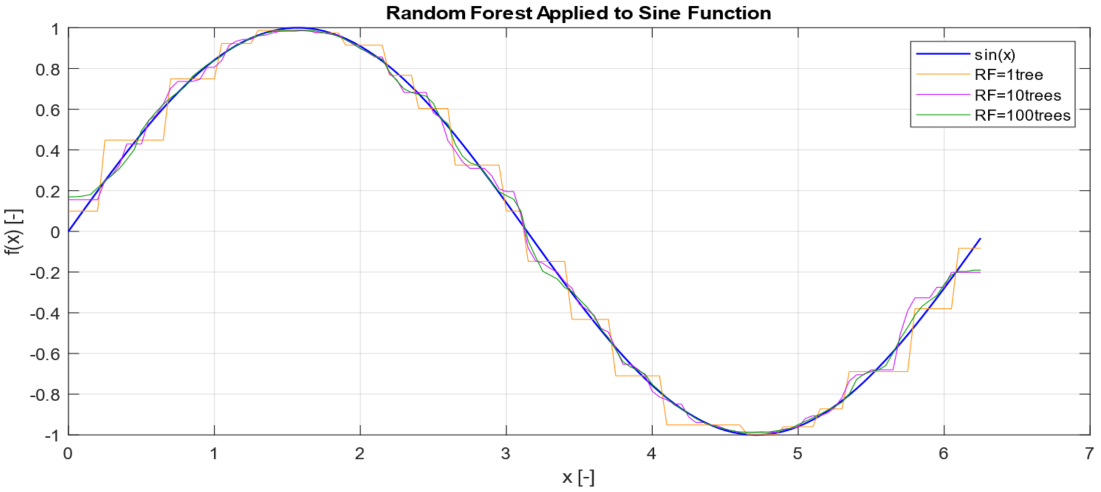


Figure 2-7 Sine Example of Random Forest (1, 10 and 100 trees)

The most influential parameter of the regression is the number of training points. See Figure 2-8, where an example of random forest algorithm is applied to a sine function, while changing the number of input training data (100 %, 50 %, 30 % and 20 % of samples, grid divided into 126 equidistantly distributed points). The difference between the regression based on 20 % and 30 % of training data is significantly higher than the difference between 50 % and 100 %. While the smaller number of trees (in Figure 2-7 only 1 tree) does not change the regression significantly, the smaller number of input training data derives from the input (sine) function significantly (see 20 % line in Figure 2-8).

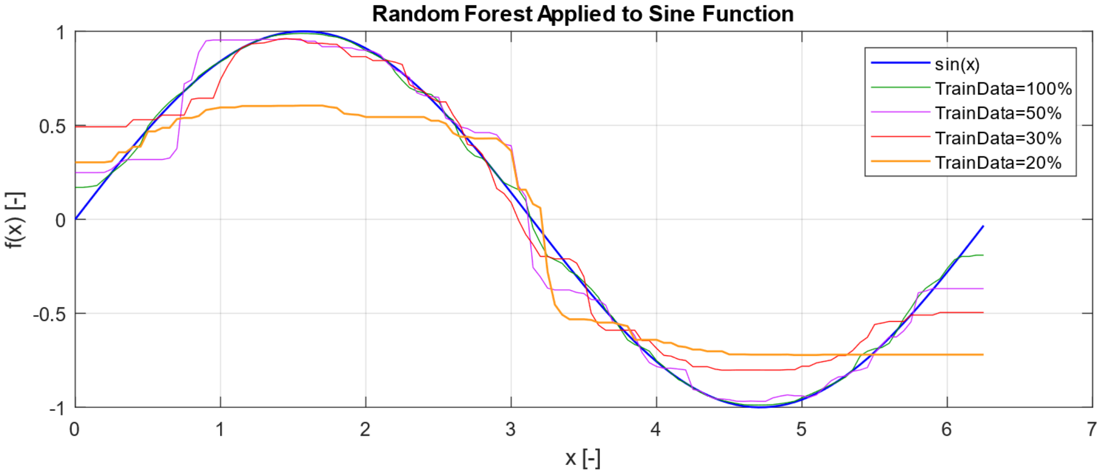


Figure 2-8 Sine Example of Random Forest (100%, 50%, 30% and 20% of training data)

### 2.2.6 GAUSSIAN PROCESS REGRESSION

The Gaussian process regression (GPR) is a widely used technique used in machine learning with highly positive results. The objective is to assign probability density functions (pdf) with Gaussian distribution to characterize the behaviour of the examined quantity. The process can be described as calculation of the unknown point from the training data based on their similarity. The calculations become much less complex thanks to Gaussian probability function’s characteristic as it turns out. To describe the machine learning process using Gaussian regression, first the Gaussian process must be modelled [32] [33].

Let’s assume, that the function $f(\boldsymbol{x})$ behaves as a Gaussian process and $\boldsymbol{x}$ refers to the vector of variables. Function $f(\boldsymbol{x})$ can also be expressed using a vector of weights $\boldsymbol{w}$ as $f(\boldsymbol{x}) = \boldsymbol{x}^T \cdot \boldsymbol{w}$. The mean $\boldsymbol{m}(\boldsymbol{x})$ of the function can be expressed as

$$\boldsymbol{m}(\boldsymbol{x}) = \boldsymbol{E}\{\boldsymbol{f}(\boldsymbol{x})\}, \tag{2.7}$$

Where $\boldsymbol{E}\{f(\boldsymbol{x})\}$ is the expectation of $f(\boldsymbol{x})$. The covariance function of the variable $\boldsymbol{x}$ has the form of

$$\boldsymbol{k}(\boldsymbol{x},\boldsymbol{x}') = \boldsymbol{E}\{(\boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{m}(\boldsymbol{x}))(\boldsymbol{f}(\boldsymbol{x}') - \boldsymbol{m}(\boldsymbol{x}'))\}, \tag{2.8}$$

Where $\boldsymbol{x}'$ refers to the transpose of $\boldsymbol{x}$.and the Gaussian process can then be written as

$$\boldsymbol{f}(\boldsymbol{x}) \sim \mathcal{GP}(\boldsymbol{m}(\boldsymbol{x}), \boldsymbol{k}(\boldsymbol{x},\boldsymbol{x}')). \tag{2.9}$$

The joint probability density function of $N$ random variables $f(x_i), i = 1, \dots, N$, combined into random vector $\boldsymbol{f} = (f(x_1) \dots f(x_N))$ gives joint probability density function in form of

$$\boldsymbol{p}(\boldsymbol{f}) = \frac{\boldsymbol{1}}{\sqrt{(\boldsymbol{2} \cdot \boldsymbol{\pi})^{\boldsymbol{N}} \cdot \mathbf{det}(\boldsymbol{C})}} \cdot \boldsymbol{e}^{-\frac{1}{2}(\boldsymbol{f}-\boldsymbol{m})^{\boldsymbol{T}} \cdot \boldsymbol{C}^{-1} \cdot (\boldsymbol{f}-\boldsymbol{m})}, \tag{2.10}$$

where $det(\boldsymbol{C})$ stands for determinant of the matrix $\boldsymbol{C}$, $\boldsymbol{C}^{-1}$stands for inverse of the matrix $\boldsymbol{C}$. Mean $\boldsymbol{m}$ can be expressed as:

$$\boldsymbol{m} = \boldsymbol{E}\{\boldsymbol{f}\} = \left(\boldsymbol{m}(\boldsymbol{x_1}) \dots \boldsymbol{m}(\boldsymbol{x_n})\right)^{\boldsymbol{T}}, \tag{2.11}$$

and

$$\boldsymbol{C} = \boldsymbol{cov}\{\boldsymbol{f}\} = \begin{bmatrix} \boldsymbol{k}(\boldsymbol{x_1}, \boldsymbol{x_1}) & \dots & \boldsymbol{k}(\boldsymbol{x_1}, \boldsymbol{x_N}) \\ \dots & \dots & \dots \\ \boldsymbol{k}(\boldsymbol{x_N}, \boldsymbol{x_1}) & \dots & \boldsymbol{k}(\boldsymbol{x_N}, \boldsymbol{x_N}) \end{bmatrix}. \tag{2.12}$$

The above can be written in short as

$$\boldsymbol{f} \sim \boldsymbol{\mathcal{N}}(\boldsymbol{m}, \boldsymbol{C}). \tag{2.13}$$

The probability distribution described above sets basic requirements for the variable behaviour in Gaussian regression method.

To realize a Gaussian process regression, training data are needed to create a “knowledge basis” for the system. Training data consist of measured (noisy) parameters at known locations with unknown, statistically independent error $\boldsymbol{\varepsilon}$. That error has (assumed) zero mean, variance $\sigma_e^2$ and can be written as

$$\boldsymbol{\varepsilon} \sim \boldsymbol{\mathcal{N}}(\boldsymbol{0}, \boldsymbol{\sigma_e^2} \cdot \boldsymbol{I_N}). \tag{2.14}$$

$\boldsymbol{I_N}$ stand for identity matrix of dimension N. Therefore, it can be written, that

$$\boldsymbol{y} = \boldsymbol{f} + \boldsymbol{\varepsilon} \tag{2.15}$$

and the training data set $\mathcal{D}$ can be expressed as

$$\boldsymbol{\mathcal{D}} = \{(\boldsymbol{x_i}, \boldsymbol{y_i}), \boldsymbol{i} = \boldsymbol{1}, \dots, \boldsymbol{N}\} = (\boldsymbol{X}, \boldsymbol{y}), \tag{2.16}$$

where

$$\boldsymbol{X} = (\boldsymbol{x_1}, \dots, \boldsymbol{x_N}). \tag{2.17}$$

We also define likelihood, as the pdf (probability density function) of the event factored over the results of the training set as

$$\boldsymbol{p}(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{w}) = \boldsymbol{\mathcal{N}}(\boldsymbol{X^T} \cdot \boldsymbol{w}, \boldsymbol{\sigma_e^2} \cdot \boldsymbol{I_N}), \tag{2.18}$$

Where $p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{w})$ stands for the conditional probability density function and $\boldsymbol{X}^T$stand for transposed matrix $\boldsymbol{X}$. The weight vector has the Gaussian distribution $\boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{C}_w)$, where $\boldsymbol{C}_w$ is the covariance matrix of the weights. For the prediction, the value of $f(x_*)$ will be calculated at the location $x_*$ using the training data set $\mathcal{D}$. Using Bayesian interference theory, it is possible to create posterior (or predictive) pdf based on Bayes’ rule [34]. The Gaussian posterior is a joint pdf with mean

$$\overline{\boldsymbol{w}} = \frac{\boldsymbol{1}}{\boldsymbol{\sigma_e^2}} \cdot \boldsymbol{x_*^T} \cdot \boldsymbol{A^{-1}} \cdot \boldsymbol{X} \cdot \boldsymbol{y} \tag{2.19}$$

and covariance matrix $\boldsymbol{A}^{-1}$, where

$$\boldsymbol{A} = \boldsymbol{\sigma_e^2} \cdot \boldsymbol{X} \cdot \boldsymbol{X^T} + \boldsymbol{C_w^{-1}} \tag{2.20}$$

in the form of

$$\boldsymbol{p}(\boldsymbol{w}|\boldsymbol{X}, \boldsymbol{y}) = \boldsymbol{\mathcal{N}}(\overline{\boldsymbol{w}}, \boldsymbol{A^{-1}}). \tag{2.21}$$

The predictive distribution for $\boldsymbol{f}_*$ is given by the average output of all possible linear models of the Gaussian posterior and is Gaussian as well.

$$\boldsymbol{p}(\boldsymbol{f}_*|\boldsymbol{x}_*,\boldsymbol{X},\boldsymbol{y}) = \boldsymbol{\mathcal{N}}\left(\frac{\mathbf{1}}{\boldsymbol{\sigma}_{\boldsymbol{e}}^2}\cdot\boldsymbol{x}_*^{\boldsymbol{T}}\cdot\boldsymbol{A}^{-\mathbf{1}}\cdot\boldsymbol{X}\cdot\boldsymbol{y},\boldsymbol{x}_*^{\boldsymbol{T}}\cdot\boldsymbol{A}^{-\mathbf{1}}\cdot\boldsymbol{x}_*\right). \tag{2.22}$$

It is also possible to express the vector of test outputs $\boldsymbol{f}_*$using training data without noise, correlation and cross-correlation matrices as (considering zero mean of the input)

$$\begin{bmatrix}\boldsymbol{f}\\ \boldsymbol{f}_*\end{bmatrix}\sim\boldsymbol{\mathcal{N}}\left(\mathbf{0},\begin{bmatrix}\boldsymbol{C}(\boldsymbol{X},\boldsymbol{X}) & \boldsymbol{C}(\boldsymbol{X},\boldsymbol{X}_*)\\ \boldsymbol{C}(\boldsymbol{X}_*,\boldsymbol{X}) & \boldsymbol{C}(\boldsymbol{X}_*,\boldsymbol{X}_*)\end{bmatrix}\right). \tag{2.23}$$

and using training data with noise, correlation and cross-correlation matrices as

$$\begin{bmatrix}\boldsymbol{y}\\ \boldsymbol{f}_*\end{bmatrix}\sim\boldsymbol{\mathcal{N}}\left(\mathbf{0},\begin{bmatrix}\boldsymbol{C}(\boldsymbol{X},\boldsymbol{X})+\boldsymbol{\sigma}_{\boldsymbol{e}}^2\cdot\boldsymbol{I} & \boldsymbol{C}(\boldsymbol{X},\boldsymbol{X}_*)\\ \boldsymbol{C}(\boldsymbol{X}_*,\boldsymbol{X}) & \boldsymbol{C}(\boldsymbol{X}_*,\boldsymbol{X}_*)\end{bmatrix}\right). \tag{2.24}$$

There are several ways to implement the Gaussian process regression. The whole process is well explained in the sources [32] [35] [36], including detailed implementations and types.

Gaussian process regression works as a linear smoother of the data, since by finding the maximum likelihood (even in the coordinates with noisy reference data), the process supresses the noise. With this in mind, there are several key parameters of the GPR which define the behaviour of the regression. The following parameters are implemented in Matlab while creating a GPR model, which can be implemented and optimized exactly to match the problem at hand [37]. There are several key parameters and functions influencing the model. Parameter ‘Sigma’ refers to the noise standard deviation of the signal and training symbols, influencing the regression by representing the noise influence on the signal. Its value is optimized by Matlab during the regression.

Kernel function (or covariance function) is utilized as projection of training points into the input space. The parameters of kernel function define the shape of the regression around each training point as well as between them. There is a number of predefined functions as ‘KernelFunction’ including squared exponential kernel (default), exponential kernel, Matern kernel or rational quadratic kernel. Also, the kernel function can be user-defined. It is possible to change the standard deviation and the characteristic length of the kernel function using ’KernelParameters’. Figure 2-9 shows the GPR applied to the three samples (marked by black X in all following graphs) with different kernel functions. The differences between kernels are apparent. Squared exponential kernel strongly holds onto the current trend within the data, whereas exponential kernel requires higher density of samples to significantly react due to the high reliability on Euclidian distance between samples. The remaining kernel functions have slightly different shapes and slopes. The choice of the kernel function influences the regression in the areas with high density of reference points only slightly, yet with increasing distance from reference points the choice of kernel function plays significant role.

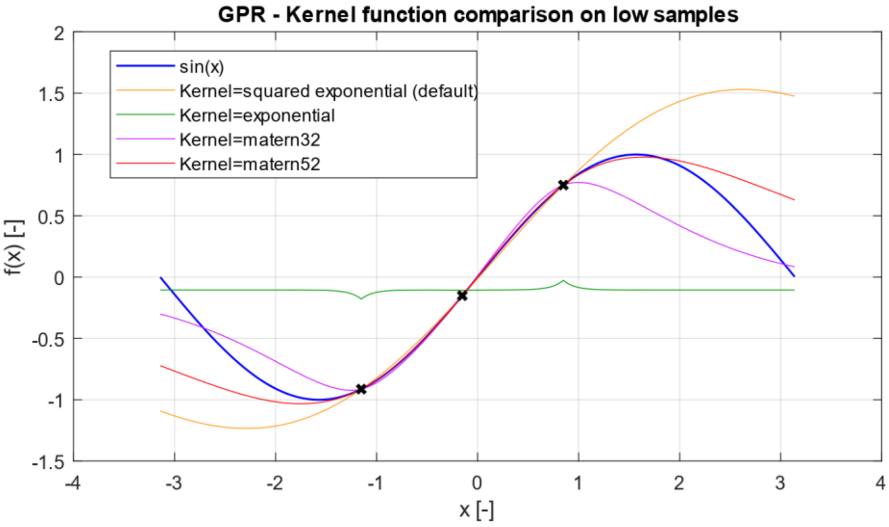


Figure 2-9 Kernel Function Comparison on Low Sample Input (GPR)

Basis function defines the space, into which the problem is projected. It is possible to choose empty basis ‘none’, which does not hold any information for the predictor, which will then always lead to zero or explicitly defined value. Choosing the basis function as ‘constant’ calculates the global trend in data as a constant value derived from the training data. With constant basis the GPR regression returns outputs similar to the IDW. Linear basis ‘linear’ assumes linear trend in data, projecting the results further from the training points onto linear space (inclined line in 1D, inclined surface in 2D). ‘pureQuadratic’ basis function assumes quadratic projection surface. It is possible to define and implement own basis function to the regression. The initial value can be specified using parameter ‘Beta’. See Figure 2-10 for basis function comparison on low input sample example. There, the ‘constant’ basis draws the regression towards the average value of the input samples and stays constant. The ‘linear’ basis finds the linear trend within the data and holds its slope along the whole range. The ‘none’ basis approximates the values between input data points as nicely as ‘pureQuadratic’ basis, but outside of the range of input samples it inclines to the zero value, whereas the ‘pureQuadratic’ basis holds its trend.

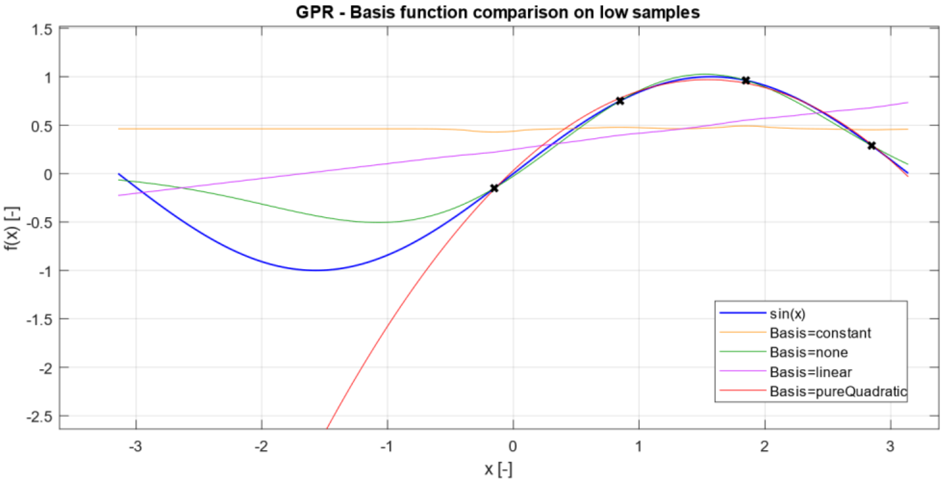


Figure 2-10 Basis Function Comparison on Low Sample Input (GPR)

The method to estimate the parameters of the GPR training symbols ‘FitMethod’ can influence the outcomes. If no estimation is selected (‘none’), the input parameter values will be considered as output. Exact process regression (‘exact’) fully utilizes algorithm on all the training data samples. ‘sd’ option is utilized for regression, where the number of observations is high, therefore the algorithm considers only the subset of the closest observations to estimate the value at each training data point. For a subset of regressors approximation ‘sr’, the kernel function is approximated using a reduced number of training symbols. Fully independent conditional approximation ‘fic’ returns the most accurate results on the cost of highly increased complexity.

The prediction is realized either using all training symbols or by selecting a subset of training symbols in case of a large number of input data (to significantly reduce the computational complexity). The available reduced methods are Block Coordinate descent, Subset of Datapoints or Regressors or Fully Independent Conditional approximation.

## 2.3 MEAN ABSOLUTE ERROR

Mean Absolute Error (MAE) is used to evaluate the precision of utilized regressions within the scope of this thesis. Other considered methods for this thesis were MSE (Mean Square Error) and RMSE (Root Mean Square Error). MAE is utilized, since it preserves the unit, whereas MSE returns the second power of it, and is easier to interpret than RMSE, which is a square root of the sum of squared absolute errors. The formula for calculating MAE is:

$$MAE = \frac{1}{N} \cdot \sum_{\mathrm{i}=1}^{N} |x_i - s_i| \qquad 2.25$$

where $N$ is the total number of samples, $x_i$ is the i$^{\text{th}}$ reference sample and $s_i$ is i$^{\text{th}}$ regression sample.

Based on the definitions of utilized regressions (see Chapter 2.2), it is obvious that some regressions (e.g. LI and IDW) have the value of error = 0 [unit] for the input sample point (see Figure 2-11). This is caused by the fact, that the value of regression at that point is the same as the input data point. Therefore, to evaluate the precision of a regression, it is necessary to distinguish two methods of implementing the MAE calculation. One for those cases, where the regression value is equal to the input sample value and the other for those cases, where these two values are not equal. Both these methods are limited by the number of input data samples.

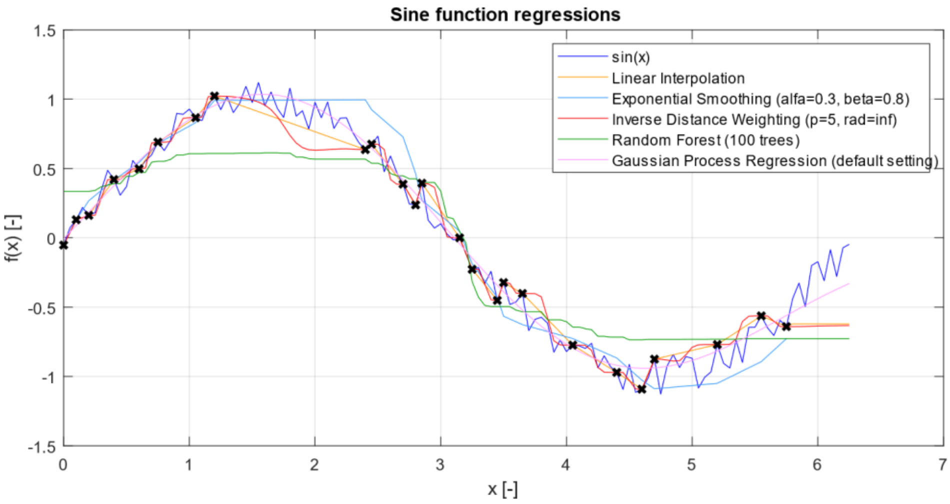


Figure 2-11 Sine Function Regressions

First way how to implement the MAE calculation (MAE1) is to calculate the value of MAE for each point of the regression related to each value of input data points. This method is suitable for those regressions, where the value of input data and regression data is not equal (e.g. GPR and RF). If these values equal (e.g. IDW, LI), the unfair treatment towards such a regression is introduced in relation towards the other regression types. If the set of input data points equals the set of regression-based points, the overall MAE would be 0 (see Figure 2-12, left). If the input data set equals in e.g. 20 % of the data samples, the MAE would have 20 % advantage when compared to the other regression types and therefore their comparison would not be accurate.

The second way of MAE calculation (MAE2) is suitable for methods where the value of input data and regression data is equal (e.g. IDW and LI). It calculates the value of MAE while ignoring the values on reference data coordinates (see Figure 2-12, right). This fact limits the utilization possibilities of this approach based on the amount of data that are considered as reference data. If 100 % of the considered points for the regression are also the reference data, removing all reference data coordinates from the calculation leaves empty set for MAE2 calculation. High (relative) density of reference data limit the accuracy of the method. See Figure 2-13 for the algorithm describing both types of MAE calculation.

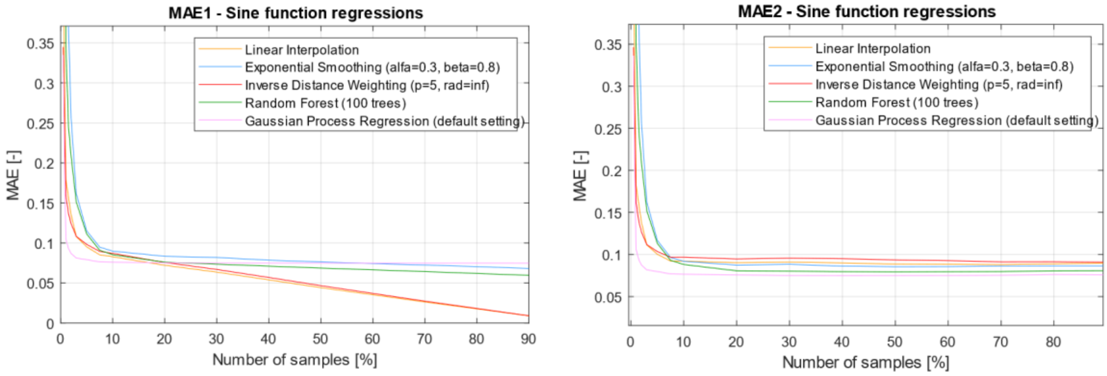


Figure 2-12 Comparison of MAE1 (left) and MAE2 (right)

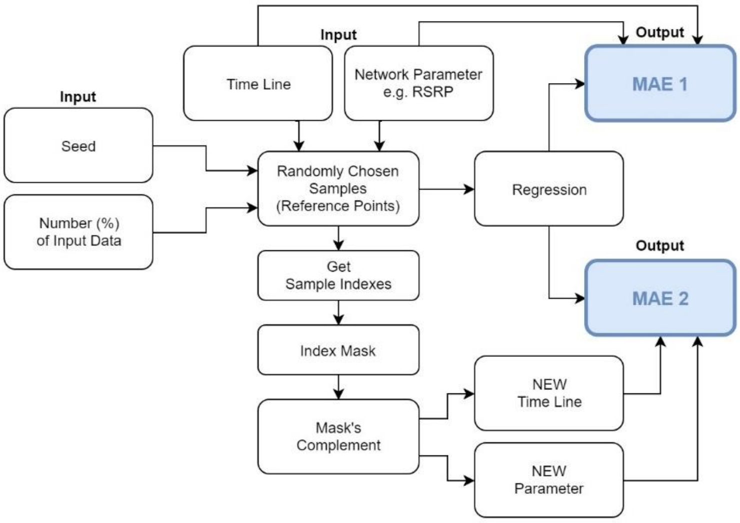


Figure 2-13 MAE Calculation Approaches (MAE1, MAE2)

The algorithm randomly chooses samples from the input data based on the selected random generator seed and selected percentage of data. The chosen samples are considered reference points and the regression is realized with the original time as a grid. MAE1 is calculated across all data points. The mask is derived from the randomly chosen permutation of samples and its complement is used to create new time basis with corresponding parameter. The algorithm applies the mask complement on the previously realized regression. MAE2 is calculated from the reduced (new) parameter and reduced regression result, therefore it does not consider the reference points.

# 3 DATA SOURCES

In this chapter the two primary sources of data are presented and evaluated. First, RTR-Netztest application, testing procedure and test results are presented, followed by an overview of NEMO by Keysight tool. The RTR-Netztest provides a crowdsource-based database while NEMO system is an exact (drive) testing tool.

Drive-testing describes classic testing, when a test attendant operates a testing equipment and measures the network parameters at the defined time and space, either statically, while walking or driving. The results of drive test are always accurate (if done properly) but they offer the information only in time and space when/where the measurement was realized.

Crowdsourcing is a technique, which utilizes the data provided by “the crowd”. Vast databases of data gathered via data-mining application (in our case RTR NetzTest) provides the test results with high diversity (spatial, temporal and qualitative). Using such database to determine the parameters offers savings of time, effort and money for the measurement realization and equipment maintenance in exchange for more demanding data processing.

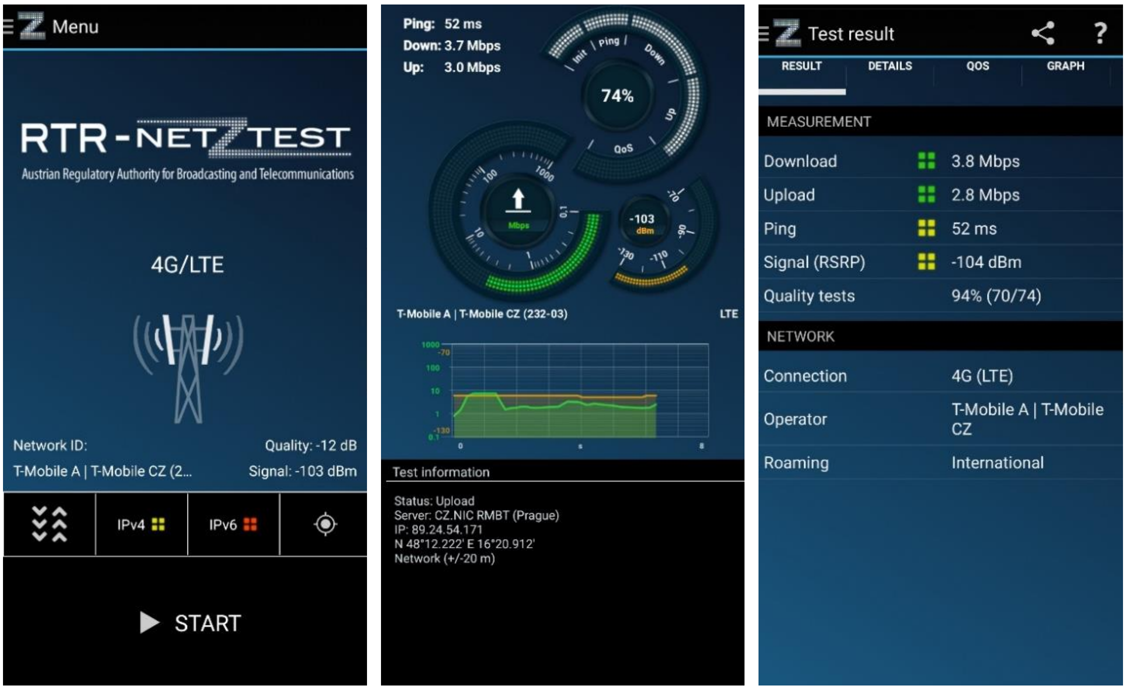
## 3.1 RTR-NETZTEST

In this chapter the RTR-NetTest tool is described. It is the tool used as a main source of crowdsourced data that are later processed. This chapter includes the application overview, its possibilities, settings, viewing option, RMBT test procedure, test outcomes and divides the set of locations based on the test availability and results.

### 3.1.1 APPLICATION PROPERTIES AND INTERFACE

The RTR-NetTest (officially Netztest in German) is an open-data tool provided by RTR (Austrian Regulator Authority for Broadcasting and Telecommunications) to inform a user about service qualities in terms of his internet connection. It gives information including current upload, download, ping, signal strength, background data transmission, IP address, location etc. Also, it enables the user to access the previous measurement statistics, map views and overall summary characterizing the connection characteristics provided by operator. All the measured data are stored within the RTR’s database [18].

The RTR-NetTest may be downloaded for free as an application in Google Play Store as an Android-App in version 2.2, which is available for devices with Android 4.0 ("Ice Cream Sandwich") or above. The iOS-App may be downloaded from Apple iTunes-Store and is available for devices with iOS 7.0 or above. The test is also available in web browser version for devices with other mobile operating systems or while using desktop computers. The example of the Android-App version design is shown in Figure 3-1.



Network ID: T-Mobile A | T-Mobile CZ (2...

Quality: -12 dB

Signal: -103 dBm

Ping: 52 ms
Down: 3.7 Mbps
Up: 3.0 Mbps

T-Mobile A | T-Mobile CZ (232-03) LTE

Test information

Status: Upload
Server: CZ.NIC RMBT (Prague)
IP: 89.24.54.171
N 48°12.222' E 16°20.912'
Network (+/-20 m)

| MEASUREMENT | |
|---|---|
| Download | 3.8 Mbps |
| Upload | 2.8 Mbps |
| Ping | 52 ms |
| Signal (RSRP) | -104 dBm |
| Quality tests | 94% (70/74) |
| **NETWORK** | |
| Connection | 4G (LTE) |
| Operator | T-Mobile A \| T-Mobile CZ |
| Roaming | International |

Figure 3-1 RTR-Netztest Android Design – Home Screen (Left), Ongoing Test (Middle) and Test Result (Right); Screenshot from the RTR-NetTest Android-App

The home screen (see Figure 3-1, left) gives the information about network's ID (internet connection, mobile data/WLAN), current data transmission (upload and download), IP address, location etc. Also, it enables the user to access the menu. In menu, History of user's measurement can be found, followed by Map of all measurements of all users and Statistics. In both Maps and Statistics, the user may filter the displayed data based on his interest. Menu includes Help, Information and Settings buttons as well. The "START" button is located at the bottom of the screen to enable the test procedure to start. When pressed, the application does everything on its own without any interaction from the user needed. First, the application runs speed tests (see Figure 3-1, middle), then it runs a Quality of Service (QoS) tests. Once all tests are completed the results are shown in the detailed test results tables (example of one of the result tables is shown in Figure 3-1, right).

The Android-App requires several permissions from the device including device's location to gain the GPS coordinates of the device and for the devices with Android 8.1 and above to gain the WLAN identifiers SSID (Service Set Identifier), BSSID (Basic Service Set Identifier). Additional permissions include telephone status and identity to be able to recognize the dual SIM devices, Google service configuration to read configuration data for the presentation of Google maps used in the overall result part of the app, battery access to prevent the phone from turning into sleeping mode during the test and to grant full network access and access to Wi-Fi connections in order to set the network's parameters. Disabling any of these permissions could resolve in reduction of the app's functionality.

### 3.1.2 RMBT TEST PROCESS

The following chapter describes the testing process of the RTR NetTest application called RTR Multithreaded Broadband Test (RMBT). The description of the testing process is important to understand the functionality of the algorithm, which is the core aspect of the whole application.

The test itself is supposed to gain a precise measurement of several quantities over the whole available bandwidth of current connection by transferring a multiple parallel data streams over separate Transmission Control Protocol (TCP) connections within a specific time span. The data for each of the multiple parallel data stream are generated with high entropy and are not allowed to be compressed during the stream. To prevent most of the possible conflicts with firewalls and proxy servers the TLS (Transport Layer Security) connection is used. The test procedure may be described in 7 steps. These steps are processed one by one and do not overlap. See Figure 3-2.

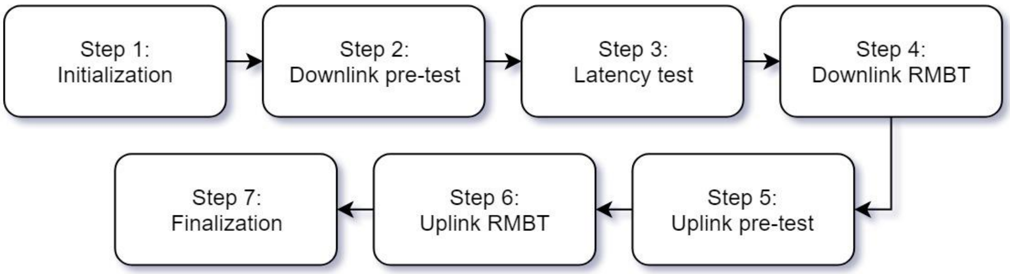


Figure 3-2 RMBT Test Process Diagram

**Step 1:** Initialization begins with the client sending a request, trying to connect to the control server which is acknowledged by the control server and responded with connection establishment information. The client sends then a test request to the control server. The control server selects the RMBT server for testing, generates and sends the token consisting of unique ID, time information of the test and HMAC key identifying the user to the RMBT server, and sends it to the user along with additional test parameters. The client then opens a number of TCP connections to the RMBT server using the token.

**Step 2:** Downlink pre-test is used to ensure proper internet connection to the RMBT server. The pre-test estimates bandwidth, which determines the number of parallel connections. The client requests a data block consisting of random data with high entropy which is then sent by the server. The client sends another request after receiving the whole data block. In case the time of the test did not run out, the server sends another block of double the size of the previous one. The whole data block will be sent even after the time of the test has ran out. In case less than four data blocks were sent during pre-test, all connections except one are terminated and the downlink test will be carried out only using the remaining link.

**Step 3:** Latency test consists of client sending pings to the server, which are responded by the RMBT. Both sides measure the time between sending and receiving the ping message.

During **Step 4:** Downlink RMBT all established connections in Step 1 are being used to receive data blocks of the same size by the client until the time runs out and the last chunk is received. The downlink data rate is then calculated according to time information and the number of chunks received.

**Step 5:** Uplink pre-test proceeds similarly to the Step 2, only with the RMBT server as the receiver and the client as the sender. Either the connections remain opened after Step 4, or the new ones are being established. Each sent block's size is double compared to the previous one. The established connections remain open for the Step 6.

**Step 6:** Uplink RMBT test operates analogous to the downlink RMBT. The client sends packets of data to the server until the test time runs out, after which the termination byte is sent signalling the end of the test. During the test the server additionally sends the time information about the received data and after receiving the termination byte the time information about the whole test is sent to the client.

**Step 7:** Finalization consists of the client sending the measured data to the control server, where all the measurements are being stored.

After the test is finished the measured values are displayed within the application (or the browser depending on the measuring platform). The user may display all his measurements in the app history to compare the progress over time and/or display the history and statistics of all users. This option will be described in the following chapter in detail.

### 3.1.3 MAPS, STATISTICS AND TEST'S EVALUATION

The RTR-NetTest allows users to display measured results in statistics manner as well as in map view. In this chapter the possible filters provided by the app are described and presented by set of maps and app snips, widened by the statistics views. Based on this, further evaluation of the application and extent of the current measurement possibilities will be provided.

The map view consists of four basic display options. These are heat maps (sum of several measurements taken in a small geographical area), points (either one measurement or a combination of measurements taken at the same GPS position), cadastral communities and automatic (points at high zoom level, heatmap at low zoom level). All of them have the same map key (separate for each of measured quantities), which is a colour scale from red through yellow to green (standing for highest value). The map views options are extended by large view/small view to change the map window size, plus/minus buttons to bring points closer/farther from the user and list of base maps, including OpenStreetMap, Basemap.at (standard, high dpi, grey and satellite option) and Bing Maps to allow the user to choose the map style and quality that is desired. For visualisation see Figure 3-3. The accuracy of the viewed result depends on devices participating in the test and the location determining technology which it uses. These may be GPS (the most accurate), determining through the network (WLAN or mobile, only rough results) or via IP address (may not be very useful).

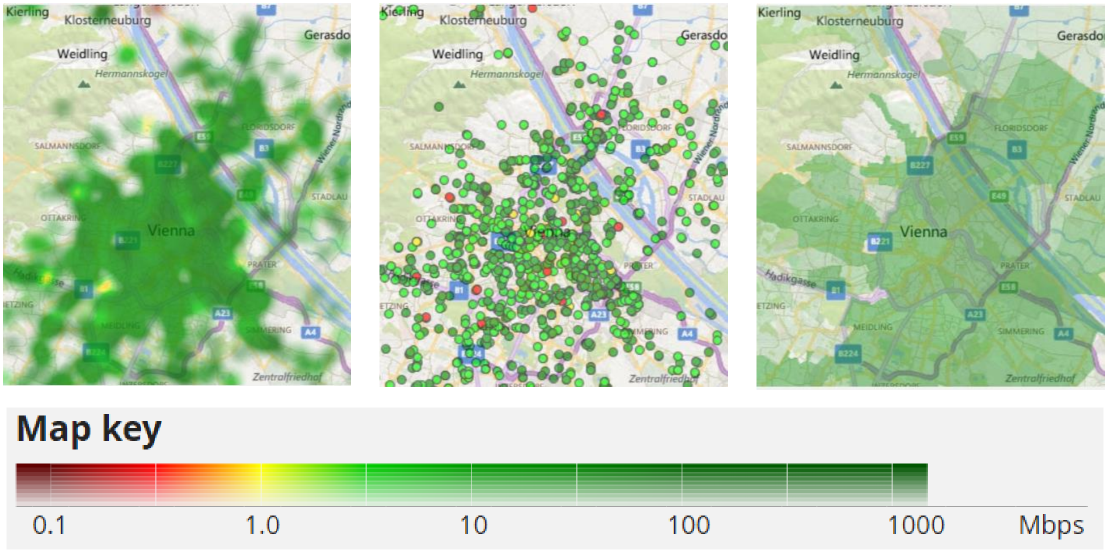
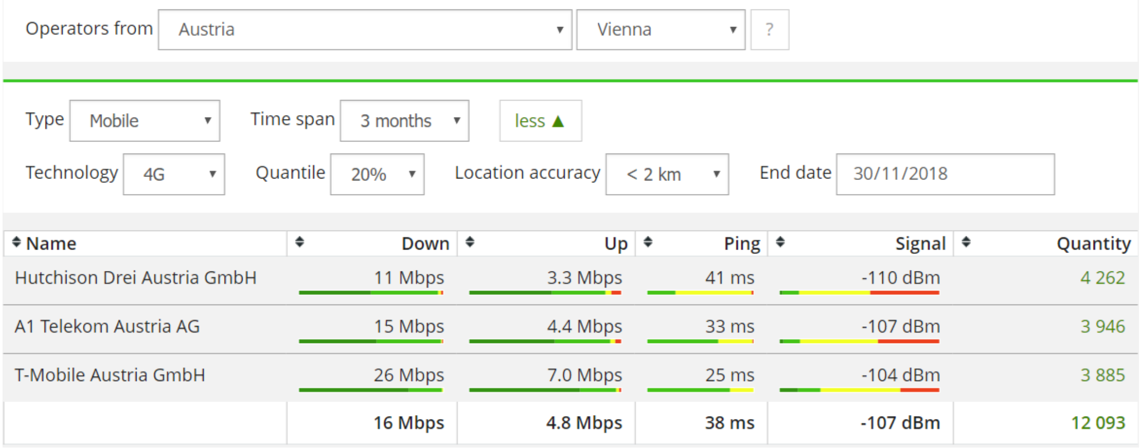


Figure 3-3 Different Map View Options (Heat Map – left, Points – middle, Cadastral Communities – right) [18]

The main filter division of the map view is based on the quantity of interest. These are upload, download, ping and signal. Each of them is then divided based on the measuring platform into mobile, WLAN (app), browser and “all” (standing for the combination of all previously named platforms). The measurements form outside Austria are shown as well. There is an option of selecting the operator of interest between A1 AT, Hutchison Drei, T-Mobile AT and all network operators measured. The choice between 2G, 3G or 4G network generations is present with option of displaying all measurements or their combinations. Further, the filter may be specified by a time range for the last day, week, 1, 3 or 6 months, 1, 2, 4 or 8 years. The last quantity that may be used to specify the viewed search is median and percentile (quantil).

The value of median represents the value of the middle results regardless of the result values [38], in other words the value of median has the same number of results greater than itself and smaller than itself. A percentile indicates a number of percent for which that percentage of results fall below that number. For example, 20 % percentile means the result is higher than 20 % of all results [39]. The application includes 20 and 80 percentile options. This is shown in Figure 3-4, Figure 3-5 and Figure 3-6, where the comparison of measured results with a different percentile is captured in the form of statistic view. With higher amount of “worse” results the average uplink/downlink speed decreases.



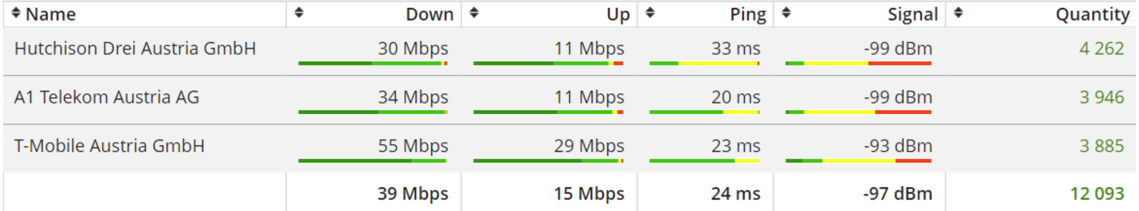
Operators from Austria ▾ Vienna ▾ ?

Type Mobile ▾ Time span 3 months ▾ less ▲

Technology 4G ▾ Quantile 20% ▾ Location accuracy < 2 km ▾ End date 30/11/2018

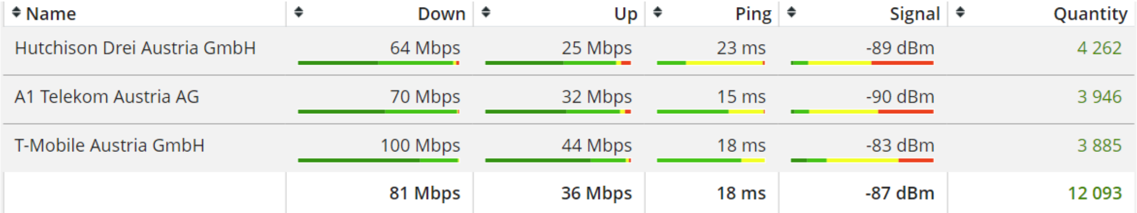
| Name | Down | Up | Ping | Signal | Quantity |
|---|---|---|---|---|---|
| Hutchison Drei Austria GmbH | 11 Mbps | 3.3 Mbps | 41 ms | -110 dBm | 4 262 |
| A1 Telekom Austria AG | 15 Mbps | 4.4 Mbps | 33 ms | -107 dBm | 3 946 |
| T-Mobile Austria GmbH | 26 Mbps | 7.0 Mbps | 25 ms | -104 dBm | 3 885 |
| | **16 Mbps** | **4.8 Mbps** | **38 ms** | **-107 dBm** | **12 093** |

Figure 3-4 Comparison of Statistics - Percentile 20 %



| Name | Down | Up | Ping | Signal | Quantity |
|---|---|---|---|---|---|
| Hutchison Drei Austria GmbH | 30 Mbps | 11 Mbps | 33 ms | -99 dBm | 4 262 |
| A1 Telekom Austria AG | 34 Mbps | 11 Mbps | 20 ms | -99 dBm | 3 946 |
| T-Mobile Austria GmbH | 55 Mbps | 29 Mbps | 23 ms | -93 dBm | 3 885 |
| | **39 Mbps** | **15 Mbps** | **24 ms** | **-97 dBm** | **12 093** |

Figure 3-5 Comparison of Statistics - Percentile 50 %



| Name | Down | Up | Ping | Signal | Quantity |
|---|---|---|---|---|---|
| Hutchison Drei Austria GmbH | 64 Mbps | 25 Mbps | 23 ms | -89 dBm | 4 262 |
| A1 Telekom Austria AG | 70 Mbps | 32 Mbps | 15 ms | -90 dBm | 3 946 |
| T-Mobile Austria GmbH | 100 Mbps | 44 Mbps | 18 ms | -83 dBm | 3 885 |
| | **81 Mbps** | **36 Mbps** | **18 ms** | **-87 dBm** | **12 093** |

Figure 3-6 Comparison of Statistics - Percentile 80 %

It is important to mention, that neither the percentile indicator nor the filter's possibilities capture the quality of coverage by specific operator or the quality of their network capabilities. The required data do not include some crucial information e.g. user's tariff limitation (maximum download/upload speed that is paid for). Also, the connection speed may be affected by other factors than network itself e.g. device capabilities (device with a particularly good/bad throughput), technology of connection (2G, 3G, 4G) or user's location. Any higher amount of measurements in a specific place with particularly good (e.g. roof top) of bad (e.g. basement) reception conditions may influence the final results.

The statistics view allows the user to use the same filters as the maps do – the data may be filtered by the type (mobile, WLAN, browser), time span (including amount of time and the possibility of adding the end date), technology (2G, 3G, 4G, mixed), quantile and location accuracy. In addition, the filters allow to choose the operators based on their country of origin (e.g. Austria) and the region in it (see Figure 3-4, where in the header the filter is set for operators from Austria and region of Vienna). The filter may also display the statistics based on the device, which is used for the test from which the map of measurements done by the same device type may be gained. Additionally, the filters offer the option of advanced search, where each value may be even more thoroughly specified and may be set to a fixed data range that is the most convenient for the user. These settings may also be displayed in the form of histogram with user defined step size and limits from which the set of involved test results may be found (see Figure 3-7, where the filter is set the same as for Figure 3-4, specialized for Samsung Galaxy S8, 3 months span).

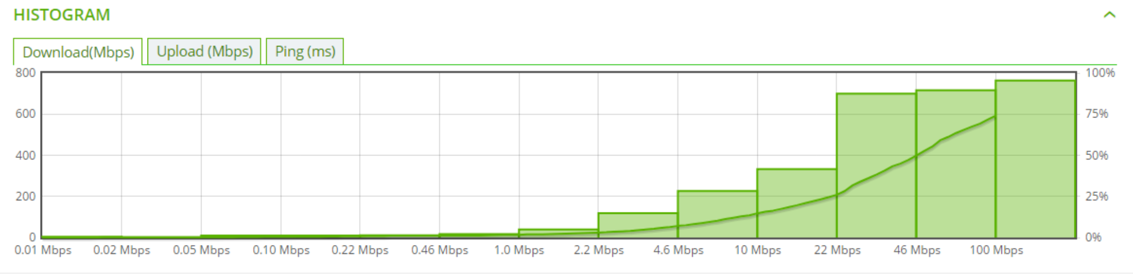
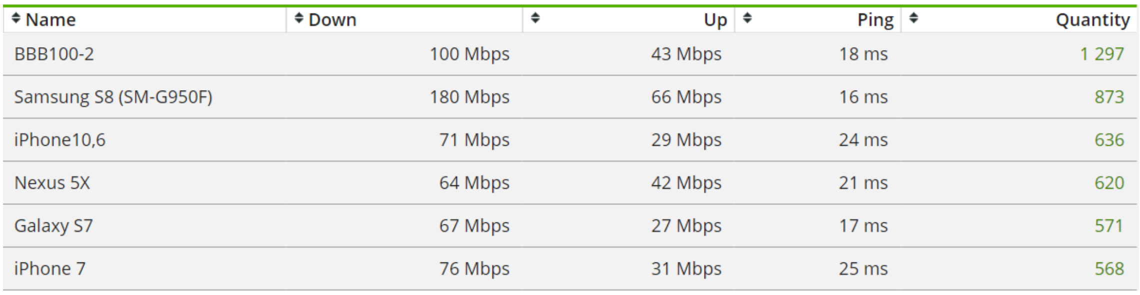


Figure 3-7 Histogram Viewing Mode

In Figure 3-8, the statistics show top 6 devices used for the test in last 3 months (the same filter setting as for Figure 3-4). Based on these statistics it is possible to assume that users with more expensive and newer-model devices (e.g. Samsung Galaxy S8, S7 phones or later iPhones) also pay for more expensive data tariffs, since their download/upload is higher than the one of the users with less expensive or older devices (e.g. Galaxy S5 statistics (with same filter applied) state 35 Mbps for download, 38 Mbps for upload). Also, these devices have a better throughput and therefore are less limiting during the measurement. Additionally, the device settings and software version may have influence on the test result. This may present one of the challenges later in this thesis.



| Name | Down | Up | Ping | Quantity |
|---|---|---|---|---|
| BBB100-2 | 100 Mbps | 43 Mbps | 18 ms | 1 297 |
| Samsung S8 (SM-G950F) | 180 Mbps | 66 Mbps | 16 ms | 873 |
| iPhone10,6 | 71 Mbps | 29 Mbps | 24 ms | 636 |
| Nexus 5X | 64 Mbps | 42 Mbps | 21 ms | 620 |
| Galaxy S7 | 67 Mbps | 27 Mbps | 17 ms | 571 |
| iPhone 7 | 76 Mbps | 31 Mbps | 25 ms | 568 |

Figure 3-8 Top 6 Devices Statistics

To describe the measurement coverage of Austria the Figure 3-9 is used. It captures the map of Austria with results of the mobile download measurements during the period of three months displayed on it. The result is shown for 4G connection and for all local network providers. The areas with high population density have a greater amount of measurements than areas with lower population density. This especially holds for the areas around the big cities (e.g. Vienna and Graz). Also, the areas around the main transport routes (e.g. highways and railroads) are thoroughly measured. This trend may be for example explained by frequent measurements taken by the operators and can be seen in Figure 3-10 (the same filters as for Figure 3-4). A similar trend can be observed in detail of the main Austrian cities. In the detail capturing Vienna (see Figure 3-10) the traffic connections and mainly populated areas have a bigger amount of measurements. On the other hand, the mountain areas (e.g. the Alps) are due to lower population density rarely measured, have lower network coverage quality and are mostly not displayed in the maps/statistics due to the percentile filter, which does not allow to display 20 % of the “worst” measurements. The tests from locations with no service cannot be finished and therefore cannot be stored on the server nor displayed.

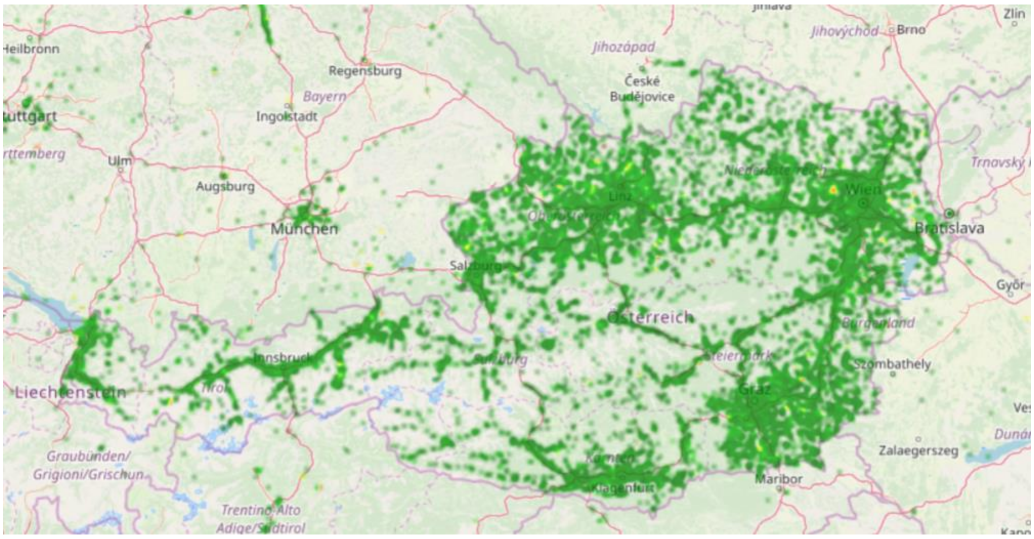


Figure 3-9 Measurement Coverage of Austria

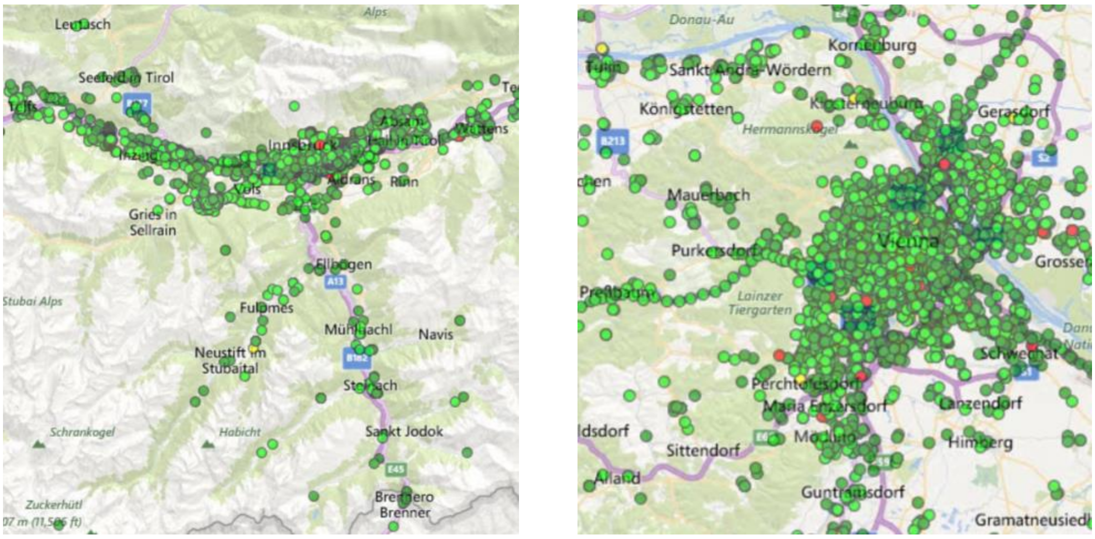


Figure 3-10 Measurement Coverage Demonstration (Left - Innsbruck and Alps, Right - Vienna and Main Rotes)

Based on the analysis above the areas may be divided into 5 categories depending on the amount of measurements per area unit. These categories are characterized below:

**City areas** – areas characterized by high density of measurements, where the vast majority of tests is realized via 4G LTE due to high density of user devices. The user devices are modern, new models which support newest technologies and highest connection speeds. The connection speed in these areas is not limited by general signal quality or user device, but by network load, tariff limitation or measurement location (open-space, underground tunnel or elevator). The network load may vary depending on the time of the day, local events (sport matches, concerts) etc.

**Rural and town areas** – areas with significantly smaller density of measurements than city areas, but still containing a significant number of measurements in populated areas. The network coverage is weaker than in city areas but 4G is still available. There is lower network load. User devices tend to be older by average, therefore number of measurements via older technologies is higher. The connection speed is limited by user device capability and tariff limitation.

**Main communication routes** – areas with locally frequent measurements along the path, either a road or railroad. The network coverage depends strongly on the location, but due to the frequency of the measurements the connection speed is known along the path, probably due to measurements performed by the operator. Using the knowledge of the connection capabilities along the road, it might be possible to calculate the connection speed in the nearby areas.

**Flatland unpopulated areas** – areas with no or almost no measurements with the exception of communication routes, with unknown signal strength or network capabilities. Some technologies (LTE) may not be available in these areas. Due to the lack of signal obstructions, it is possible to calculate the network connection speed based on the network quality on nearby road or in nearby town. The signal behaves similarly in all directions. There is a possibility of ensuring additional measurements.

**Highland, forested areas** – areas with no or almost no measurements with the exception of communication routes, with unknown signal strength or network capabilities. Some technologies (LTE) may not be available in these areas. Due to the presence of various natural obstructions, it may be impossible to accurately predict the network quality in these areas.

This characteristic is supported by the officially published coverage maps of all previously mentioned Austrian providers (see Figure 3-11, Figure 3-12 and Figure 3-13).

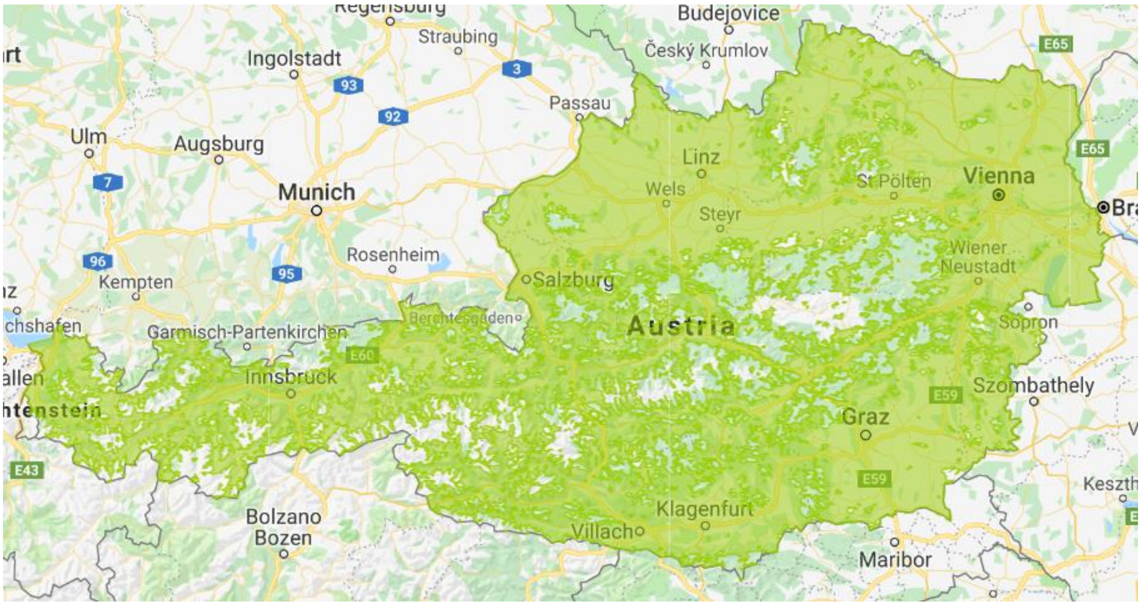


Figure 3-11 Drei Austria - Map of LTE Coverage (LTE - up to 150 Mbps) [40]

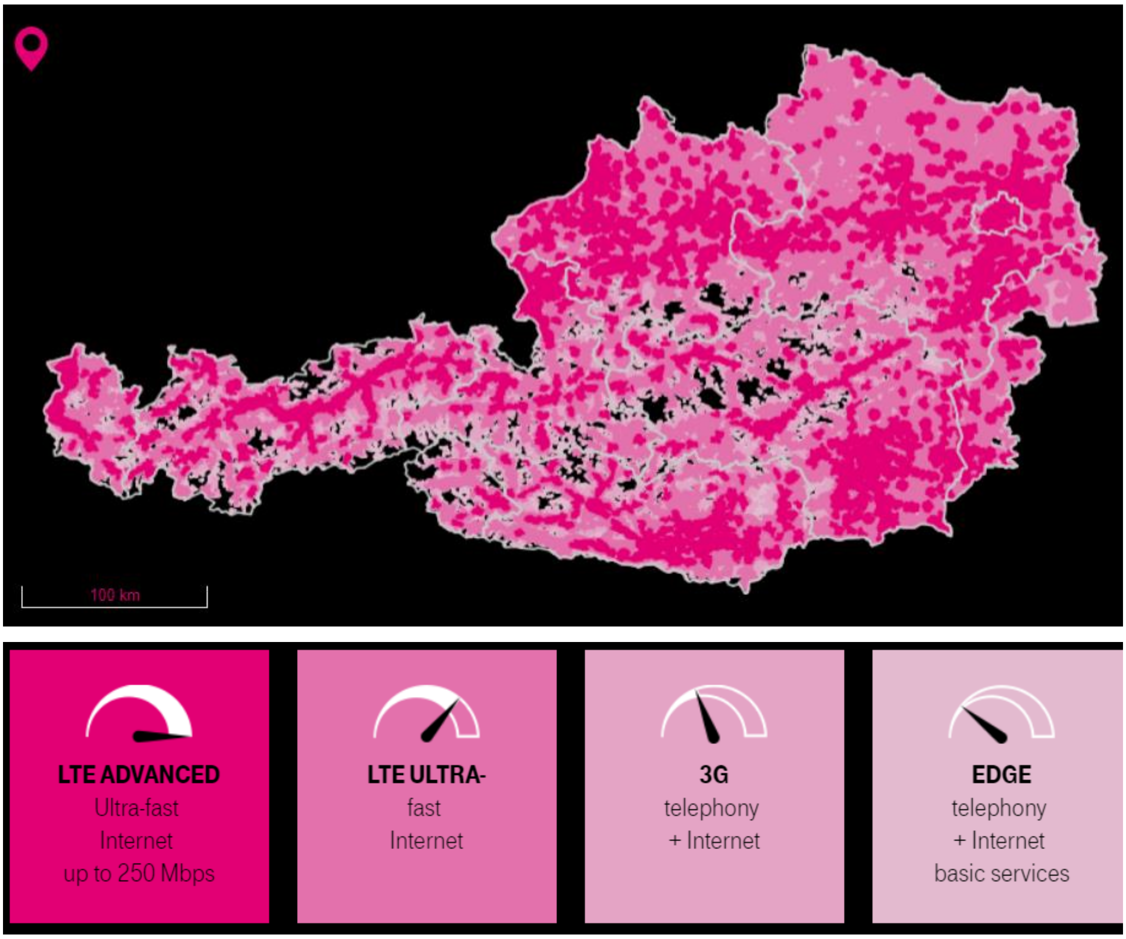


Figure 3-12 T-Mobile AT - Map of Coverage [41]

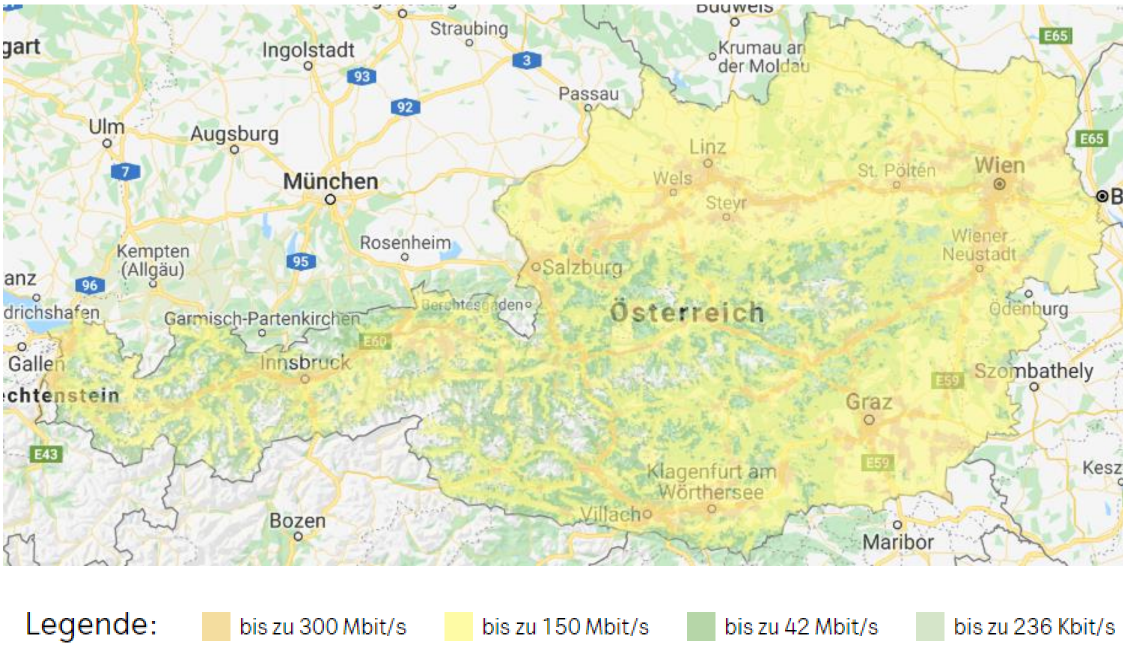


Figure 3-13 A1 Austria - Map of Coverage [42]

### 3.1.4 TEST OUTCOME DATA

As a result of RTR-NetTest measurement the set of data is acquired. In this chapter, the type, definition and form of these data will be presented.

The RTR Multithreaded Broadband Test is capable of measuring a large number of quantities, which are saved and available in the application on the client's device, online on the webpage of RTR and as Open Data available for download. Among the variety of measured parameters, only the most important and impactful data for this thesis will be presented and described. The most important parameters measured by the test are speed of download, which is the speed of transferring data to user's device from another device [43], speed of upload, which is the speed at which the data are being transferred from the user's device to another device, ping or latency, which in our case means the delay between the time of sending a message to the server and the time of receiving a response signal strength [44], which is a power of the signal measured at the user's device, in case of RTR test referring to RSSI or RSRP measurement [45]. The signal strength for mobile phones usually ranges from -30 dBm to -110 dBm [46]. All measured parameters and their descriptions can be found on the NetTest website [18] with full test reports. Also, the website presents the way the results can be requested either in CSV format most commonly used in Microsoft Excel, or as JSON (JavaScript Object Notation) Open Data format. There are two categories of test output, where the first one consists of a single string or a number and the other one as an array of multiple variables (for example data volume over time). The chosen single value results are shown and described in Table 1.

Table 1 Chosen Single Value Results

| Parameter | Description | Type | Example |
|---|---|---|---|
| open_test_uuid | The unique identification of the test | String | "O10b9e95c-d47a-4328-b2ff-82ef24c8e6fe" |
| model | Type of the used device | String | "Sony Xperia Tablet Z LTE" |
| platform | Platform on which the test was run | String | "Android" |
| network_type | Type of the network during test | String | "MOBILE" |
| cat_technology | Category of the technology used during test | String | "3G" |
| time | UTC date and time of the start of the test | String | "2013-07-15 01:56" |
| country_location | Country where the test was run | String | "AT" |
| lat | Latitude of the device's position | Numeric | 48.2029025 |
| long | Longitude of the device's position | Numeric | 16.3967842 |

| | | | |
|---|---|---|---|
| loc_accuracy | Estimation of the positioning accuracy | Numeric | 5356 |
| signal_strength | Estimated signal strength during the measurement in dBm | Numeric | -55 |
| download_kbit | Estimated download speed in kbps | Numeric | 6904 |
| upload_kbit | Estimated upload speed in kbps | Numeric | 712 |
| distance | Distance that the user moved during the course of the measurement in meters | Numeric | 667.16 |

The second category of the measurements presents the information about variables that changed over the duration of the test. The arrays are in JSON Array format to comply with the international standards. Table 2 highlights the array-type results of the measurement important for this thesis. The results are available thread wise (per established TCP connection during testing) and as a total.

Table 2 Array-Type Results of the Measurement

| Name | Explanation | Subarrays | Subarray's meaning |
|---|---|---|---|
| download | Data about downloaded bytes during the test | time_elapsed | Time elapsed since the beginning of the test |
| | | bytes_total | sum of bytes transferred since the start of the test |
| upload | Data about uploaded bytes during the test | time_elapsed | Time elapsed since the beginning of the test |
| | | bytes_total | sum of bytes transferred since the start of the test |
| signal | Information about the measured signal strength during the test | time_elapsed | Time elapsed since the beginning of the test |
| | | cat_technology | Category of the technology used during test ("4G") |
| | | signal_strength | Estimated signal strength during the measurement in dBm |
| | | lte_rsrq | Signal quality in dB in case of LTE network |
| | | lte_rsrp | Signal strength in dBm in case of LTE network |

| | | | |
|---|---|---|---|
| | | cell_info_4G | Additional information in case of LTE network (band, cell identity etc.) |
| location | Information about device's location throughout the test | time_elapsed | Time elapsed since the beginning of the test |
| | | loc_accuracy | Estimation of the positioning accuracy |
| | | long | Longitude |
| | | lat | Latitude |
| | | altitude | Altitude (height above sea level) in meters |
| | | speed | Current speed in meters per second |
| | | bearing | Direction in which the device is moving in degrees, relative to the true north |

As a part of the test, Quality of Service (QoS) measurements are also realized to establish, whether the connection supports various services. "Voice over IP" test checks whether the VoIP connection is possible. It is supported only for Android version of the application. "Unmodified content" test checks, whether the content (picture) downloaded from the server is identical to the content on the server. The test is unsuccessful if the data on the device differs from the original data. "Web page" test tests, whether and how fast the content of the website is downloaded and viewed on the user device. "Transparent connection" test checks, whether the connection between the client and server is direct, or whether the intervening entity (e.g. proxy) changes the request (filters or alters the data). "DNS" test checks whether the website addresses are successfully converted into addresses using DNS system (Domain Name System). "TCP" test checks whether the connection is possible using special TCP ports. "UDP" test checks whether the connection is possible using special UDP ports.

The test offers more detailed version of the test measuring additional quantities. These quantities are not important to this thesis therefore their specification is not discussed here.

Within the scope of the open-data the source code of the test is available online [18].

**Other sources of crowdsource-based data**

Aside from RTR-Netztest, other open-data speed test's results are available for analysis within the university resources but are not focused on Austria. All of them are based on the same software as Netztest, which was formerly developed by Alladin [47], then owned by Specure, now Martes-Specure [48]. These open-data tests are operated by regulatory bodies from across the Europe and include for example AKOS from Slovenia (AKOS Test Net [49]), RATEL from Serbia (RATEL NetTest [50]) or ILR from Luxemburg (checkmynet [51]).

# 3.2 NEMO FROM KEYSIGHT

In this chapter the Keysight NEMO system is presented. The measurement setup, possibilities and available data are described, and the challenges discussed. The algorithm matching results of two separate measurements is shown and discussed.

## 3.2.1 NEMO SYSTEM OVERVIEW

The Nemo from Keysight is a complete setup for measurement and benchmarking various wireless technologies, including the option of real-time, indoor and outdoor measurements. The set-up is capable of testing and recording any desired drive-test measurements and the user experience for various applications and parameters. Besides the usual testing metrices including uplink, downlink, ping etc. the system features include the possibility to measure e.g. voice quality, email processing, YouTube streaming, http data transfer, SMS and MMS messaging, HTML browsing, audio quality tests etc. [52].

The measurement set-up consists of a backpack carrying the whole system, including up to 8 measuring devices and USB batteries (see Figure 3-14). The system may be operated for up to 10 hours. The typically used slave devices within the measurement set-up are e.g. Samsung Galaxy S9 or S8 cell phones, all being operated by a single master device. The master device, e.g. tablet, is used to send commands via Bluetooth connection to the measuring units. The results of the tests are then uploaded to FTP/HTTP server from the master unit or downloaded directly from the slave units [53].

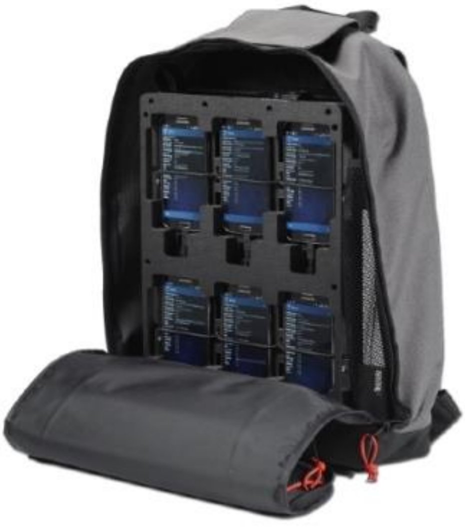


Figure 3-14 NEMO Measuring Kit [53]

The data collected by the system may be later analysed using Keysight's Nemo post-processing tools or their downloaded version may be processed locally, depending on the user's needs and requirements. The primary measurement may be obtained as open ASCII file format and may be then converted into required data format for further processing (e.g. .mat format).

### 3.2.2 AVAILABLE DATA

The available data include the records of nine repetitions of network performance measurement in the centre of Vienna. For these measurements, the identity information of the operator is not available. The data are stored in .mat format and are structured in the structure of structures. The system conducts several types of measurements independently, saving all quantities separately. The resulting measured parameters correspond to immediate values of the measured quantity. The individual measurements consist of the following measurements: Application List (APPLIST), Packet Channel Information (PCHI), Media Access Control (MAC) layer throughput (MACRATE), MAC layer throughput uplink (MACRATEU), Physical Downlink Shared Channel (PDSCH), Physical Uplink Shared Channel (PUSCH), Data Rate (DRATE), Packet Link Adaptation Info for Downlink (PLAID), Packet Link Adaptation Info for Uplink (PLAIU), Channel Information (CHI), Random Access Channel Information (RACHI), Cell Measurement (CELLMEAS), Multiple Input Multiple Output (MIMO) measurement (MIMOMEAS), Transmitter (TX) Power Control (TXPC), Carrier per Interference (CI), Timing Advance (TAD), Data Connection Attempt (DAA), Channel Quality Indicator (CQI), Peer-to-Peer Protocol Layer Throughput (PPPRATE), Radio Link Control Layer Throughput (RLCRATE), External Application Launch (APP), and Global Navigation System information (GPS). Each measurement is available as a structure, containing a number of arrays with measured quantities. Table 3 contains the list of quantities of interest for this thesis.

Table 3 List of selected outputs of NEMO measurement

| **Measurement** | **Quantity name** | **Relevance** | **Description** |
|---|---|---|---|
| GPS | long | Geographic longitude | Accurate to 10-6 longitude degrees |
| GPS | lat | Geographic latitude | Accurate to 10-6 longitude degrees |
| GPS | times | Timestamp of GPS coordinates | Corresponding time information to GPS coordinates, irregular intervals varying from 0.5 to 38 s with mean 1 s, accurate to 10-3 s |
| CELLMEAS | RSRP | RSRP value | Accurate to 0.1 dBm |
| CELLMEAS | RSRQ | RSRQ value | Accurate to 0.1 dBm |
| CELLMEAS | RSSI | RSSI value | Accurate to 0.1 dBm |
| CELLMEAS | times | Timestamp of CELLMEAS values | Corresponding time information to CELLMEAS values, irregular intervals varying from 0.916 to 10.2 s with mean 1.016 s, accurate |

|  |  |  |  |
|---|---|---|---|
|  |  |  | to 10-3 s |
| DRATE | DLrate | Application layer downlink data rate | In bits per second |
| DRATE | times | Timestamp of DRATE values | Corresponding time information to DRATE values, irregular intervals varying from 0.45 to 0.66 s, accurate to 10-3 s |

The table shows, that the individual measurements function independently and to match a value from CELLMEAS measurement to a GPS coordinates, it is necessary to find a way to synchronize the time axes (see Figure 3-15). Assuming that all measurements started at the same moment and since the intervals between samples are short, it is possible to find the closest time values for such matching. Also, the GPS measurements are taken less frequently than CELLMEAS measurements (24337 RSRP samples over all 9 measurements), resulting in reduced number of 2D CELLMEAS samples (11518) compared to 1D. Alternatively, it is possible to interpolate the nearest GPS times to find the corresponding GPS coordinates for every CELLMEAS value, yet since the density of samples is sufficiently high, this approach was not implemented.

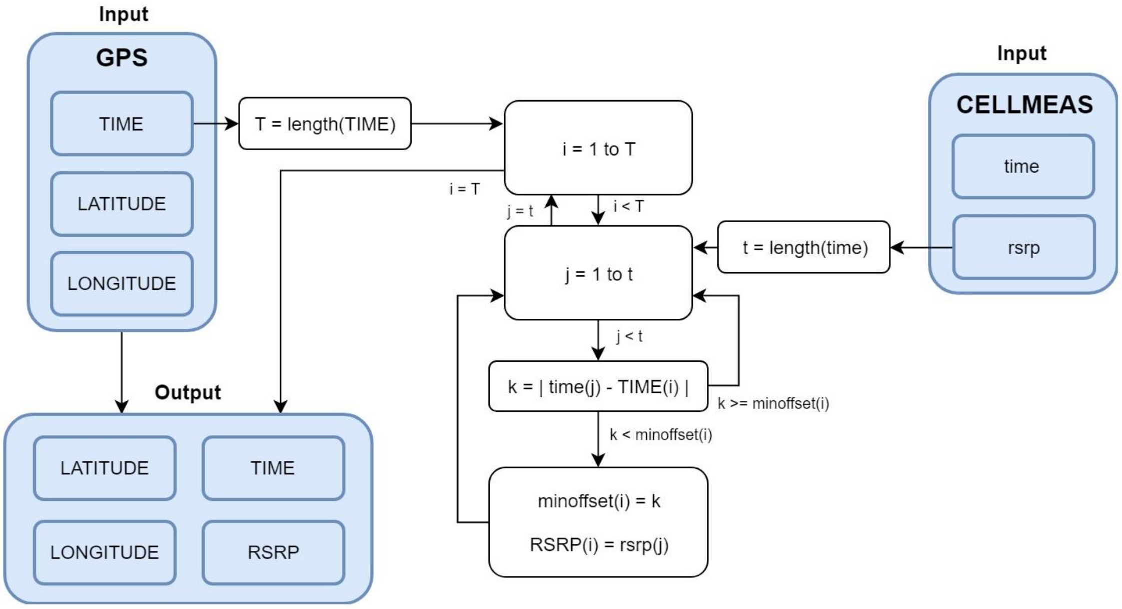


Figure 3-15 Time to GPS Matching Algorithm

The algorithm shown above matches RSRP (or other selected quantity) values from CELLMEAS measurement to GPS measurement by comparing times of each measurement's sample (function “getRSRPxy”). For every time sample from GPS measurement (first for loop) the algorithm checks all time samples from CELLMEAS measurement (second for loop) and iteratively finds the closest samples from each measurement. With each iteration of successfully finding new closest value, RSRP value corresponding to the new closest sample from CELLMEAS measurement is assigned to the current GPS measurement. Additionally, the algorithm checks for duplicate LATTITUDE-LONGITUDE pairs and in case of the repeated measurement at the same coordinates, leaves only the first entry. In practice, this step removes a few samples at the beginning of the measurement, when the measurement setup is being finalized with already running NEMO system while staying at the same GPS position. As the result, to every unique GPS (latitude, longitude) sample is assigned a single RSRP value with time information, resulting into 11518 samples over all 9 measurements.

# 4 1D ANALYSIS

In this chapter the utilization of several regression methods onto a presented input data is described. Each of the regression methods was implemented in Matlab and later compared to the original dataset. At the end of the chapter the mutual comparison is discussed and evaluated for purposes of 2D regression utilization.

## 4.1 INPUT DATA

For the purpose of comparing the individual methods, each of the regressions was first based on the time domain measurements rather than location-based measurements to reduce the problem to a single dimension (1D). The evaluation of the methods in time dimension provides the information in the whole range of the axis. The data were measured using Keysight NEMO [52] devices on the route through the centre of Vienna and were provided by the TU Vienna. In order to get enough data points, the same test was taken repeatedly while measuring the required parameters along the same path through the centre of Vienna, resulting in 9 repetitions of the test. For example, all measurements of RSRP in a single graph are shown in Figure 4-1, one measurement of RSRP is shown in Figure 4-2.

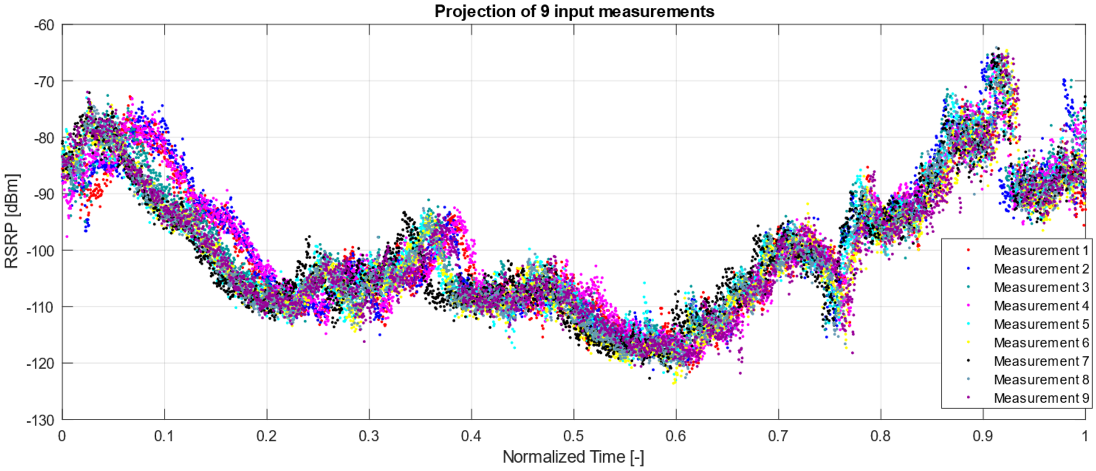


Figure 4-1 Projection of 9 Input Measurements

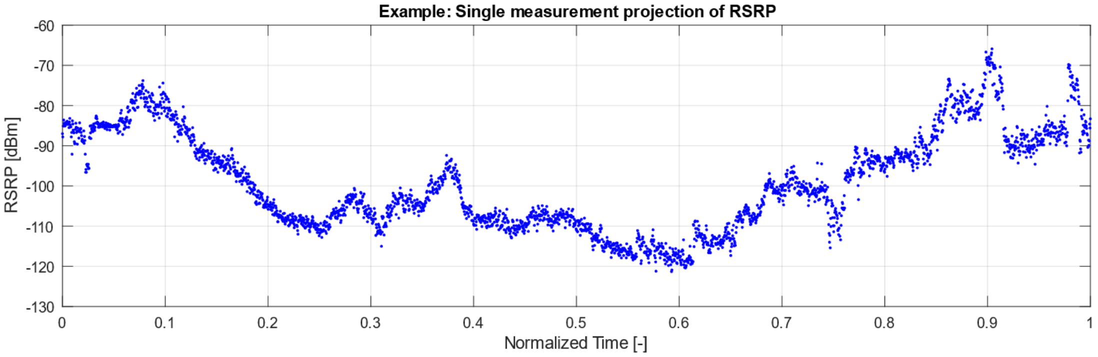


Figure 4-2 Projection of Single Input Measurement

The normalization from 0 to 1 for the time axis was made for each measurement to synchronize the start and the end of the timeline due to variations in the duration of the measurements while the path remained the same. Although this procedure resolves the varying length of the test duration problem, it does not suppress all the distortions in the data. For example, the data were acquired while walking along the approximately the same route nine times. However, slight differences in the path, together with the changing speed of the walker during the test caused some shifts in between the individual tests. See Figure 4-3, capturing the shift of RSRP measurement between measurements number 2 and 9. The maximum RSRP is shifted by approximately 0.05 of normalized time from each other. The difference between RSRP level at t = 0.1 is 11.6 dBm (Measurement 2, RSRP = -80.7 dBm and measurement 9 RSRP = -92.3 dBm).

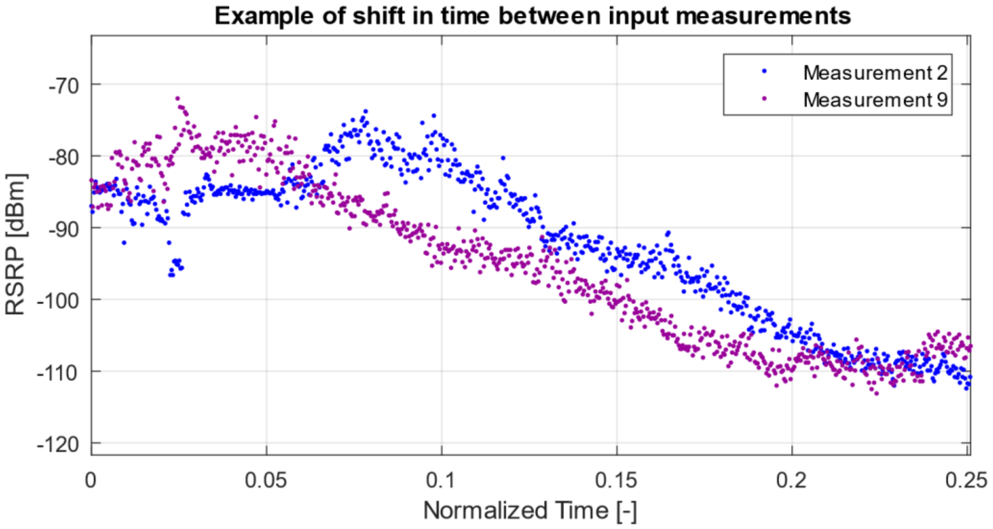


Figure 4-3 Example of the Shift between Measurements

These distortions could not be removed without further processing and adjusting the data, which could not be done without compromising the data information. Therefore, the measurement data fusion does not give any other information than the average information in a specific part of the test (e.g. in the middle of the route/time which the test took). In this case this fact is not a concern, since the fusion of the measurements in 1D is only done to gain more data samples so that more regression realizations can be created. Applying the regression to each measurement realization results in gaining more reliable error matric by averaging the resulting calculated errors. These distortion matters do not apply to the 2D regression solutions, where neither the speed of the walker nor the changes in the route are relevant due to its GPS-only dependency and time variance independency. Each measurement consists of over 2500 input samples (see Table 4 for the number of samples in CELLMEAS measurement for RSRP).

Table 4 Number of RSRP Samples per Measurement

| Meas. Number | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| Number of Samples | 2532 | 2724 | 2739 | 2757 | 2620 | 2837 | 2543 | 2816 | 2769 |

## 4.2 1D REGRESSION BASIS

There are two available bases for 1D regression. The first option is to analyse the data in time, the other is by considering the distance between the measured samples. The comparison of the two methods, both in basic and normalized form, are evaluated within this chapter. As the best-fitting result, the normalized time option is used for 1D analysis. It is easily implemented, since each measurement (and sample) already contains a time stamp, from which the basis is easily derived. To find the distance basis, recalculation from latitude and longitude data to distance in meters is required.

The recalculation from lat/long information to distance has several widely known solutions. The first and the easiest to implement is calculating Pythagorian or Euclidian distance between the samples. This solution offers fast and easy recalculation but introduces an error since it does not account for the spherical surface of the Earth. Haversine [54] formula is a method used to calculate the distance on a sphere. Vincenty's formula [55] is an advanced, iterative method used in geodesy to measure the distance between two points on a spheroid.

The chosen solution for the recalculation is the Haversine formula with spherical parameters adjusted to Austrian (Viennese) geographical position. The latitude of Vienna is 48.2 degrees north with approximately 200 meters above sea level, resulting in local Earth radius of $r = 6366.5$ km. These parameters overcome the fact, that Earth is an ellipsoid by considering the parameters above and limiting the method's accuracy to latitude-wise similar locations. The Haversine formula for distance between point 1 and point 2 is shown in Equation 4.1:

$$d = 2 \cdot r \cdot \arcsin\left(\sqrt{sin^2\left(\frac{\varphi_1 - \varphi_2}{2}\right) + cos(\varphi_1) \cdot cos(\varphi_2) \cdot sin^2\left(\frac{\lambda_1 - \lambda_2}{2}\right)}\right) \qquad 4.1$$

Where $\varphi_1$ and $\varphi_2$ refer to the latitude point 1 and point 2 in degrees and $\lambda_1$ and $\lambda_2$ refer to the longitude of point 1 and point 2 in degrees. The implemented algorithm for distance basis calculation takes latitude and longitude arrays as input and returns corresponding arrays containing distances between the neighbouring samples, as well as cumulative distance between each sample and the current one. First value in both distance and cumulative distance arrays are 0. The algorithm then iteratively calculates the Haversine distance from latitude and longitude coordinates of the current and previous samples until all distances are calculated. The cumulative distance $cd$ is a sum of all previous Haversine distances, as shown in Equation 4.2:

$$cd_i = \sum_{k=1}^{i} d_k \qquad 4.2$$

Figure 4-4 (left) shows the comparison of measurement 1 and measurement 9 with distance basis. Due to the varying noise and path of the two measurements, the length of the measurements significantly varies (3.87 km in measurement 1 and 3.10 km in measurement 9), resulting in over 20 % distance difference between the two measurements. Normalizing the distance using rescaling function, which is implemented in Matlab, is shown in Figure 4-4 (right). This method compensates the length variation, but introduces the bias within the axis, as clearly seen between 0.3 and 0.4 x-axis values.

This bias is randomly increased and decreased based on the varying noise strength and differences in paths between individual measurements, yielding uncertain results.

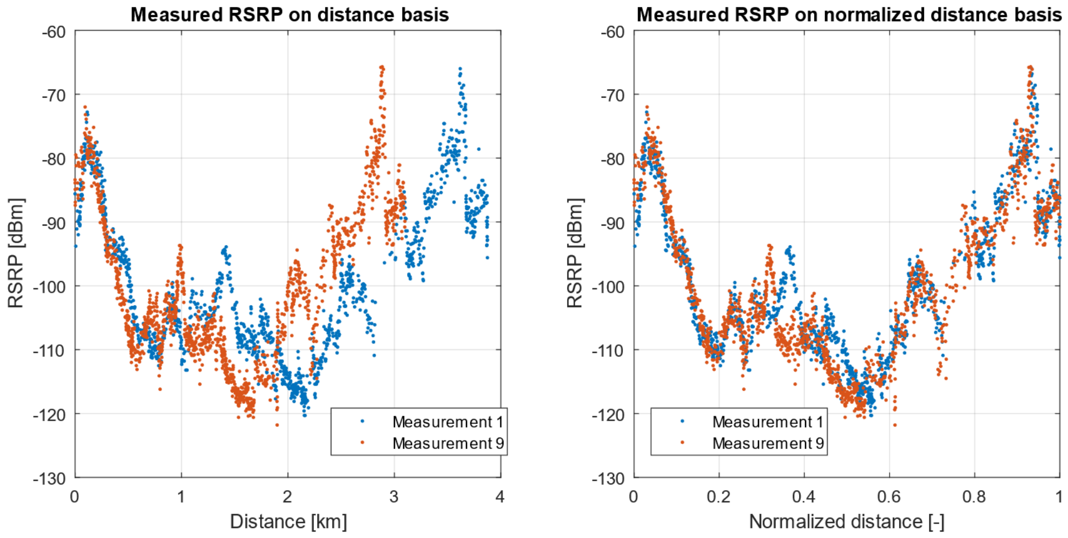


Figure 4-4 Example of Distance Basis

The second solution for 1D basis is plotting the data in time, while setting the first time sample to zero. This solution is shown in Figure 4-5 (left) and clearly introduces the same error as the distance base, although on the smaller scale (the difference in duration is 119 seconds between the two measurements, resulting in 8 % of time difference). Figure 4-5 (right) shows the measurements on the normalized time axis. The figure shows the smallest distortions between the measurements, as the time variations within the measurements were almost uniformly spread throughout the duration of each measurement.

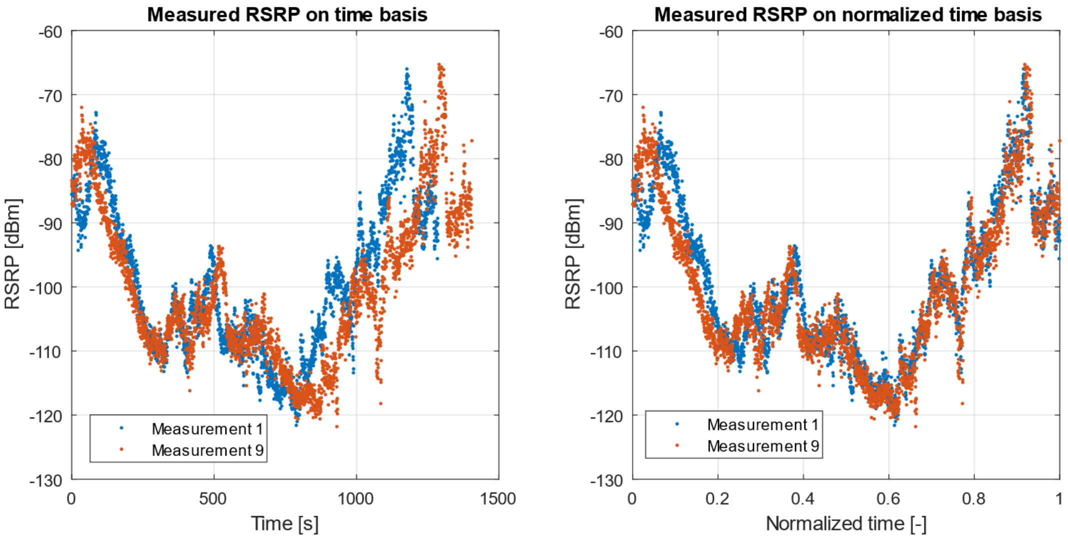


Figure 4-5 Example of Time Basis

After comparing the four possible solutions for x-axis in 1D comparison, normalized time is considered for the rest of this chapter as the method with the highest concurrency of the measured quantity across the whole range of the axis.

## 4.3 1D REGRESSIONS

In this chapter a set of regression methods used to predict the data values in the time intervals for which there are no measurements available is presented. This process is realized as a 1D utilization of regression methods, that are to be evaluated and the results of their performance serves as a base for 2D utilization, leading to the creation of 2D coverage maps of network parameters.

The introduction to each method and its basic theory may be found in Chapter 2.2, followed by Chapter 2.3, where the MAE definition and calculation is introduced. MAE serves as a basic metric of evaluation of the performance and comparison of all the methods. This evaluation is to be found at the end of this Chapter.

### 4.3.1 LINEAR INTERPOLATION

As stated in Chapter 2.2, the linear interpolation is based on establishing the unknown value points from the set of known points geometrically by connecting two adjacent points in plane by the straight line, as “connect the dots” algorithm.

This technique itself does not predict any additional points outside of the input data range. There is a built-in function for linear interpolation implemented in Matlab called interp1(). This function interpolates all original points and then stops. The prediction was then implemented as an algorithm considering the last known point as the constant prediction value. The practical utilization of this extension is to ensure the same vector length at the output as the input grid. See Figure 4-6 for an example of linear interpolation applied to a training data while the amount of input data was changing – input data thinning (100 %, 10 % and 1 %).

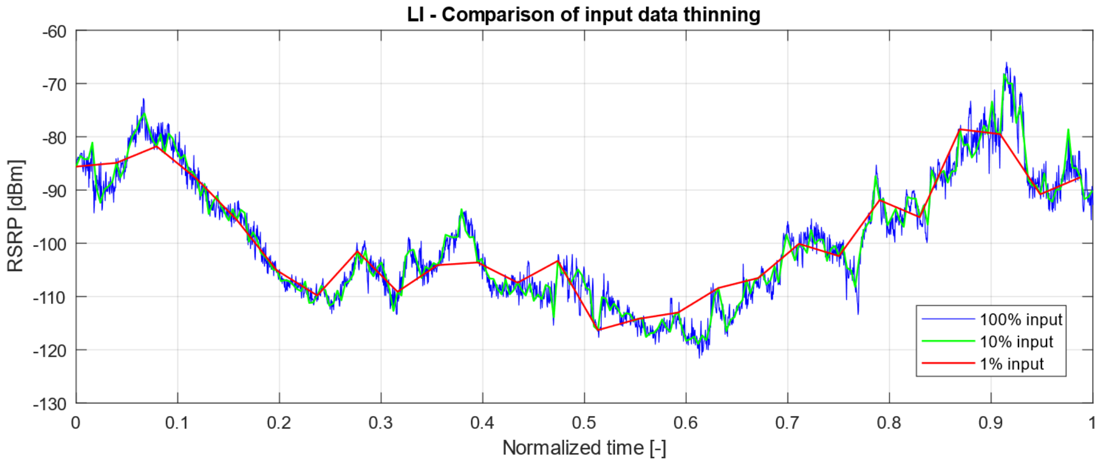


Figure 4-6 LI - Input Data Thinning

To evaluate the method, MAE calculation is utilized in the same manner as described in Chapter 2.3. Linear interpolation has the same value of regression points as are the values of training samples, therefore the MAE2 approach of calculation is used to evaluate this method. In Figure 4-7, the comparison of MAE1 and MAE2 may be found. With the increasing number of input samples, MAE1 approach tends to decrease the MAE value towards zero. While the set of regression points is equal to the set of input points, the value of MAE is zero, which may cause a misinterpretation of results. The MAE2 neglects this problem and therefore it is the preferred method for the comparison of this method to another methods. The same approach is used to evaluate the IDW regression.

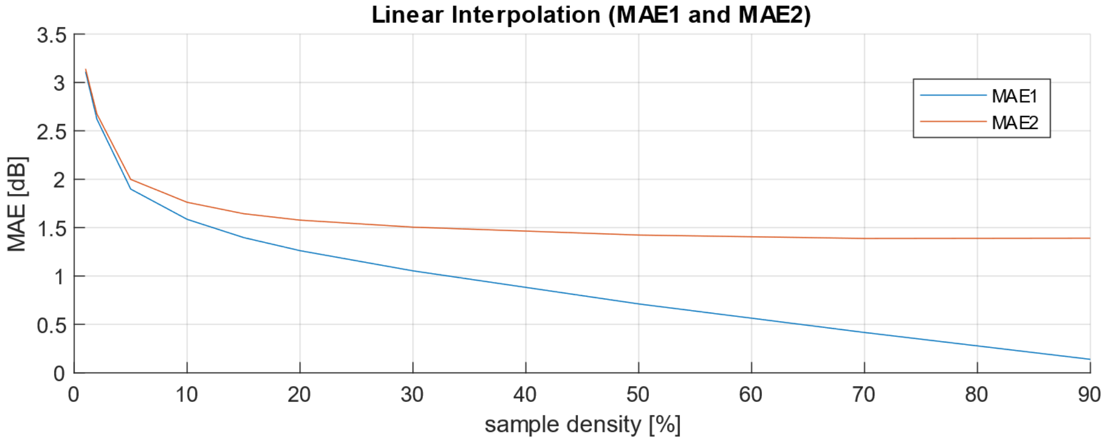


Figure 4-7 LI - MAE Comparison

To ensure the precision and validity of linear interpolation MAE calculation, the regression algorithm was run to evaluate MAE for each sample density across all 9 measurements 5 times while using a different random generator seed to pick different samples and ensure reproducibility. Each 5 different, randomly picked sets of input samples for each measurement were then averaged to acquire statistically more reliable MAE calculation result. All nine measurements were then averaged to gain the final MAE dependency on the number of input data, as shown in previous figure. The MAE of LI applied to used data is 1.5 dB at 30 % of input data and less while using more than 30 % of input data.

## 4.3.2 SMOOTHING TECHNIQUES

The techniques not generating any new samples will be discussed in the following text. First, the exponential smoothing's utilization as a regression method is tested, then exponential smoothing and other smoothing techniques are evaluated as noise-reduction techniques on noisy sine simulation. Reducing the noise form the measurement is utilized in later chapters of this thesis combined with regression techniques such as IDW, which return the exact values of the reference points at their coordinates as the output. These techniques will be evaluated by applying the smoothening to the noisy sine function with MAE calculation with sine function (without noise) as reference. Such approach is impossible to apply directly on NEMO data, since there is no noiseless reference available.

### EXPONENTIAL SMOOTHING

Exponential smoothing (ES) is a technique for smoothing time series sample-by-sample. Compared to the other regressions, ES does not create new data points (see Chapter 2.2). See Figure 4-8 for an example of ES, while changing values of alpha parameter (for fixed value of parameter beta), applied to the full set of input data gained from NEMO measurements.

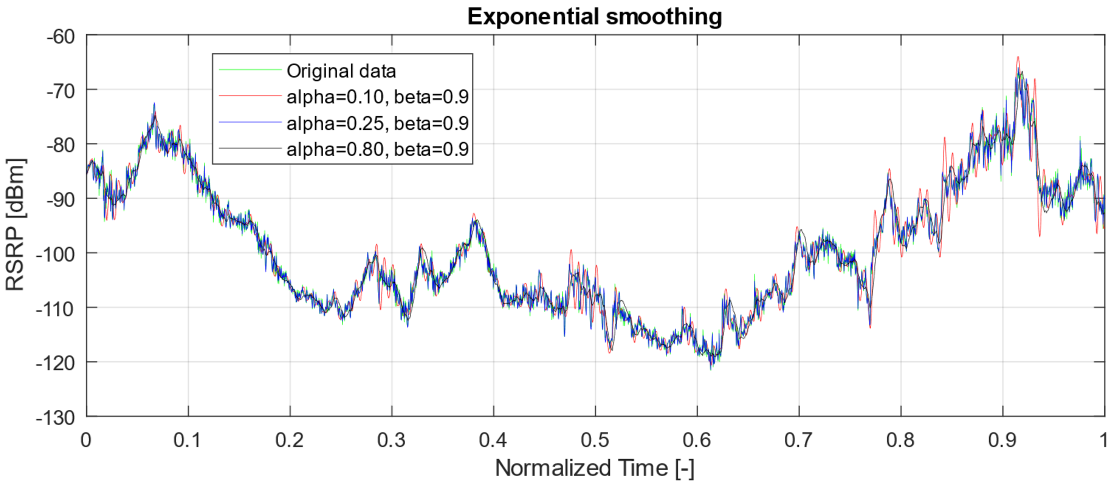


Figure 4-8 Exponential Smoothing Example

The algorithm weights the past observations with the decreasing weight over time and based on the chosen parameters, assigns the new point as a combination of the new value and the value from the past. The name exponential refers to the exponentially decreasing weight of the sample points from the past on the current sample.

The mean average error was calculated based on the number of used samples across all 9 measurements for 5 different seeds the same way as it was for linear interpolation. Since the exponential smoothing is not able to generate new samples, linear interpolation was used to connect the estimated points. The sweep was realized across both alpha and beta values from 0 to 1 with 0.1 step and the results were compared. With the MAE2 comparison, the smallest error was measured for each entry with alpha equal to 1 (see Chapter 2), disregarding beta parameter completely the code as well as the results can be found in the attachment of the thesis. The comparison shows, that minimizing MAE2 (or MAE1) cannot be considered as the only metric for evaluating the estimations, because the results inaccurately suggest, that exponential smoothing is not usable as a regression method since linear interpolation (alpha equal to 1 in ES) gives the smallest average error.

Exponential smoothing, as the noise-filtering technique, was also applied to the noisy sine signal. The evaluation of the method was repeated for 20 different seeds for noise. The signal amplitude 1 [-] and noise amplitude 1 [-] was generated for one period of sine signal (0 to 2 pi) with 0.01 step (629 samples). The MAE1 (since all samples were used to evaluate the method) was calculated for signal - noise and for signal – smoothing across all parameters with maximum resulting MAE reduction by 56.34 % for alpha equal to 0.2 and beta equal to 0.1. After changing the step to 0.05 resulting in 126 data points, the maximum resulting MAE reduction by 45.5 % was found for alpha equal to 0.3 and beta equal to 0.2 (see Figure 4-9, where three different settings were used for comparison). After lowering the noise amplitude to 0.1 [-] and setting the sampling to 0.01 the resulting in 54.84 % MAE improvement with alpha = 0.2 and beta = 0.1. As a conclusion exponential smoothing can be utilized as a noise-reduction technique with approximately 50 % efficiency.

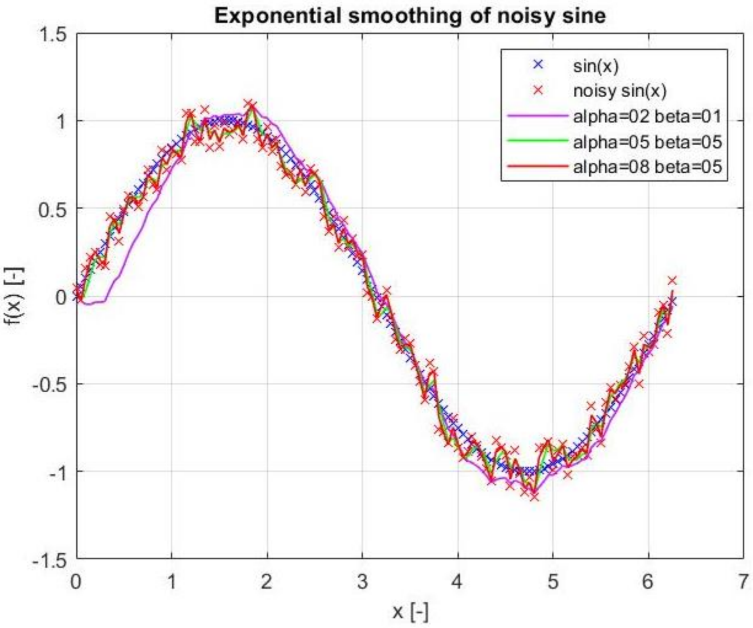


Figure 4-9 Exponential Smoothing of Noisy Sine Function

## MOVING AVERAGE

Utilizing moving average as noise-cancelling technique is commonly utilized throughout the community [56] as an easily implemented and efficient technique. Similar to the exponential smoothing comparison, the moving average MAE calculation was utilized on the training noisy sine with amplitude 1 and noise amplitude 1. The sweep was realized on window sizes from 1 to 99 with step size 2 (to exclude even-sized windows) for 0.05 signal step size. The smallest MAE between signal (without noise) and moving average estimation was for window of 17 samples with MAE reduction by 71.2 % in comparison to signal – noisy signal MAE. For step size 0.01, the MAE reduction was 86.6 % for window size of 61 samples. Figure 4-10 (left) shows the dependency of MAE on the size of the window for step 0.01 samples, where the values of the window size minimizing MAE are in the lower parts of the graph. There, the blue crosses symbolize the MAE of sine signal to noisy signal, yellow ones symbolize the MAE of moving average estimation to noisy signal and red ones symbolize the MAE of moving average estimation to sine signal. The graph clearly shows the significant reduction of error caused by noise by applying moving average technique.

Figure 4-10 (right) shows the moving average with window sizes of 15, 31, 61 and 121 samples applied to the NEMO measurement results. Considering that sampling interval in NEMO CELLMEAS measurement was 0.5 seconds, the scattering of the values of RSRP is caused by static noise. Utilizing moving average algorithm with 15 to 31 samples corresponds to averaging the measured value with ± 3.5 seconds to ± 7.5 seconds of measured values. An average person walks with approximate speed of 1.5 meters per second, meaning the ± 3.5 second to ± 7.5 second time window corresponds to 10.5-meter to 22.5-meter spatial bin size, which is more than sufficient for performance mapping.

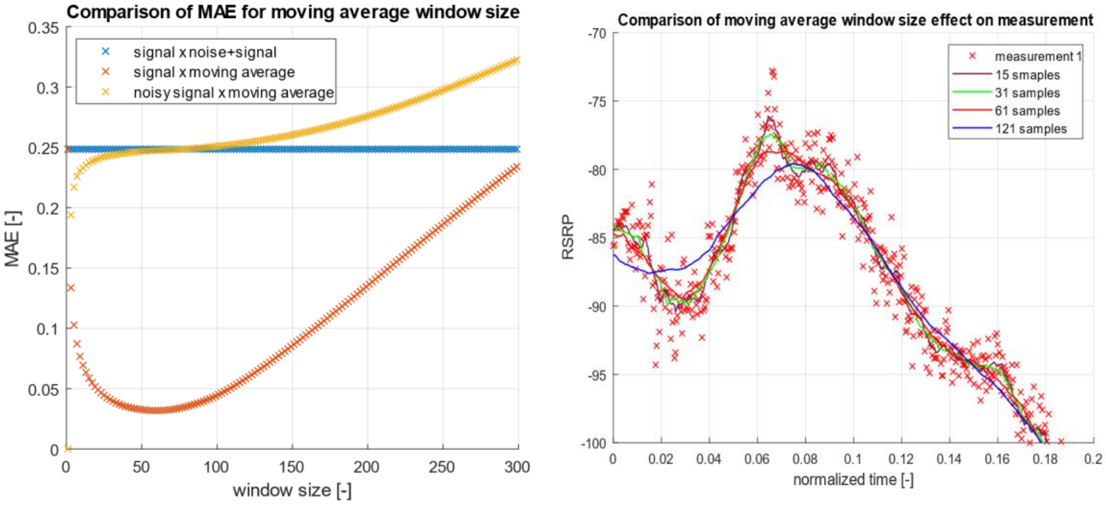


Figure 4-10 Moving Average Comparison (MAE - left, Window Size - right)

## SMOOTHING FUNCTIONS

Techniques similar to moving average were created using different window weighting shapes than constant value. Such techniques include Gaussian-shaped window, triangular window, moving median or Savitzky–Golay window etc. The comparison of MAE of Gaussian smoothing (GS), Savitzky–Golay smoothing, moving median and moving average with window size varying from 1 to 501 samples with step 2 on noisy sine (sampling 0.01 [-], 629 samples per period) was realized and shown in Figure 4-11 (left). The figure shows, that the smallest value of MAE was measured using Savitzky-Golay filter smoothing with window size of 327 samples (larger than half of the sine period) with MAE reduction 91 %, followed by Gaussian kernel smoothing with window size of 93 samples with noise reduction of 83.6 %.

Figure 4-11 (right) depicts the detail of sine function and the smoothing methods. Since the changes in NEMO data are inconsistent in frequency, Savitzky-Golay filter smoothing with window size of 201 samples does not track the trend of data with sufficient dynamics (see Figure 4-12). The lower-sample Savitzky-Golay filter and 31 and 61-sample Gaussian smoothing give comparable results with consistent dynamics and noise suppression throughout the whole measurement.

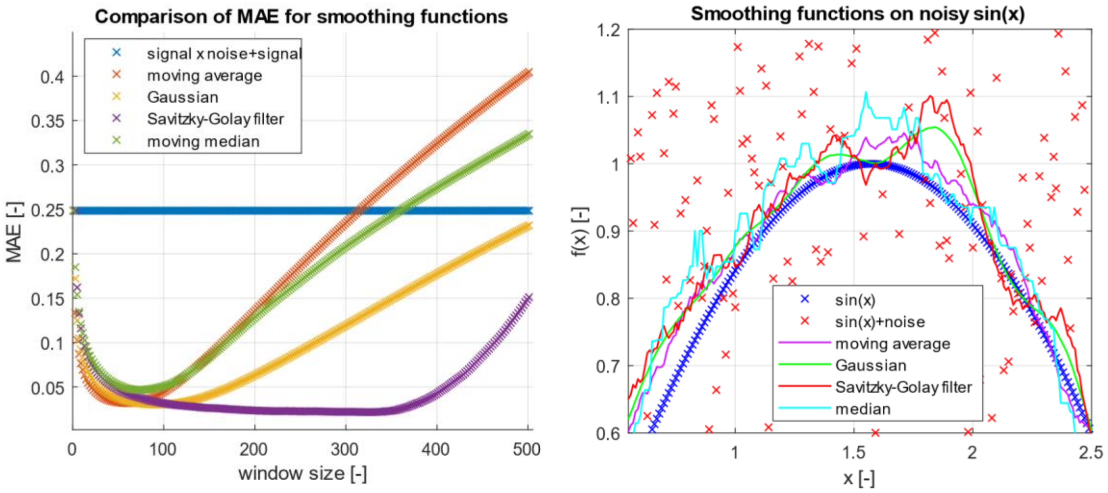


Figure 4-11 Smoothing Function Comparison (MAE - left, Noisy Sine - right)

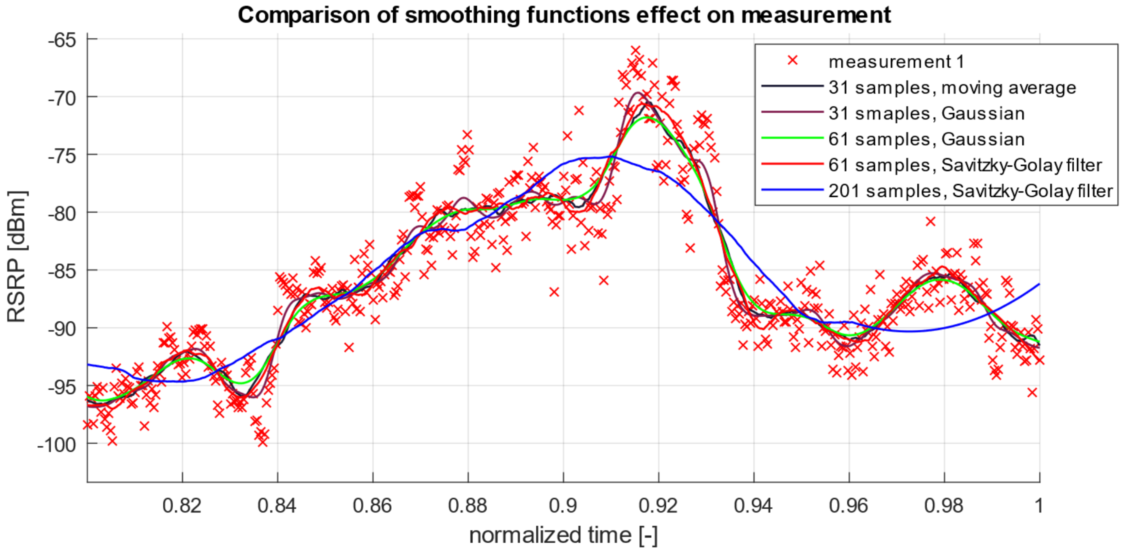


Figure 4-12 Smoothing Function Effect

### 4.3.3 INVERSE DISTANCE WEIGHTING

The Inverse Distance Weighting (IDW) determines cell values using an inversely weighted combination of a set of sample points (see Chapter 2.2). The two main parameters of this method are radius and power parameter. The radius determines the maximum distance between the cell of interest and the most distant point from the distance data set. The power parameter determines the power on which the inverse of the distance is applied.

To determine the optimal values for each parameter the optimization algorithm is utilized in a way to minimize the mean absolute error of the regression. For both of the parameters, the regression was run for all 9 input measurements, while using 5 different random generator seeds to pick different samples for each number of input samples, while sweeping the parameter.

The radius parameter was swept for values 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9 and infinity. The presented data show no dependency on the radius changes. Based on this, the radius parameter may be neglected in this case. It is set as “infinity” for all following regressions in this chapter. See Figure 4-13 for the comparison of IDW radius parameter, where 11 lines (each for a different radius value, gained as the average of each seed-based data set averaged for each measurement) almost merge into one.

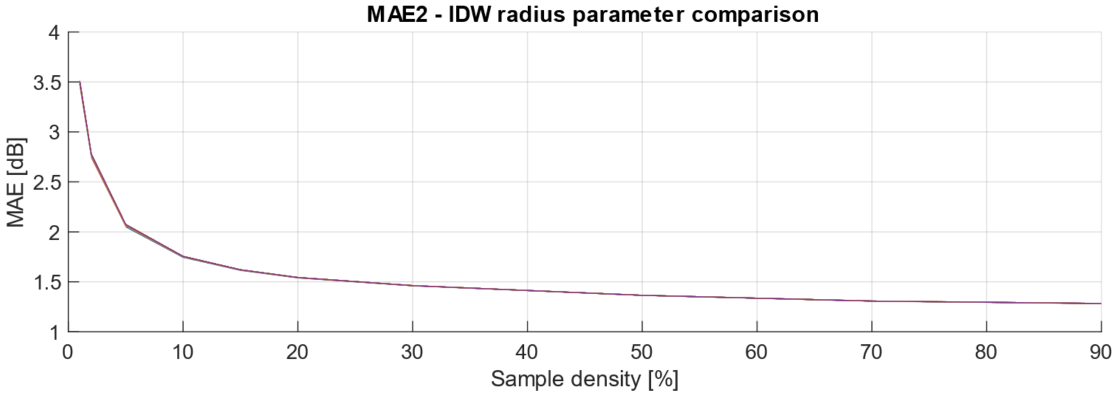


Figure 4-13 IDW - Radius Parameter Comparison

The power parameter has significantly higher impact on the presented data than the radius. See Figure 4-14, where the comparison of power parameter related to MAE is presented. It consists of 28 lines based on the input values for it (ranging from 0.6 to 6 with the step of 0.2). The highest line (blue) stands for power parameter 0.6, the lowest one represents the power of 4.8. With the increasing power factor, the MAE decreases till the 4.8 value, afterwards the value of MAE increases again. This value is therefore considered the optimal value for the following regressions within this chapter, nevertheless slightly reducing or increasing this parameter results in smoother regression and almost the same error. To ensure the precision of this optimization simulation, the regression was rerun in the same manner as the one for the radius.

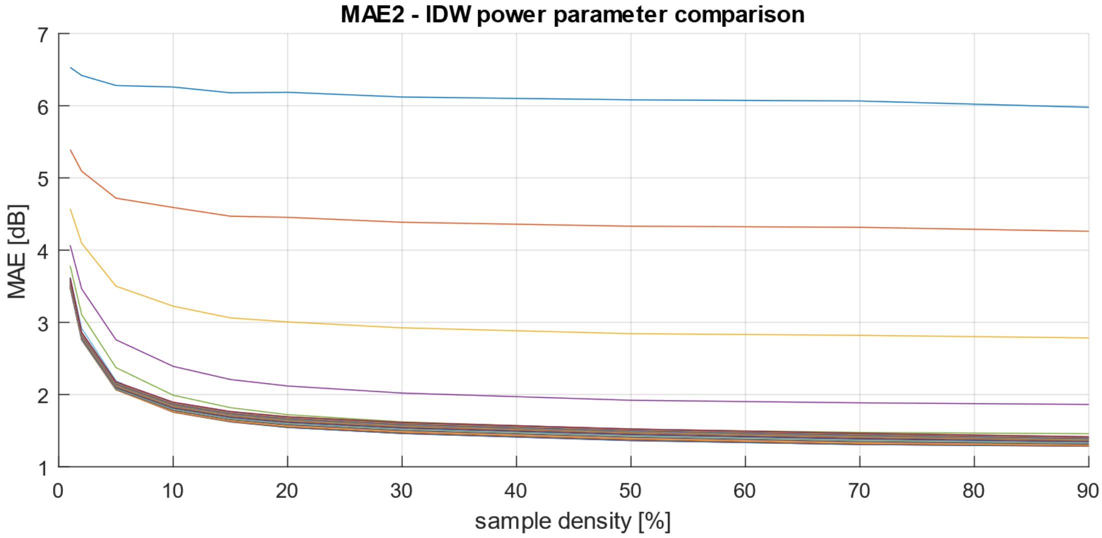


Figure 4-14 IDW - Power Parameter Comparison

One of the major downsides of IDW is that this method in reference coordinates returns the same value as the input, meaning that this method does not possess the attribute of noise filter itself. This aspect of IDW can be overcome by smoothing the data before applying the regression using any of the smoothing methods described above. The combination of smoothing and IDW offers a simple approach to create a viable regression and to decrease noise within the signal. Figure 4-15 shows the comparison in detail of full input samples of a single measurement (yellow line), Gaussian window smoothing with window size of 61 samples applied to the full samples (blue line), IDW with power parameter 4.8 applied to every tenth smoothed sample (green line) and IDW with power parameter 4.8 applied to every tenth sample (red line). Applying IDW to original samples creates random peaks and blips within the regression due to the noise variation, smoothed IDW regression differs only slightly (see stair-like shape of the function) form the noise-supressed signal. The smoothed IDW provides an elegant, transparent and easy to implement solution as the regression method.

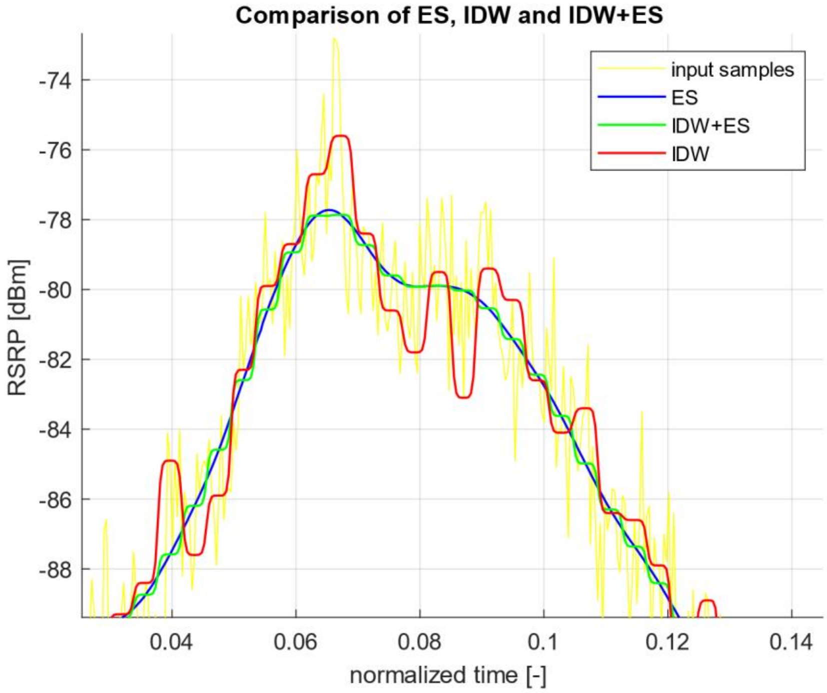


Figure 4-15 Detail of Comparison between ES, IDW and IDW+ES

The prediction capabilities of IDW method with smoothing is shown in Figure 4-16. The figure shows, that the larger the power parameter is, the longer the regression keeps its trend before leaning towards the mean. In case the predicted point is further from any training samples by more than the radius parameter, the regression does not occur (see purple line). In this case, the remaining values are either left without prediction or have to be set (e.g. as a mean of training symbols) using additional algorithm. Nevertheless, either by setting the radius to infinity or implementing a secondary regression condition, the smoothed IDW becomes a fully functional regression method. The power and radius parameter of the method has to be optimized for each implementation directly to properly represent the parameters of regression.

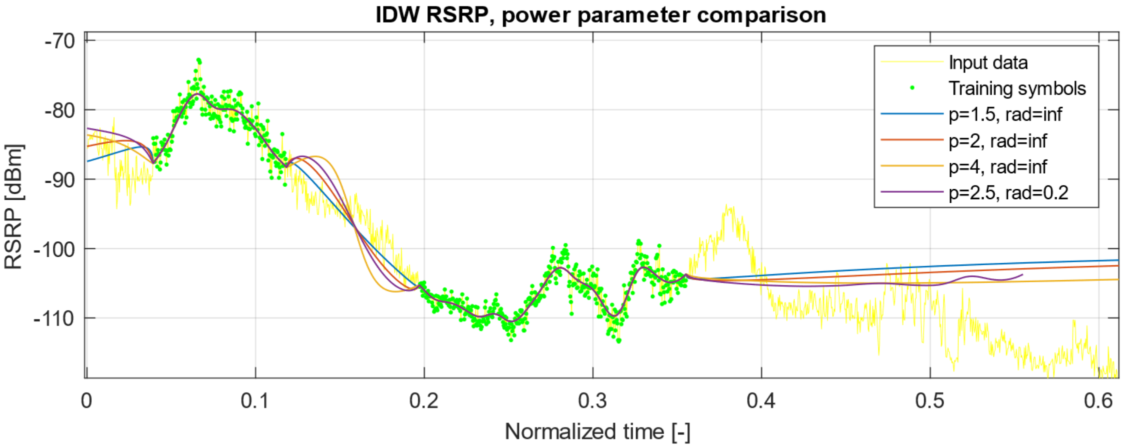


Figure 4-16 Smoothed IDW – Prediction based on power parameter and radius

### 4.3.4 RANDOM FOREST

Random forest (RF) is a supervised machine learning algorithm, which builds a "forest" of decision trees and from such forest chooses the most probable outcome (see Chapter 2.2). This regression method does not possess any predefined behaviour (as IDW) or hyperparameters such as kernel function in GPR. The prediction is only based on decision trees trained on training symbols. Figure 4-17 presents the comparison of RF regression changes with input data thinning. The behaviour of the regression between samples is unpredictable and with lower sample density the regression does not show any predefined behaviour.

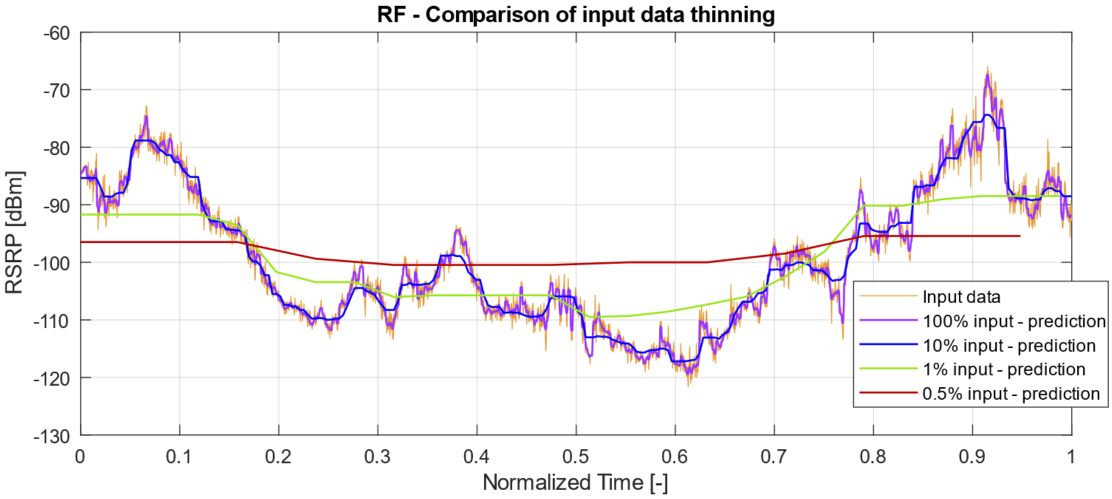


Figure 4-17 RF – Comparison of Number of Input Data

The MAE2 comparison was realized the same way as with previously discussed methods. The random data sample was chosen using different random generator seeds on all 9 measurements while calculating MAE2 depending on the number of created trees. MAE decreases with the growing number of utilized trees, as shown in Figure 4-18 up to the certain number of trees, from which the error stays within the same bounds (the difference in MAE between 1 and 5 trees is bigger than the difference between 5 and 10 trees).

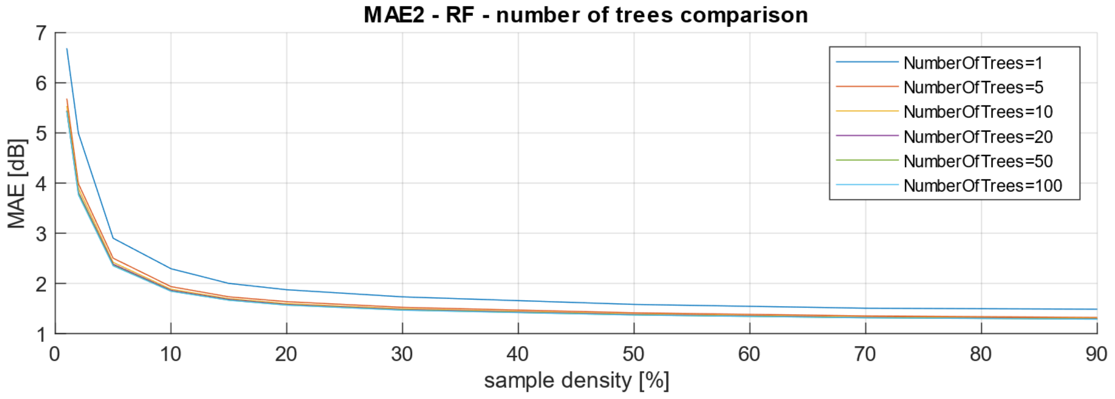


Figure 4-18 RF – MAE – Number of Trees Comparison

Figure 4-19 depicts the application of RF regression on partially removed RSRP samples. The prediction capabilities of this algorithm are limited, staying constant outside the prediction range within the data. The figure shows, that this regression method is not applicable for being "non-predictive" while the data samples are further from each other, staying constant after the last sample (similar to the extension of the linear interpolation method).

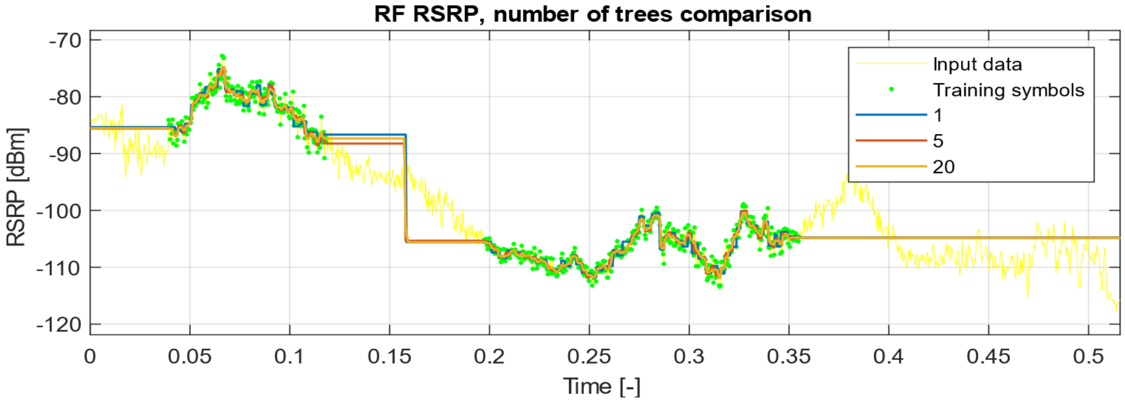


Figure 4-19 RF – Prediction based on number of trees

### 4.3.5 GRAUSSIAN PROCESS REGRESSION

The utilization of Gaussian Process Regression as the regression of choice presents a number of advantages in comparison to the remaining regression methods. Gaussian process regression is realized in Matlab by creating and training a GPR model, which defines the behaviour of the regression.

The created model estimates the exact parameters of the regression itself based on the training data and the hyperparameters [57]. Hyperparameters are the parameters, which have to be given to the model on the point of its creation and are later utilized to predict the parameters of the regression. For GPR, the main hyperparameters are the definition of the covariance function, type of the basis function, mean and standard deviation of the signal and length of the distribution. Parameters such as mean or standard deviation can be defined as the hyperparameters or can be estimated from the input data.

The pre-defined kernel functions are able to estimate the length and standard deviation of the signal from the input data if not defined differently. The choice of kernel function has a dramatic impact on the data prediction within the training symbols and more importantly in the prediction zone. Figure 4-20 shows the prediction capabilities of GPR based on the kernel function of choice. The prediction shows that the exponential kernel keeps the last known value for the longest duration and in between samples behaves almost linearly, squared exponential kernel approaches the mean the fastest (seen also in the cut-off interval from 0.12 to 0.195) and Matérn kernels predict the values with slightly larger spread than squared exponential kernel.

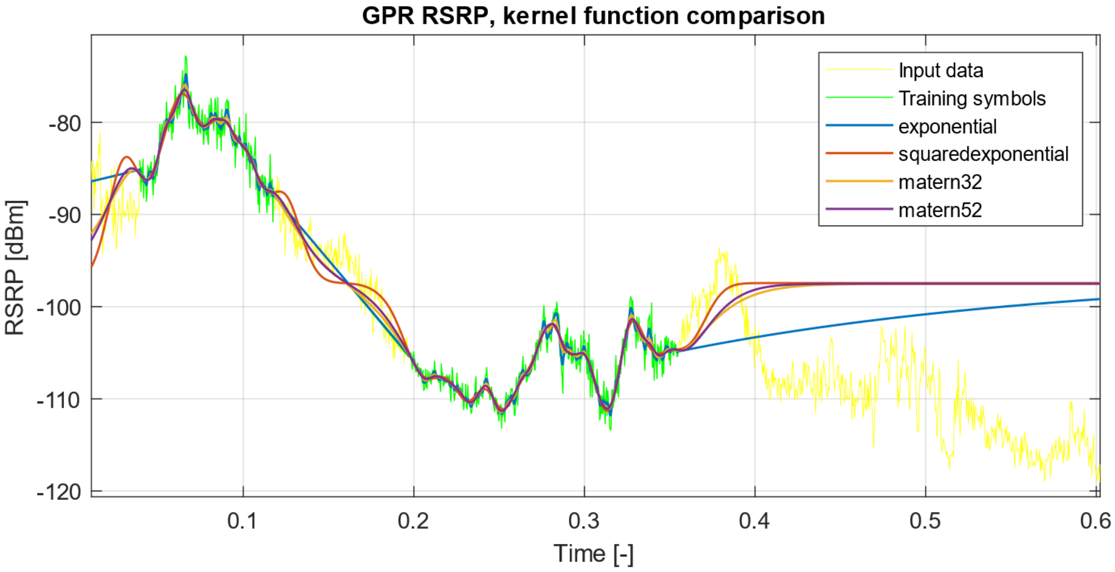


Figure 4-20 GPR - Comparison of Kernel Functions

Figure 4-21 visualizes the behaviour of the kernel function in between the thinned samples. Again, regression using exponential kernel function reacts to the new samples the fastest, while squared exponential and Matérn kernels smooth the data and supress the noise more significantly.

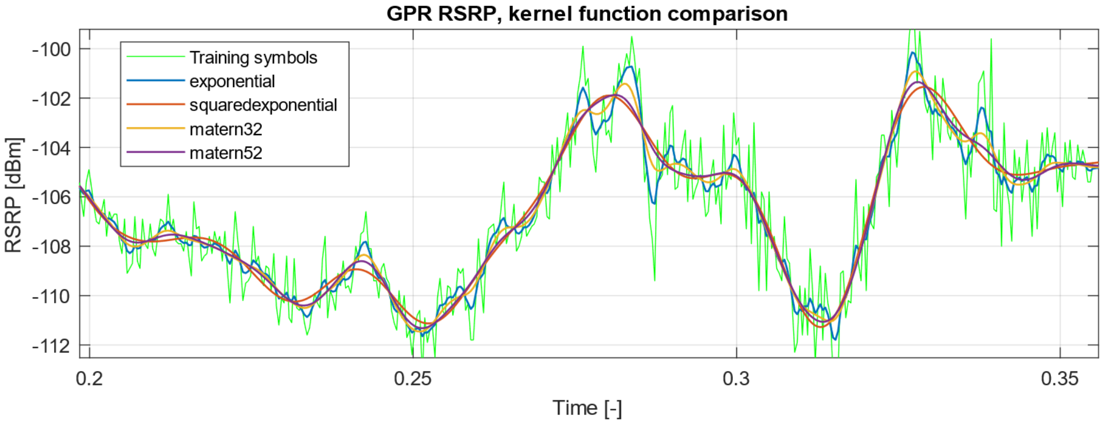


Figure 4-21 GPR - Detail of Comparison of Kernel Functions

The comparison of error of the GRP’s kernel function with varying sample density is shown in Figure 4-22. There, the exponential function shows the smallest MAE2 values, followed by rational quadratic kernel. The estimation was realized, as with previous methods, on 5 random seeds across all 9 measurements. The errors of this function are comparably smaller than the ones calculated at e.g. random forest regression.

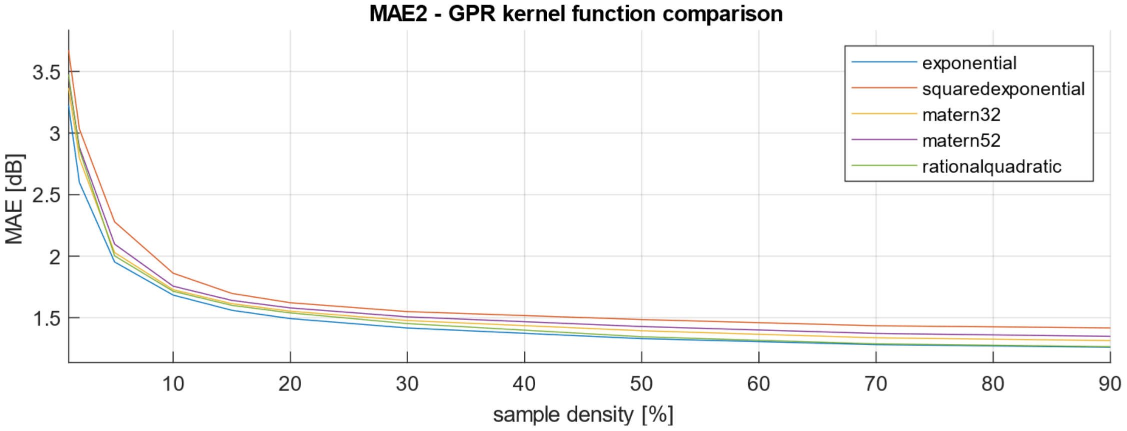


Figure 4-22 MAE - GPR - Comparison of Kernel Functions

## 4.4 COMPARISON OF 1D REGRESSIONS

To choose the regression methods for multidimensional application, factors such as magnitude of error, noise suppression, prediction capabilities, complexity and predictability need to be taken into account.

Figure 4-23 shows all the considered regression methods and compares their noise-suppressing capabilities on fully-sampled reference measurement. The figure shows, that LI (blue line) does not reduce noise, since its values at the output in reference points equal the values of the input. Smoothed IDW (red line) is able to control its sensitivity to noise variation by increasing or reducing the size of the smoothing window. GPR (purple and green lines) is capable to calculate the noise variation from the input samples and its behaviour depends on the chosen kernel function, on the other hand RF (orange line) supresses the noise in rather undirected and unpredictable manner. Since noise suppression is an important aspect of each regression, linear interpolation is not considered the suitable regression method. Exponential smoothing and other smoothing methods are used only in combination with IDW (see Chapter 4.3.2).

The remaining regression methods can be divided into the parametric and non-parametric methods. Parametric methods (smoothed IDW) have directly defined all parameters which define the overall behaviour when calling the regression function. Non-parametric methods such as GPR or RF are pre-defined only by hyperparameters, which calculate the exact parameters of the regression. Non-parametric methods utilize machine-learning and optimization algorithms, which define the final behaviour of the regression using training symbols.

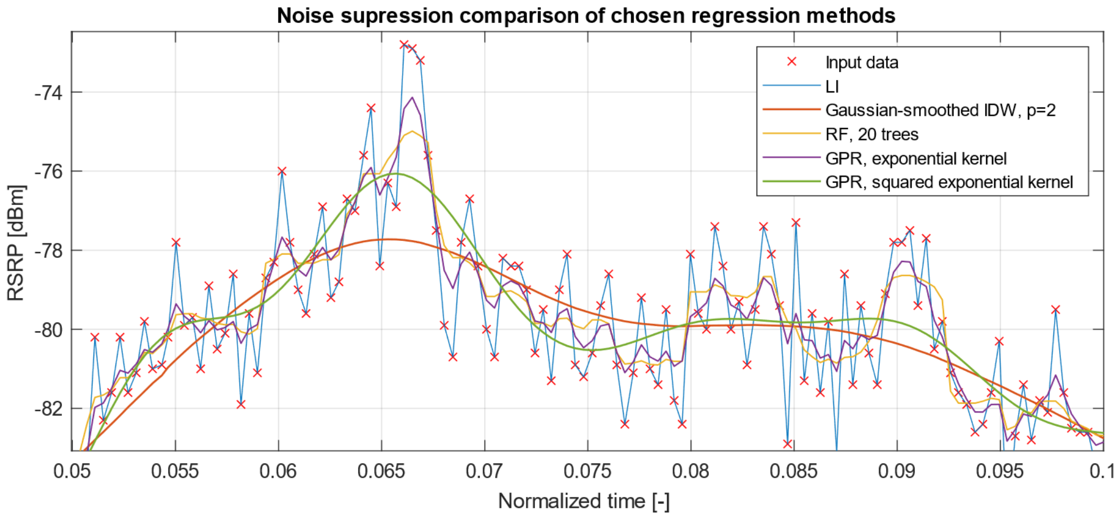


Figure 4-23 Comparison of noise-cancelling capabilities of LI, IDW+GS, RF and GPR

MAE2 (see Chapter 2.3) comparison on reduced samples of chosen prediction methods is shown in Figure 4-24, including 95 % confidence intervals. 95 % intervals represent the spread of values, within which the random result will be with 95 % probability. Technically it means the values between 2.5 and 97.5 percentile. The figures show the confidence intervals are below 0.5 dB for all realizations, when the sample density is above 8 %. The confidence intervals increase with decreasing number of samples due to higher impact of randomness on the resulting regression.

The comparison was realized on repeated measurements using 5 different random generator seeds on each of the 9 measurements. Since the reference values were the unused samples from the individual measurements which include noise, smoothed IDW shows larger errors on high samples due to pre-smoothing using GS. Both RF and GPR have linearly increasing error with decreasing number of samples. Figure also shows, that smoothed IDW crosses 1.5 dB margin at 15 % samples used, exponential kernel GPR at 19 % and random forest with 20 trees at 27 % of samples. This fact shows, that non-parametric, machine learning algorithm's performance deteriorates, when it does not have a sufficient number of training samples, faster than the parametric method, which has directly defined behaviour independent of the number of samples.

This fact is highlighted in Figure 4-25, where the low-sample detail (from 0 % to 20 % of samples) of the three regression algorithms is shown. There, the random forest regression returns the highest level of error and IDW returns the lowest. All regressions show minimal differences in MAE for sample density larger than 20 % of samples, meaning that considering only 20 % of samples to create regressions results in comparable plots as when using 100 % of the samples. MAE at 20 % samples of IDW equals 1.47 dB, 1.5 dB for GPR and 1.58 dB for RF. This conclusion also allows to dismiss confidence intervals. Since NEMO measurements were realized on the 2 km long path through the centre of Vienna with over 2500 samples per measurement (see Table 4 in Chapter 4.1), each sample was taken at least every 0.8 meters. Reducing the samples to 20 % will result in 500 samples per 2 km equal to one sample every 4 meters. At 2 % of samples IDW shows 2.4 dB MAE, GPR 2.6 dB and RF 3.8 dB. Taking only 2 % of samples would increase MAE slightly, but sampling could be realized once every 80 meters (12.5 samples per km).

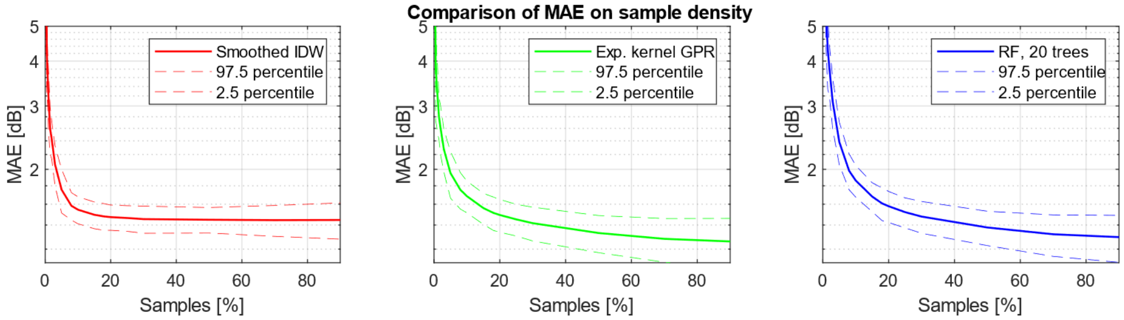


Figure 4-24  Comparison of MAE on sample density for IDW (left), GPR (centre) and RF (right)

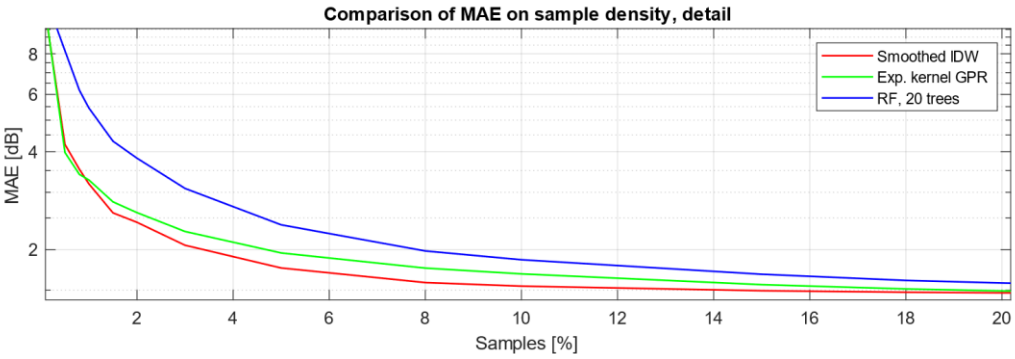


Figure 4-25  Comparison of MAE on sample density for IDW, GPR and RF in low samples

Considering prediction capabilities, GPR is the regression capable of creating various prediction models. In close proximity to the training symbols the kernel function dictates the shape of regression, whereas in further regions the prediction is defined by basis function shape. IDW's prediction capabilities are limited to exponential function from the nearest set of training symbols towards the mean and can be affected only by the power parameter. Random forest presents no reliable predictive capabilities when not given enough training samples. Figure 4-26 depicts the comparison of the mentioned methods and confirms the claims above. Predictability of GPR, based on the hyperparameters chosen, is the highest of all other regressions. IDW must be iteratively tuned to show the desired shape of the regression, but from the power parameter chosen the result is highly predictable. Random forest, as the name suggests, returns "randomly" estimated results, therefore its predictability is limited.

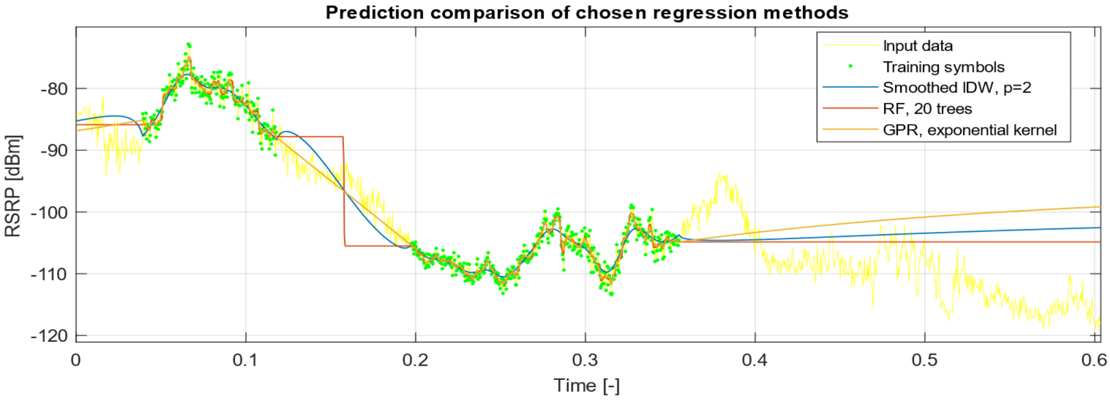


Figure 4-26  Comparison of prediction capabilities of smoothed IDW, RF and GPR

To evaluate the accuracy and sample-reliance of each method, the following algorithm was created (see Figure 4-27). It presents the procedure to evaluate the impact of parameters or hyperparameters on the results of every method. For every sample of the evaluated parameter the algorithm loads the data of the NEMO measuremets one by one. For each of the 9 measurements, the algorithm chooses a percentage (at least 10 different values) and using 5 different random seeds generates random samples as training symbols (of current percentage's size) from all input samples. The regression algorithm with current parameter sample and training symbols from current measurement is then ran and its MAE2 is calculated and saved. Afterwards, each parameter-MAE reliation was calculated and visualized. Every point of the graph was calculated from 45 independent simulations (9 measurements, 5 randomly generated inputs). The example of this procedure's result may be seen in e.g. Figure 4-22.

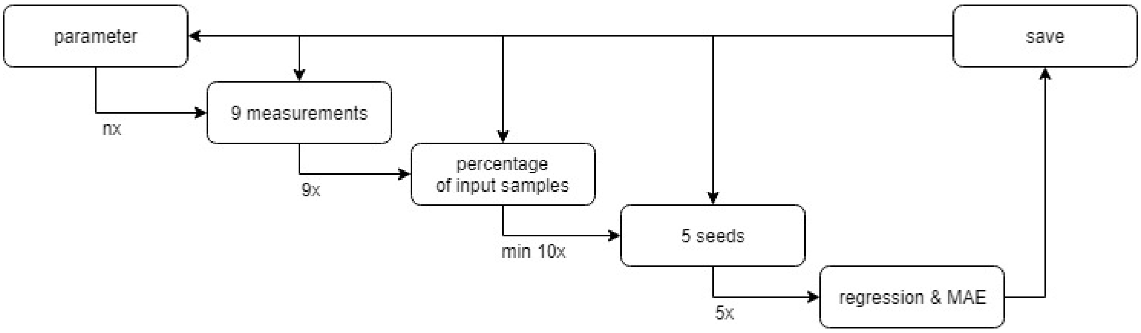


Figure 4-27 Parameter Sweeping Algorithm

The complexity differs for each method. For example, while sweeping kernel functions in GPR, the parameter sweeping algorithm took 126 minutes and 18 seconds in comparison to RF tree count, for which the computation took 3 minutes and 46 seconds. IDW sweeping of power parameter took 34 minutes and 9 seconds. The complexity of IDW depends on the number of reference symbols, the size of the grid and radius parameter. GPR's complexity additionally varies based on the chosen hyperparameters such as regression method and fitting method. Random forest's complexity relies on the number of utilized trees, number of symbols and size of the grid.

Concluding, the two regression techniques that will be implemented for 2D grid are GPR due to its scalability, natural noise suppression capabilities and pre-defined behaviour due to the hyperparameter selection and smoothed IDW for it's straight-forward parametric approach, reliability and simplicity.

# 5 2D ANALYSIS

In this chapter the utilization of chosen regression methods (see Chapter 4) onto a presented input data is described. Each of the regression methods was implemented in Matlab and its parameters or hyperparameters settings were compared. Later in the chapter the mutual comparison is discussed and evaluated for purposes of coverage maps utilization, complemented by its evaluation. At the end the estimated maps are presented.

## 5.1 INPUT DATA

For the purposes of creating an estimated coverage map, the location aware measured data including downlink and uplink speeds, ping, RSRP etc. were used. These data come from two sources, RTR NetTest and NEMO-based measurements (see Chapter 3).

The NEMO measurements were provided by TU Vienna while measured using Keysight NEMO on the route through the centre of Vienna. This measurement was repeated 9 times to gain 9 repetitions of the test (see Chapter 3.2). These data were used as a base for 1D analysis (see Chapter 4) for several regression methods, their parameters evaluation and possible utilization opportunities. In this chapter they serve as a tool to show each regression’s properties in space on previously evaluated data, which additionally contain global trend in signal strength. Using regressions on such data allows further parameter optimization. See Figure 5-1, where the route taken through the centre of Vienna during the measurements was taken. Each measurement point refers to a single measurement of all parameters at given GPS coordinates. It consists of 11518 points. Each of 9 colours of depicted dots refers to a different measurement (9 measurements in total). The depicted area of measurements is approximately 0.6930 × 1.0 km. Code for Map-plotting used in this thesis was based on [58].

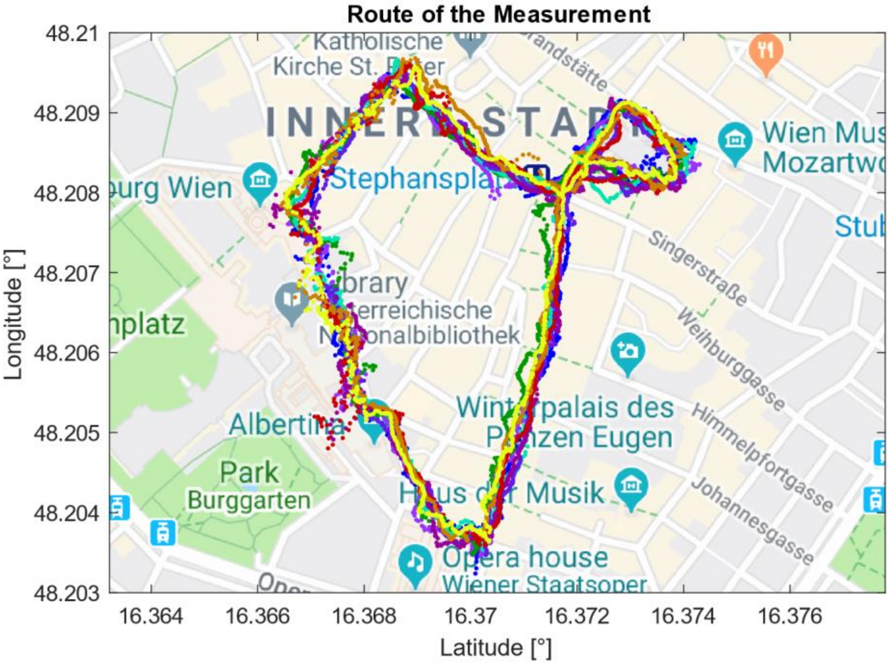


Figure 5-1 Route of NEMO-Based Measurements [58]

The data from RTR NetTest utilized in this thesis compose of the database of all LTE measurements realized over the year 2018 and the self-measured database consisting of measurements in the city area and open-air area. The database gained based on RTR NetTest [18] (see Chapter 3.1) includes a set of measured network parameters for each test. These data are directly connected to the GPS coordinates. The GPS coordinates of NEMO measurements need to be additionally connected with the value of interest based on the algorithm described in Chapter 3.2.2.

Two sets of self-measured data serve as an example of city area with high buildings surrounding narrow streets in the centre of Vienna and open-air area taken in the gardens of Schönbrunn castle, where the area is either with no obstacles or with trees only. Both data sets are depicted by blue crosses in Figure 5-2.

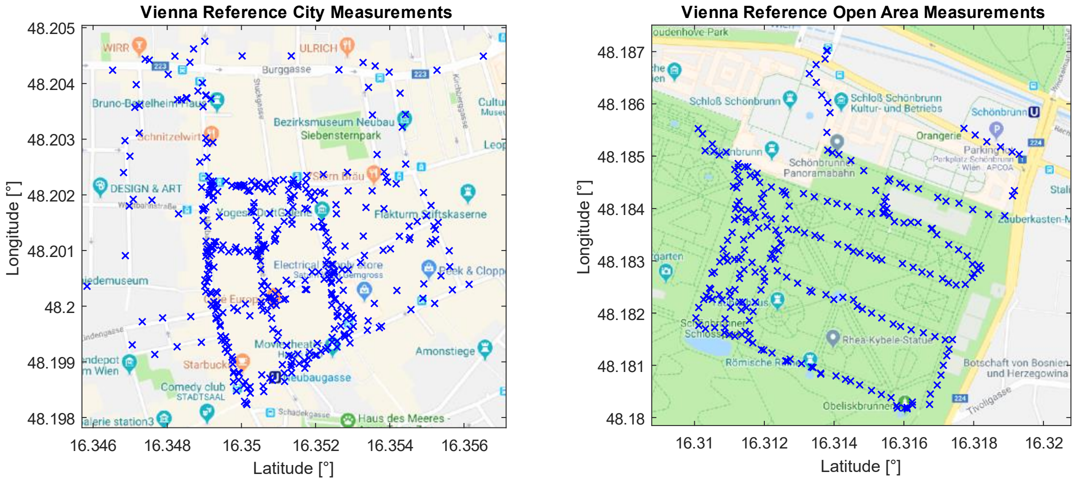


Figure 5-2 Reference Measurement - City and Open Area [58]

Both sets of measurements were obtained using SIM card with unlimited amount of data tariff from H3 operator with download speed limitation of 50 Mbit per second and upload speed limitation of 20 Mbit per second (although there were even higher speeds measured during the testing measurements). To perform this measurement, Samsung Galaxy S8 device was used. The total area of all city-area measurements consisting of 468 measured samples in larger, than the one shown in Figure 5-2 (left). The area shown in the figure consists of the measurements with high density and consists of 340 samples. The open-area database consists of 249 samples over the whole depicted area (right part of the figure) with high density measurements on the area with no obstacles (no trees) consisting of 104 samples. The depicted area of measurements is approximately 0.4798×0.8889 km for the high-density city measurements and the depicted area for open area measurements is approximately 0.7998 × 1.5556 km.

The database obtained from the RTR NetTest [18] includes all measurements by this application for the year of 2018. The dataset used in the scope of this thesis is filtered and includes only of LTE data measured on the devices using A1, H3 and T-Mobile operators (no roaming using users) in the area covering most of the Vienna. This area is depicted in Figure 5-3, where each point refers to a different measurement. Blue dots symbolize the measurements in H3 network, green dots in A1 network and pink dots in T-Mobile network. The depicted area of measurements is approximately 8.8524 × 18.6676 km. The used database consists of 17322 measurements, from which 4315 were realized in H3, 8176 measurements were realized in T-Mobile and 4831 measurements in A1 LTE network. These measurements are used to create a coverage map of Vienna for each of the stated Austrian operators.

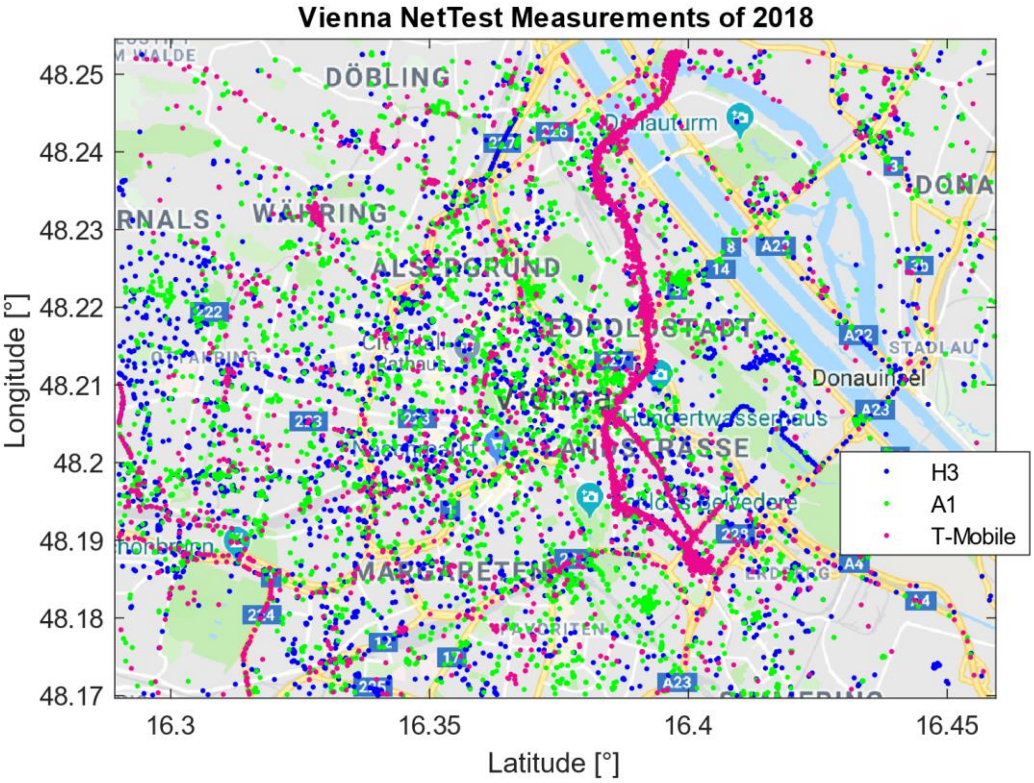


Figure 5-3  RTR NetTest Measurement Database of 2018 [58]

While using various regressions an error may occur due to the usage of degrees of latitude and longitude. The distance of 1 degree in latitude is not equal to the distance of 1 degree in longitude. For example, in geographic area of Austria, 1 degree change in latitude is corresponding to the distance of approximately 111.1 km and 1 degree change in longitude is corresponding to distance of approximately 77.1 km. In case latitude and longitude information was used for regressions such as IDW, an error would be created not only by latitude-longitude distance inequality, but also by specifying some parameters that are reliant on absolute distance (e.g. radius).

To compensate the distance proportion difference between the degrees of latitude and longitude the algorithm for recalculation of latitude and longitude degrees to distance in meters is used (see Figure 5-4). The algorithm loads the latitude and longitude and then, by implementing Heaviside formula, gains the shift from 'zero-zero' coordinate (derived as minimal latitude and minimal longitude value from the input set) to the right and to the up direction for each point of the grid. This function is called 'DegtoM'. The algorithm considers only one-dimensional shift for latitude recalculation, as well as for longitude recalculation allowing to simplify the Heaviside equation (Δlat is considered 0 for longitude recalculation and Δlong is considered zero for latitude recalculation).

GPR in Matlab is able to compensate this shift by using '*ard*' kernel functions, which require higher computational power than standard versions of kernels. Since the conversion to meters is needed and implemented to be used for IDW, '*ard'* kernel functions are not used.

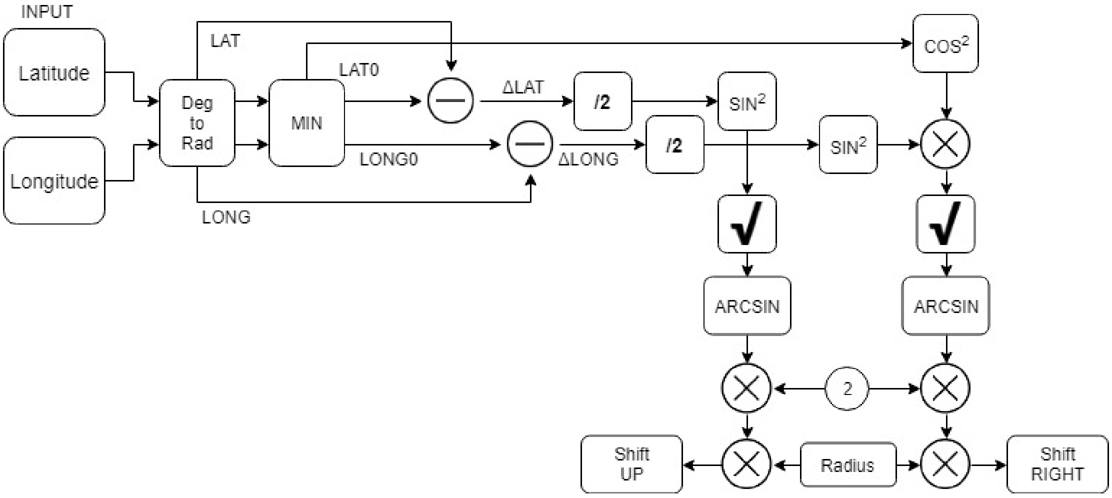


Figure 5-4 Latitude/Longitude Degrees Input Recalculation to Meters

# 5.2 2D REGRESSIONS

In this chapter the regression methods chosen in Chapter 4, Inverse Distance Weighting and Gaussian Process Regression, are utilized and their settings, parameters, prediction accuracy ability and computation complexity are presented. Later, they are used for creating of coverage maps of network parameters as the main result of this thesis.

The introduction to each method and its basic theory can be found in Chapter 2.2, followed by Chapter 4, where 1D analysis and additional extension of theory is given. MAE serves as a basic metric for evaluating the performance and for comparison of used methods. This evaluation is shown at the end of this Chapter.

## 5.2.1 INVERSE DISTANCE WEIGHTING

In this chapter, the utilization of Inverse Distance Weighting (IDW) regression method for coverage maps creation is evaluated, using several input data sources. To explain the influence of each parameter on the resulting regression, the same set of training input data is used as in Chapter 4. Next, the set of self-performed measurements using NetTest is used to evaluate the regression's ability to cope with highly diverse data at the small area vs. its ability to work with low diversity data. At the end of this chapter, the IDW regression is used to create the coverage map to evaluate the network's performance.

As determined in Chapter 2 and Chapter 4, IDW determines cell values (the density of which depends on the regression grid) using an inversely weighted combination of a set of training points. The influence of each training point on the resulting regression depends on two main parameters, radius and power parameter. The radius determines the maximum distance between the cells that influence each other. The power parameter determines the power on which the inverse of the distance is applied.

To show the parameter's influence on the performance maps, the NEMO-based measurements were used. Utilization of NEMO measurements allows to choose optimal parameters due to the existing global trend within the data (worse RSRP in upper regions, better in the lower, see figures below), varying distances between the sample points and empty areas, evaluated strictly using the regression's prediction capabilities.

Figure 5-5 shows the comparison of the two spatial IDW regressions with power parameter 4 and radius set to infinity on a single NEMO measurement. The left figure shows the regression applied on the original, unsmoothed data, whereas the data in the right figure were smoothed using Gaussian smoothing with 61-sample window before applying the regression. The smoothed IDW has lower spread of values (see the colour bar next to the pictures) due to the noise-cancelation smoothing, which reduced the extreme values within the measurement. Although in the full-sample regression the smoothing does not have such heavy impact, for the regressions with lower sample density the smoothing plays significantly bigger role, as discussed in Chapter 4.

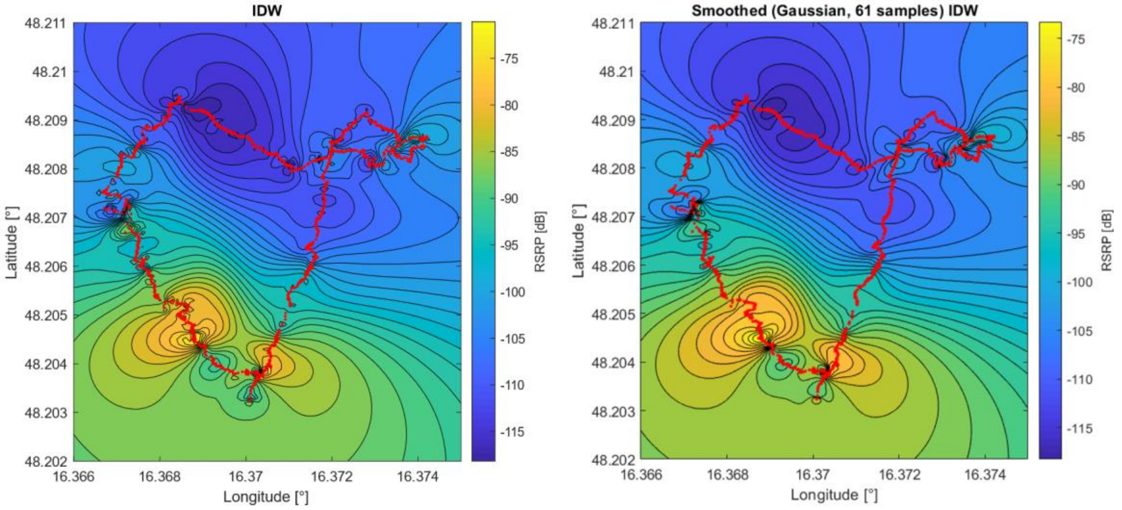


Figure 5-5 Comparison of IDW vs. Smoothed IDW

Figure 5-6 depicts the impact of different power parameter choice on the same data with plain IDW. The left figure shows the regression with power parameter set to 1, resulting in grid, which reacts to the reference points only in their closest vicinity, rapidly reaching mean values of the reference symbols with the increasing distance from the reference samples. IDW realization with power parameter 8 is shown on the right side of the figure. Here, the power parameter chosen was higher than the grid required, resulting in rapid changes of the regression at the borders between the samples or areas with reference samples with different measured values (see the vertical change in the grid in the middle of the larger path). The choice of higher power parameter should be considered when creating a regression with sparse samples, within which each sample is supposed to have impact on a larger area around itself. The optimal power parameter for the current regression is approximately 4, as derived in Chapter 4 and shown in Figure 5-5 and has to be selected individually for each utilization of IDW based on the input data properties and distribution. This choice of power parameter allows for smooth transitions of regression results in the area between the measured samples.

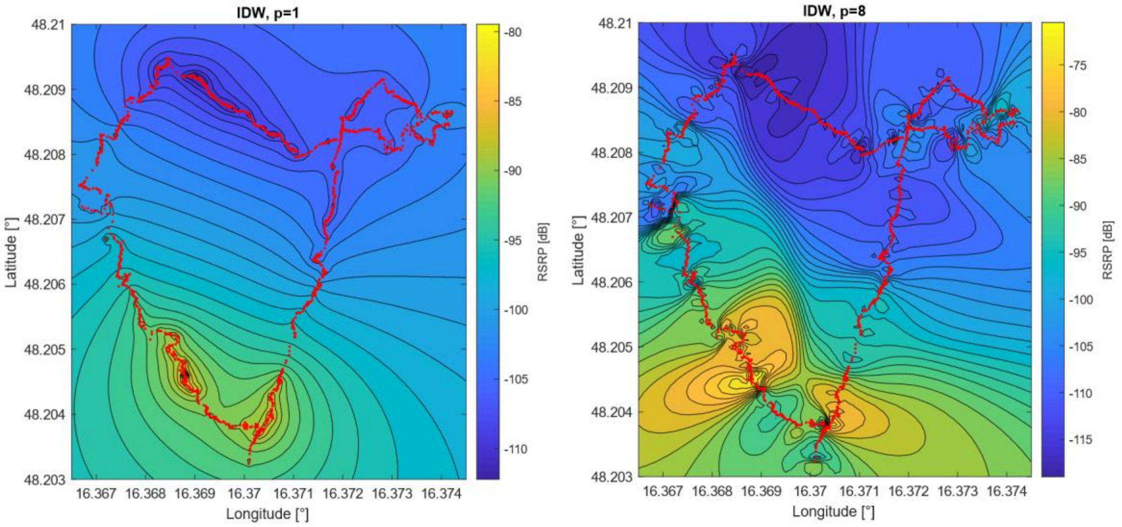


Figure 5-6 IDW Power Parameter Comparison

Choosing radius parameter other than infinity may create unreasonable predictions of the resulting regression, in case the reference samples are not spread across the whole area for the regression. On the other hand, the infinite radius may increase the computational complexity while performing regressions of a huge grid. As shown in Figure 5-7, where IDW was realized with power parameter 4 and radius 150 metres, the edge areas of the regression are influenced only by several closes (almost 150 m away) samples, obtaining their values. In right bottom part of the regression (yellow part) it holds the values of the nearest training points, but closer to the training points the values of regression are already lower (greener area between two yellow areas). Similarly, in the middle of the training data route, the area influenced by samples with lower RSRP values (dark blue) meets with the area influenced by higher RSRP values, creating a sharp edge with high slope change. While creating a spatial regression with reasonably distributed trainee data, radius parameter allows to create a local trend within the data, not considering the points that could not influence the current point of the regression.

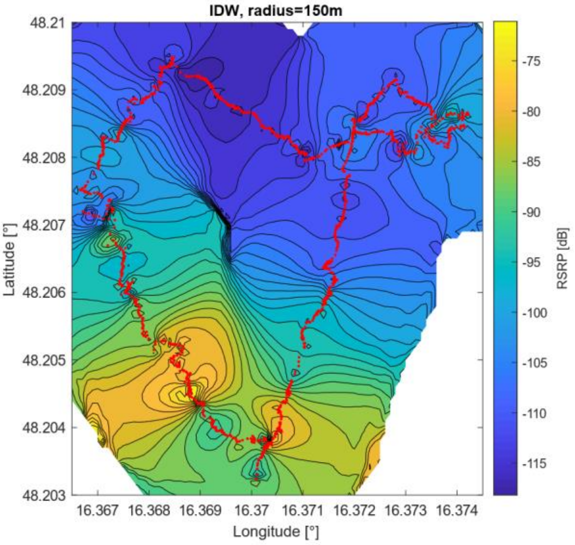


Figure 5-7 IDW Limited Radius Example

A set of self-performed measurements using NetTest is used to evaluate the regression's ability to perform in various environments. The highly diverse data at the small area is evaluated while applying the regression on the city area reference measurement in Vienna centre. Its ability to cope with low diversity data is evaluated while being applied to the open-area reference measurement from Schönbrunn castle park. The noise-suppression smoothing is required for NEMO measurement results, as each data sample reflects the currently measured parameter value including the fluctuations due to the momentary noise variation. As the RTR tests results reflect the evaluation of the whole measurement process in which the NetTest evaluation suppresses the noise within its own algorithm, the smoothing of data is not required. Additionally, the measurements of RTR data do not share a common timing information with equidistant spacing to allow for per-sample smoothing.

All the evaluations of regressions presented below are the result of the averaging between 10 regression realisations, based on 10 random seed generators (from 10 to 100 with step 10), which define the initial values of the pseudo-random generator. For the purposes of precision evaluation, the MAE metric is used in the same manner as in MAE2 scenario (see Chapter 2.3). The set of input measurements data is divided into two smaller sets, the training dataset and the evaluation dataset. The regression is then run with the input of training set while the evaluation set (consisting of the complement of the full dataset) is used to calculate MAE. The green crosses in the following figures represent the evaluation dataset and the red ones represent the training dataset.

The power parameter set to approximately 4 was determined in Chapter 4 as an optimal value for high density measurement inputs. The results of such settings are satisfying with the given input as well. Figure 5-8 (right) shows the regression applied to the high-sample reference measurement. The regression in the high-sample direction (along the street) changes dynamically and holds its trend. In low-sample direction (area filled by buildings) it dynamically varies, reflecting the differences of signal strengths between the two parallel streets and in the direction without measurements holds the last known value for sufficiently long interval. In comparison, the left figure representing IDW with power parameter 2 deteriorates too fast towards the mean. Both right and left figures were realized with full sample input.

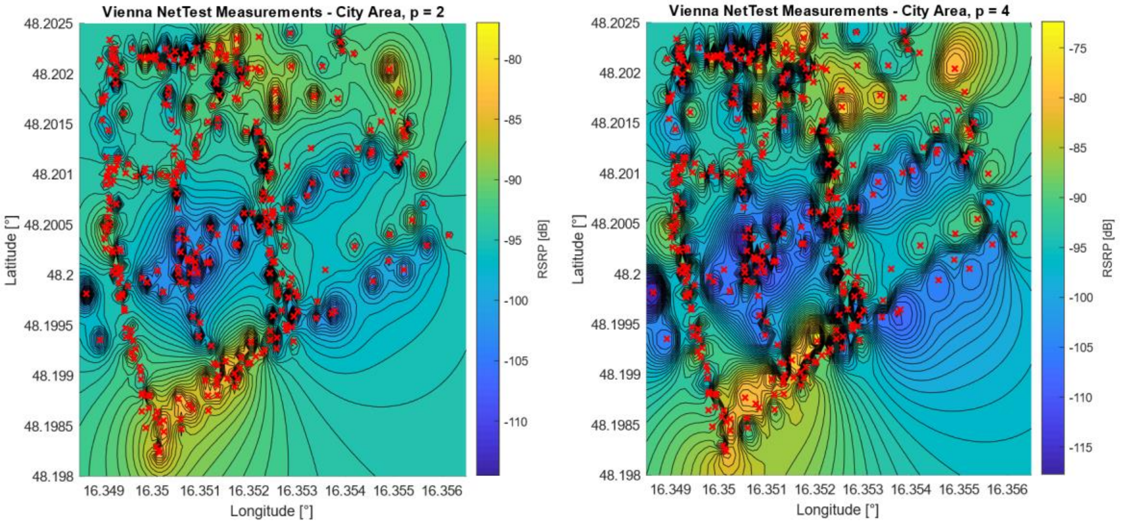


Figure 5-8 IDW – Comparison of Power Parameter in Vienna City Area

The same setting of IDW as above was applied to the open area reference measurement with 100 % samples. Figure 5-9 (right) depicts the regression with power parameter 4 to the open-area data, showing rather steep transitions of signal strengths between the two paths taken through the park (the edge between dark blue and light green). The real-world performance is not reflected, since in the open-area the transitions of signal strength are continuous since there are no strong attenuators in the area (only several low trees). The figure on left presents the IDW with power parameter 2, for which the regression drops to the mean values too soon. Slightly reducing the power parameter from 4 for open area applications reduces the steepness of the transitions, resulting in better reflection of the real-world signal behaviour.

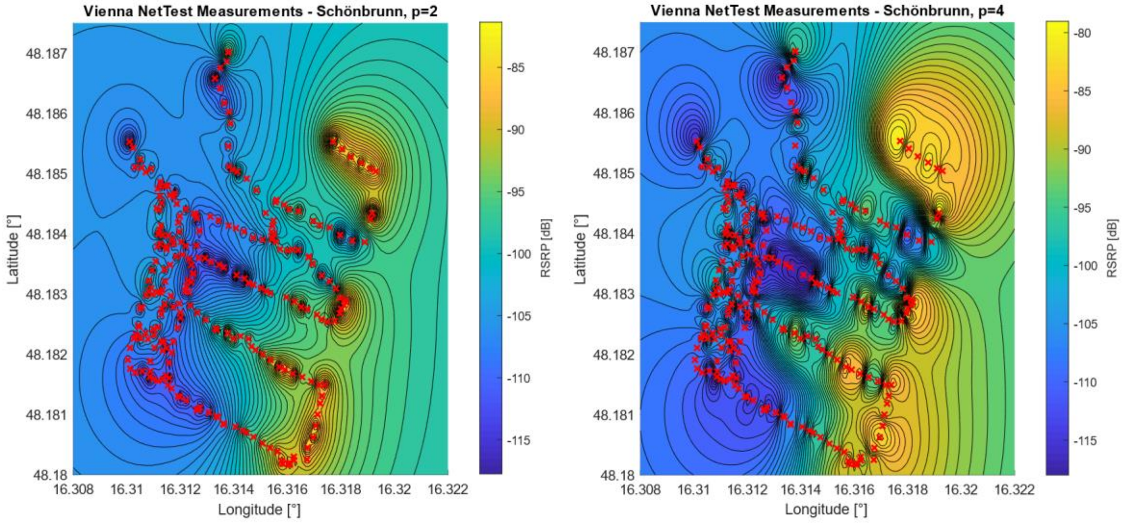


Figure 5-9 IDW – Comparison of Power Parameter in Open Space Area (park)

The comparison of IDW regression on reduced samples in the city area is shown in Figure 5-10. The prediction map on left was realized using 20 % of the input samples (red crosses) and MAE was calculated at locations of the remaining, unused 80 % of the samples (green crosses). The resulting mean average error was 7.134 dB. The figure on the right side depicts the realization of IDW regression using 80 % of the input samples as the reference data. The resulting MAE for this case is 7.119 dB confirming the conclusions drawn in Chapter 4 about the stability of MAE and sample density. 7 dB of MAE refers to the local signal strength variation and better regression results are impossible to obtain. The figures and MAE calculation show, that the IDW regression reliably reflects the real-world signal strength even at lower samples. The dark blue areas on the resulting performance map reflect areas with reduced signal strength, allowing the network operator to locate and improve the coverage within those areas.

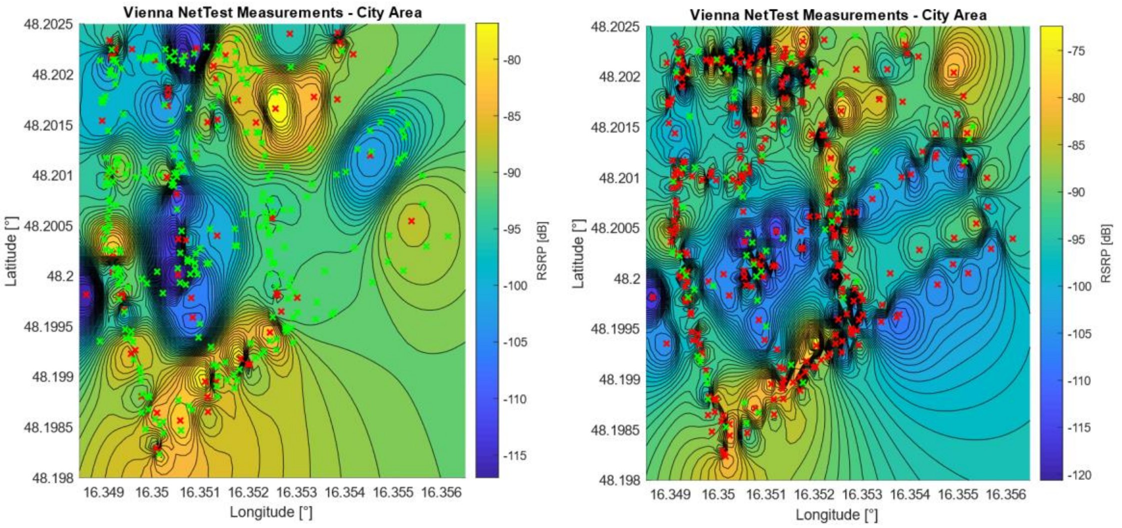


Figure 5-10 IDW – Comparison of Input Amount in Vienna City Area

The sample-density dependency of IDW in open areas is depicted in Figure 5-11 and the resulting MAE shows, that the relative change of MAE is larger. The power parameter for open area evaluation was reduced to 3 (see paragraphs above). The figure on left shows the realization of IDW with 20 % of the samples as input, resulting in 3.778 dB MAE. The figure on right depicts the regression with 80 % input samples and 2.863 dB resulting MAE.

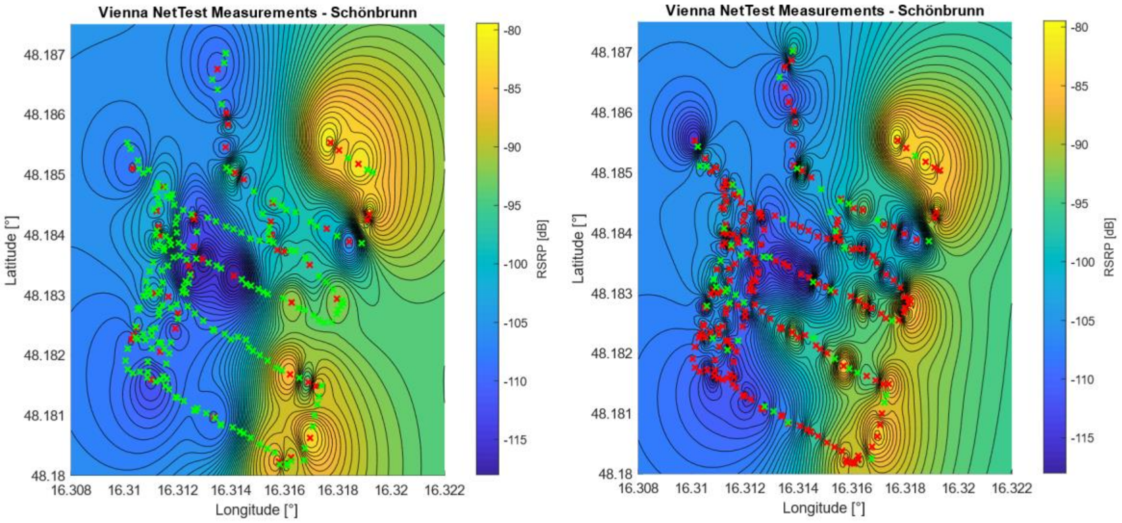


Figure 5-11 IDW – Comparison of Input Amount in Open Space Area (park)

Figure 5-12 shows the application of IDW with p = 3 on the purely open area (with no obstacles at all) large 130×360 metres. The performance map on the right is evaluated using 50 % samples as input, resulting in 2.887 dB MAE. On the left, the regression is realized using only 5 % of the input samples (5 samples). The MAE calculation shows 3.716 dB error. The resulting MAE for the regression with 95 % of samples as input is 2.667 dB. Reducing the samples to 5 % results in 1 dB increase of error above the noise variance margin.

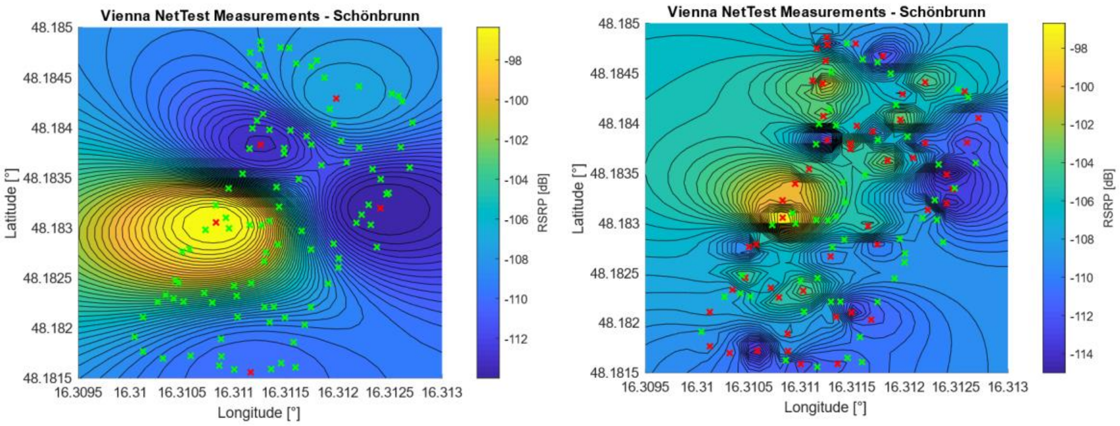


Figure 5-12 Comparison of Input Amount in PURELY Open Space Area

The comparison on MAE for IDW with power parameter 4 for city area and 3 for open area is shown in Figure 5-13. The figure shows, that the MAE for both scenarios stays within the noise variance above 15 % input samples for open area measurements and above 3 % of samples for city area. The calculation of sample density per squared kilometre to reach minimum error threshold is shown in Equation 5.1:

$$D_{min} = \frac{p \cdot N}{S \cdot 100} \tag{5.1}$$

Where $p$ stands for the threshold percentage, $N$ for the number of reference measurements and $S$ stands for the area in squared kilometres on which the measurements were realized. The resulting sample densities equal 32 measurements per squared kilometre for city areas and 29.7 measurements per squared kilometre open areas. Concluding this chapter, for both scenarios the maximum accuracy of the IDW prediction is achieved with reference sample density of 30 evenly distributed measurements per squared kilometre.

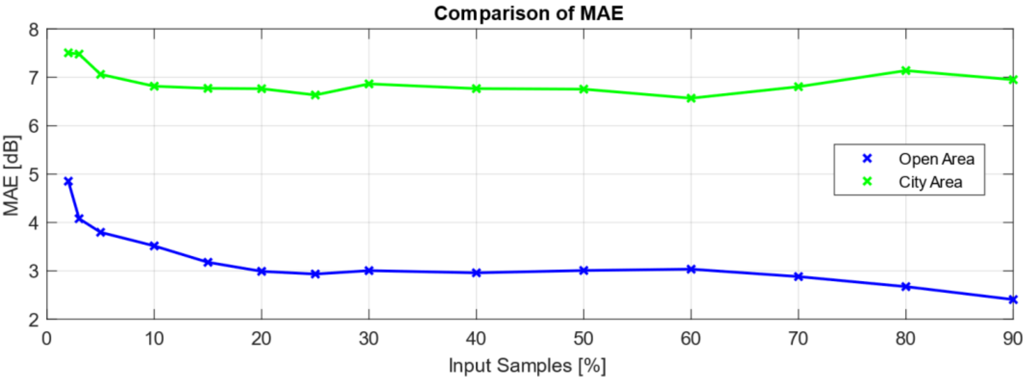


Figure 5-13 IDW - MAE on Input Samples

### 5.2.2 GAUSSIAN PROCESS REGRESSION

This chapter presents the utilization capabilities of GPR for the creation of spatial performance and coverage maps. To explain the influence of the discussed hyperparameters on the resulting regression in 2D, the same set of training input data from NEMO measurements is used as in Chapter 4. Next, self-performed database of RTR measurements for both city and open area is used as the basis for evaluating the method in high sample-data density, as well as in low sample density scenarios. At the end of this chapter, the GPR regression is used to create the coverage map to evaluate the network's performance.

As presented in Chapters 2 and 4, GPR is a non-parametric machine learning approach, the output of which is predefined only by the input samples and the hyperparameters. To show their influence on the regression result, the NEMO-based measurements are used. There is the global trend within the data (worse RSRP in upper regions, better in the lower), varying distances between the sample points and empty areas, evaluated strictly using the regression's prediction capabilities.

Figure 5-14 shows the impact of choosing different basis functions on the resulting regressions. The regression was realized with the squared exponential kernel function, with low covariance distance to highlight the impact of basis choice. Constant basis underlays the regression with constant trend within the data, which is based on the values of the input.

The choice of constant basis (top left figure) is adequate in case there is an unknown trend within the data or in case the trend is globally undefinable using other functions. The lack of global shape of the regression can be compensated by increasing the impact and the range of the kernel function. In case the basis function is set to “none” (top right figure), the regression automatically considers constant basis as 0. The value of the basis can be predefined, which is usable in case the global trend within the data is constant and known beforehand. Linear kernel function (bottom left figure) derives the linear trend within the data based on the input samples, resulting in tilted plane underlaying the data. Setting the basis function to “pureQuadratic” (bottom right figure) results in curved plane with quadratic shape underlaying the regression. The choice of this basis reflects the single source of the parameter value (e.g. eNodeB) within the considered area, as the dominant or only source of the value within the grid. Additionally, the quadratic basis considers regular fading of the parameter in all directions. As the global trend within the data is previously unknown for all data sources considered within this thesis, constant basis function will be considered as the most viable and the shape of the regression will be compensated by increasing the impact of the kernel function on the output grid.

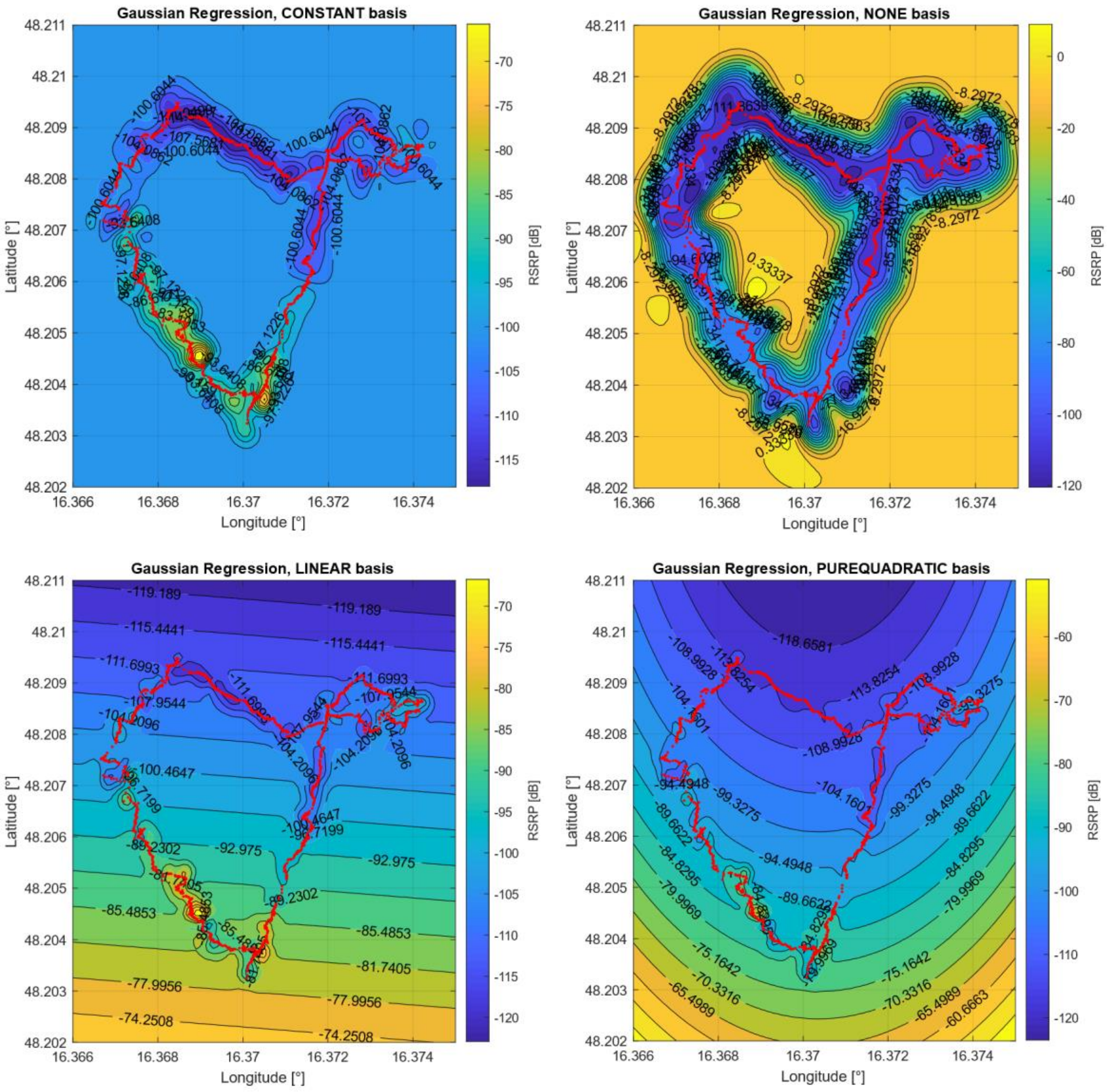
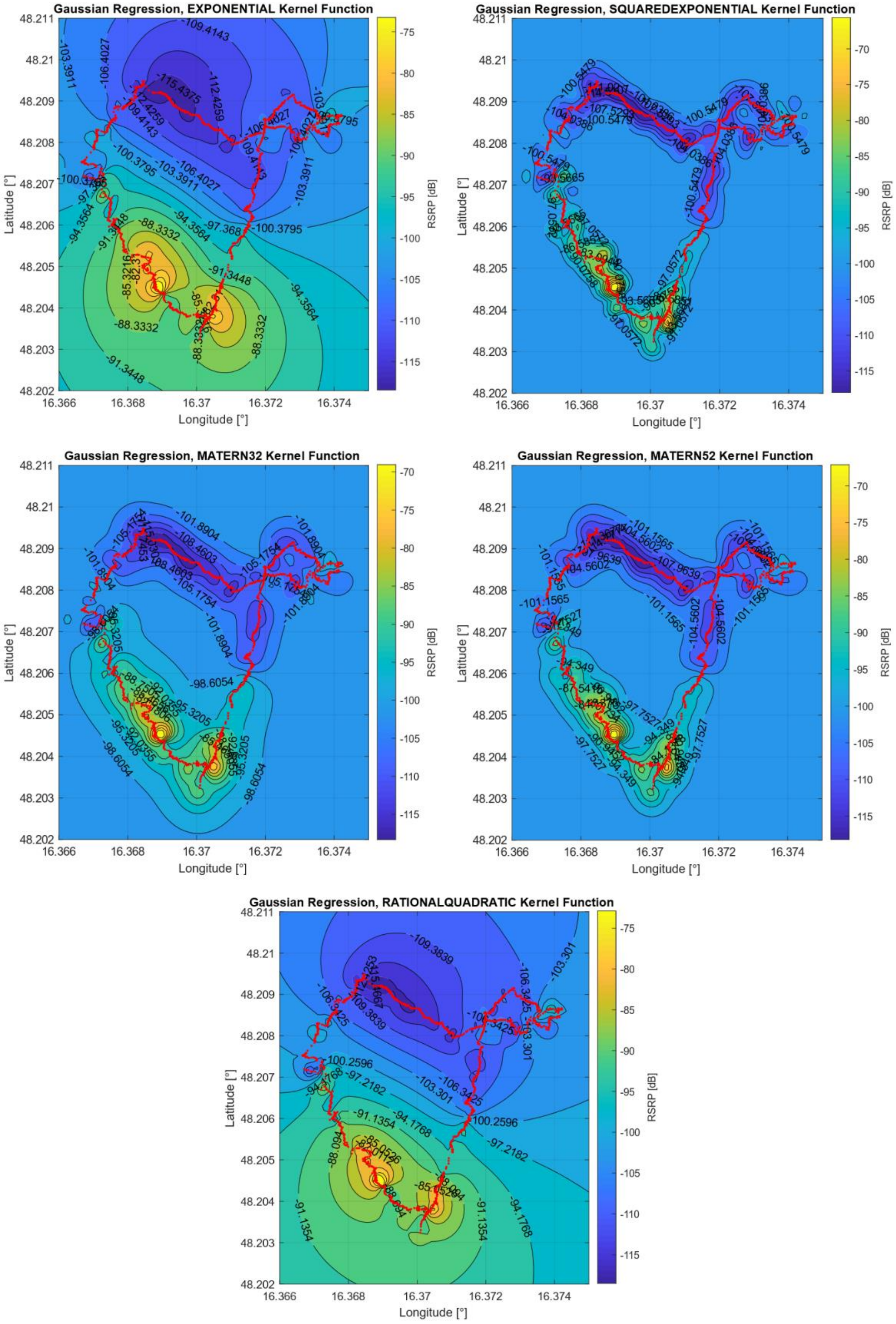


Figure 5-14 GPR - Comparison of Basis

Figure 5-15 depicts the impact on the resulting regression while using five different pre-defined kernel functions. The parameters of the kernel functions, such as characteristic length scale (similar to the radius parameter for IDW) or signal standard deviation, were evaluated from the training samples by the algorithm. The characteristic length scale is calculated as the mean standard deviation of input sample coordinates and the signal standard deviation is evaluated as standard deviation of input sample parameter values divided by the square root of 2. The top-left figure depicts the regression using exponential kernel function, which results in the regression with spread values around each sample point. Visually the similar impact on the surrounding points has the rational quadratic kernel with automatically estimated parameters (see bottom figure). Squared exponential kernel function shows the least impact on its surroundings (as in Chapter 4), followed by Matern52 and Matern32 kernels.


Gaussian Regression, EXPONENTIAL Kernel Function

Gaussian Regression, SQUAREDEXPONENTIAL Kernel Function

Gaussian Regression, MATERN32 Kernel Function

Gaussian Regression, MATERN52 Kernel Function

Gaussian Regression, RATIONALQUADRATIC Kernel Function

Figure 5-15 GPR - Comparison of Kernel Functions ('constant' Basis)

In general, the kernel function expresses the similarity of the regressed result point to the values of predictor, therefore how much does one point react the other. Its base is usually the Euclidian distance $\boldsymbol{r}$ between the points $\boldsymbol{x_i}$ and $\boldsymbol{x_j}$ corresponding to all points at the grid (see Equation 5.2), signal standard deviation $\sigma_f$ and characteristic length scale $\sigma_l$. Equation 5.3 shows the formula for calculating the covariance matrix using Exponential Kernel function.

$$\boldsymbol{r} = \sqrt{(\boldsymbol{x_i} - \boldsymbol{x_j})^T \cdot (\boldsymbol{x_i} - \boldsymbol{x_j})} \tag{5.2}$$

$$k(\boldsymbol{x_i}, \boldsymbol{x_j} | \sigma_f, \sigma_l) = \sigma_f^2 \exp\left(-\frac{\boldsymbol{r}}{\sigma_l}\right) \tag{5.3}$$

Equation 5.4 shows the formula for calculating the covariance matrix using Rational Quadratic Kernel Function.

$$k(\boldsymbol{x_i}, \boldsymbol{x_j} | \sigma_f, \sigma_l) = \sigma_f^2 \left(1 + \frac{\boldsymbol{r}^2}{2\alpha\sigma_l^2}\right)^{-\alpha} \tag{5.4}$$

Where $\alpha$ is a positive-valued scale-mixture parameter, the third hyperparameter of this kernel function (1 by default).

Although it is possible to explicitly set signal standard deviation and characteristic length as the regression function input, the optimizing algorithm considers these parameters as the initial values, adjusting them based on the reference data. In case there is a hidden trend within the data that was not detected with the default kernel parameters, the initially set kernel function parameters may enable to find it (e.g. high frequency signal).

Another way to impact the spread of the kernel function is to specify the standard noise deviation $\sigma_n$. The optimizing algorithm optimizes the value of standard noise deviation in the same way as it does with kernel function parameters. It is possible to enter the command for constant standard noise deviation, disabling the algorithm to optimize $\sigma_n$ and as such explicitly increase or decrease the kernel function spread. This operation is case-sensitive and requires individual approach for each dataset and every regression to ensure comparable results.

As the kernel function is used as the method for predicting the network performance in locations without existing measurements, the exponential and rational quadratic kernels are the two candidates for the kernel function of choice. Exponential kernel's shape is highly similar to the one of Gaussian kernel's (as is its formula), being only less sensitive to scale length parameter. Rational quadratic kernel is a less computationally extensive variant of the Gaussian with adjustable scale length parameter [59]. Its shape is equivalent to the normalized sum of mane Gaussian kernel function with different length parameters [60]. To choose the more appropriate covariance function, the two considered solutions were swept for MAE metric across sample densities, for city area and open area measurements. Figure 5-16 shows, that the two kernel functions return almost identical errors across the sweep for scenarios. The rational quadratic function is chosen as the kernel function of choice for its lower computational complexity.

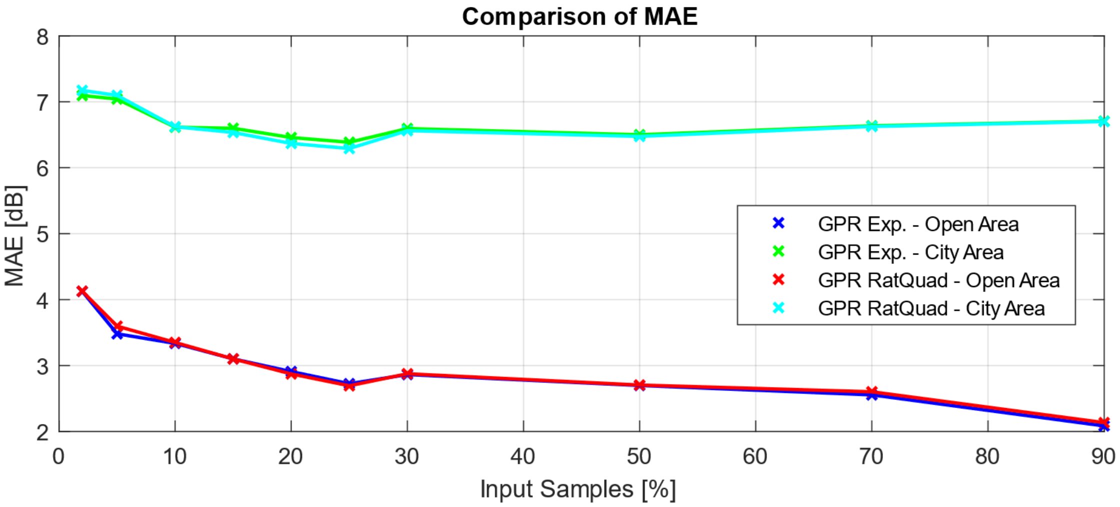


Figure 5-16 MAE Comparison of Kernel Functions

To evaluate the distribution of the input data, the binned distributions of the RSRP, as shown in Figure 5-17 (left), was fitted a Gaussian distribution (green line) and windowed Gaussian distribution (red line). The value of RSRP is considered in absolute value to enable fitting the desired distributions. As the central tendency of the data changed over the coordinated measurement, the RSRP distribution does not resemble Gaussian. The figure on right shows the distribution of RSRP of all 9 NEMO measurements in the small interval of the measurement, where the mean of data remained the same. The resulting distribution strongly resembles Gaussian, concluding that RSRP during the NEMO measurement exhibits normally distributed behaviour with changing mean over the duration of the testing.

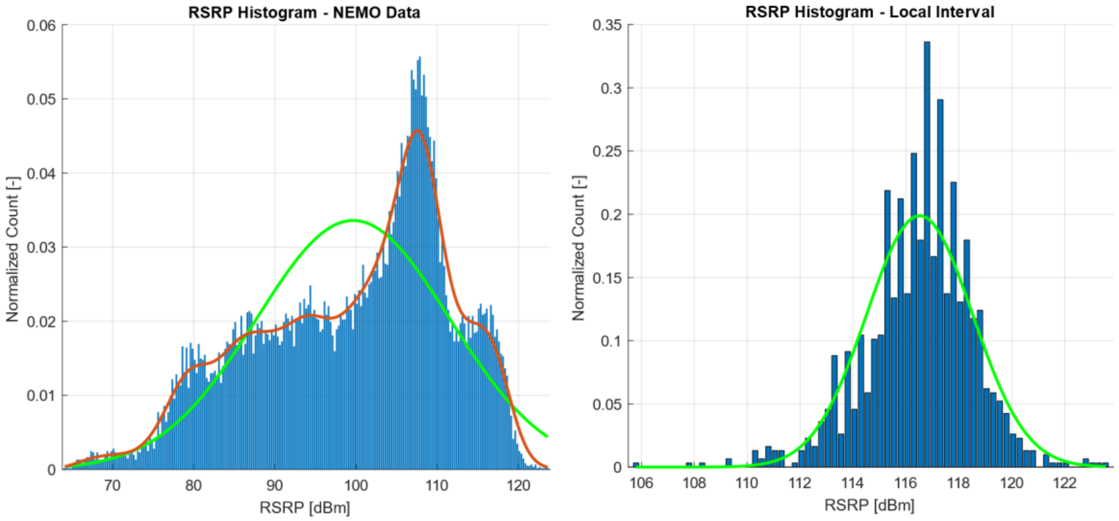


Figure 5-17 RSRP Histogram - NEMO data, Full/Partial Samples

To confirm whether RSRP (in absolute values) indicates Gaussian behaviour, the results of the two reference tests, both in city area and in open area, were plotted in bar graph and compared to fitted normal distribution, as shown in Figure 5-18. The reference distribution approximates the distributions of the RSRP for both city area (left) and open are (right) significantly. Additionally, the Gaussian distribution of RSRP was confirmed by Anderson-Darling [61] test on samples in purely open area (see Figure 5-12).

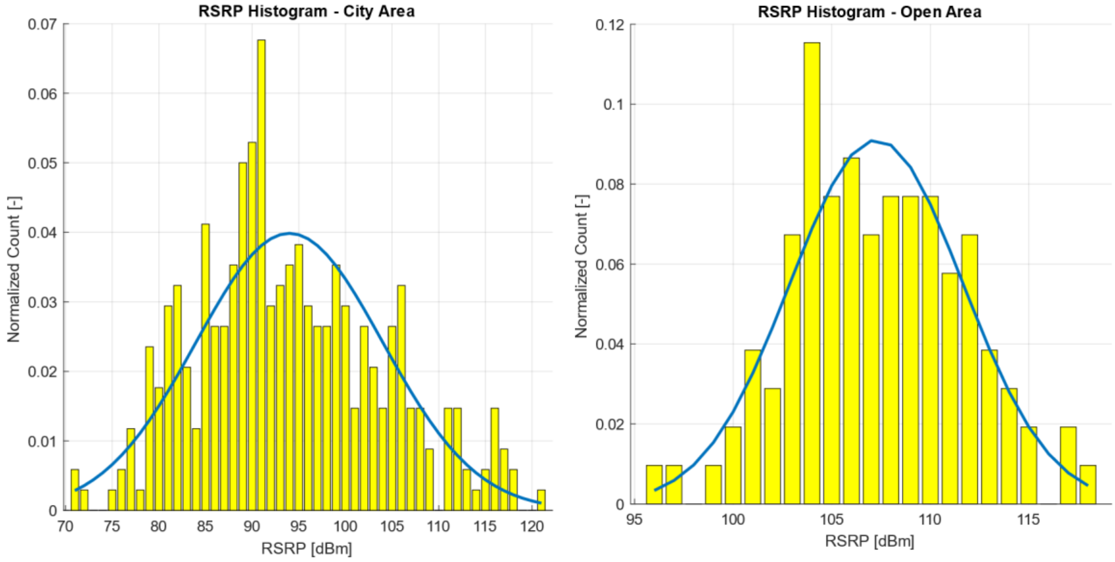


Figure 5-18 Comparison of RSRP Histograms - City/Open Area

Figure 5-19 depicts the RSRP performance map on reference measurements data in the city (left) and open areas (right). The regression was created using constant basis and rational quadratic kernel function. Both realizations of the regression have smooth signal transitions and sufficiently large prediction range. It can be derived from the Figure 5-16 it can be derived, that GPR MAE has the minimum threshold at the same, or slightly lower samples than IDW, concluding that Gaussian Process Regression method with rational quadratic kernel function reaches minimum error matric at 30 evenly distributed samples per squared kilometre.

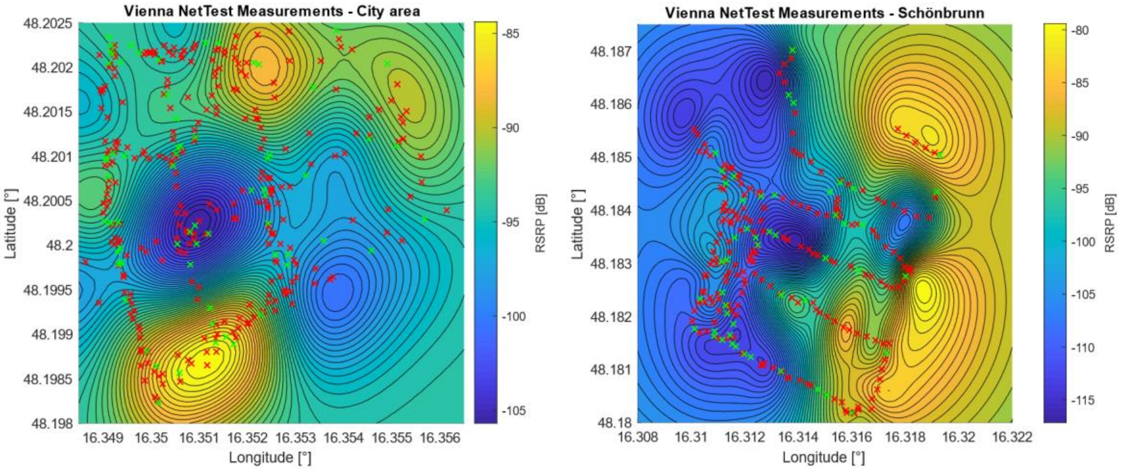


Figure 5-19 GPR - City/Open Area Comparison

## 5.3 COMPARISON OF 2D REGRESSIONS

In this chapter, the utilization of GPR and IDW is compared in terms of such aspects as error metric or resulting regression characteristics. The direct comparison of error is realized by evaluating the error on the two reference measurements, followed by the application of the regression methods onto the crowdsource-based data.

Figure 5-20 shows the resulting MAE comparison of IDW and GPR based measurements. The settings of GPR were constant basis function and rational quadratic kernel for both city and open area evaluations. The radius of IDW was set to infinity for both areas, the power parameter was set to 4 for the city area and 3 for the open area. The figure shows, that the error or GPR is lower on the whole range of axis for both realizations except for 2 % input samples in city area, where the error of GPR is larger. The smaller measured error of GPR caused by the non-parametric nature of this method, which is able to iteratively optimize its parameters based on the input samples to minimize the potential error. The calculation of the resulting MAE was realized by a sweep, done by randomly choosing a set of input data while using 15 random generator seeds, for each point of the graph. The remaining (unused) samples were used for MAE calculation. This is the same procedure as explained in Chapter 2.3 and used in previous chapters.

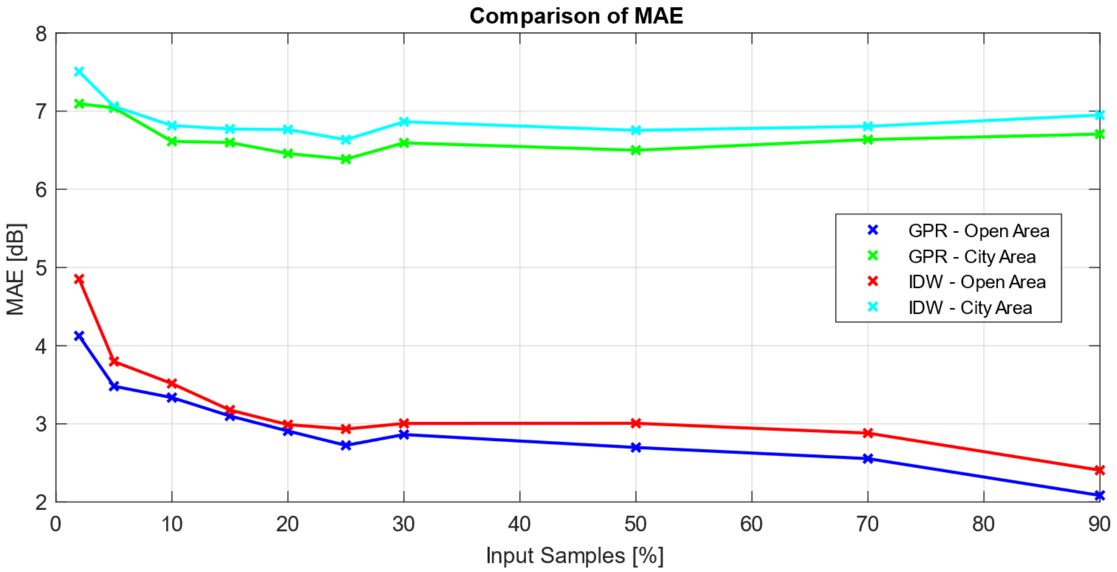


Figure 5-20 Comparison of GPR and IDW based on MAE, Open/City Area

Figure 5-21 shows the application of GPR on the measurements from the crowdsource-based database of RTR in the centre of Vienna. The figure shows, that the distribution of samples is not consistent in space, resulting in “blank” areas. In such locations, GPR evaluates the resulting RSRP as the value close to the mean of the input samples. MAE of GPR was estimated as 6.540 dB, while utilizing 95 % of samples as the regression input. The density of samples in the figure is 310 samples per squared kilometre, which is more than ten-fold higher than the error threshold derived in the previous chapter. Nevertheless, large amounts of measurements are frequently grouped together, reducing the effective number of samples.

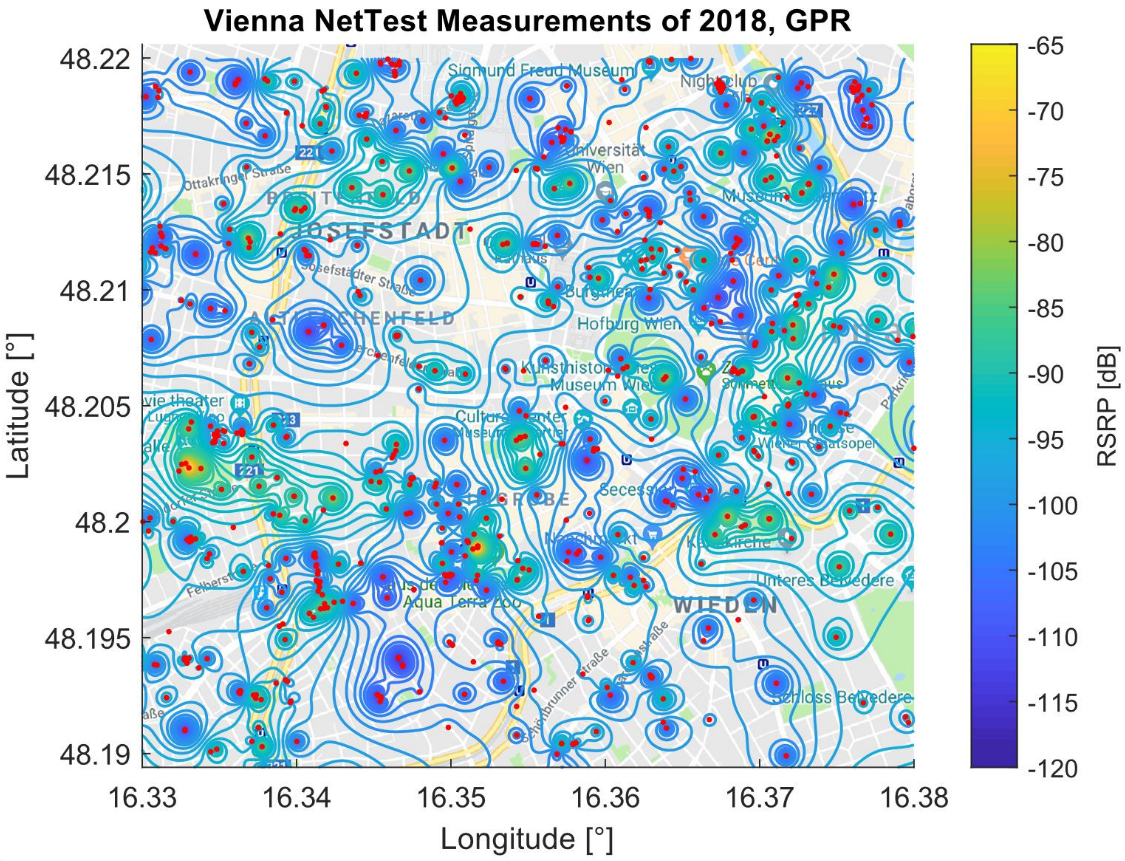


Figure 5-21 Vienna NetTest measurement, GPR, H3

After applying the established IDW regression to the crowdsource-based data from RTR, several observations were made, which result in adjusting the previously established findings. The power parameter 4 for IDW in city areas reliably describes the shifts of signal strength between the adjacent streets in case the measurements are frequent. As shown below, the density of measurements of the RTR's database that is available in the centre of Vienna is inconsistent, resulting in frequent edges in open spaces. The power parameter is therefore adjusted to 3 for crowdsource-based performance evaluation maps. The variation of radius parameter did not have any positive impact on the data, creating inconsistent fluctuations when too small and having no visible effect when increased. The IDW with power parameter 3 and infinite radius applied to the crowdsource-based data from the centre of Vienna is shown in Figure 5-22. Reducing the power parameter decreases the steepness of the edges between the two neighbouring samples and effectively reduces MAE from 7.211 dB (for power parameter equal to 4) to 6.927 dB.

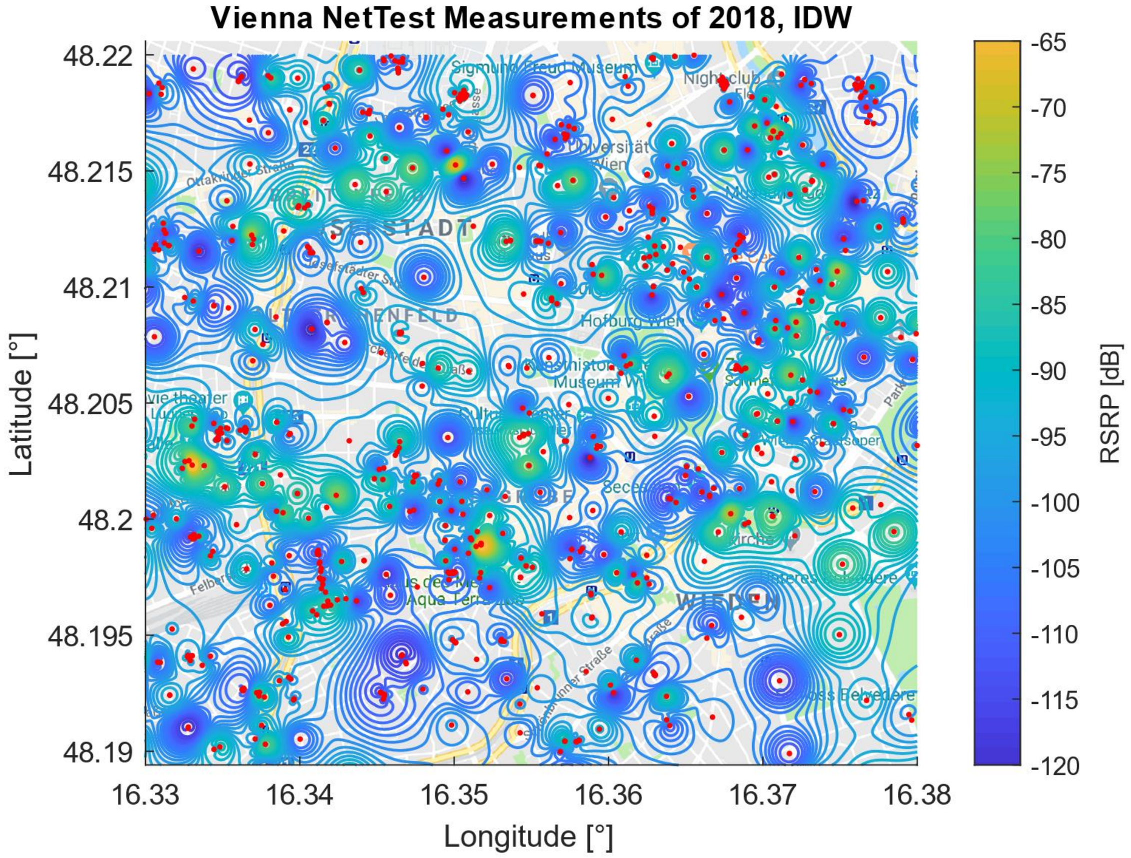


Figure 5-22 Vienna NetTest measurement, IDW, H3

By comparing the results and the figures, several core differences between the two solutions can be highlighted. The resulting MAE of GPR is smaller by 0.38 dB than the MAE of IDW. The GPR performance map drops faster from the reference values to mean values than the IDW (for which the spread of circles is larger around each of the red points). While IDW takes all samples into consideration equally, GPR smoothens the resulting grid, effectively reducing the impact of the extreme values on the final regression.

## 5.4 PERFORMANCE COVERAGE MAPS

The final evaluation of RTR-based data and their utilization to create accurate prediction maps is realised, followed by the examples of the prediction maps for different network parameters. First, the part of the Austria's area, that can be reliably evaluated using the created analysis. The accuracy of the applied method is evaluated, and the final findings are discussed. Before creating performance maps of other parameters than RSRP, their distribution was evaluated, compared to the Gaussian, described and commented in the paragraphs below. This analysis was realized for RTR reference measurements, as well as the measurements from the crowdsource-based database. All map bases displayed in figures within this chapter are based on function from [58].

To estimate the possible coverage density using the created method, the percentage of Austria's area covered by the RTR NetTest measurements is derived. The validity interval of a measurement until it becomes irrelevant first has to be established. Since for both GPR and IDW the error threshold is 30 evenly distributed measurements per squared kilometre, this information serves as a basis for determining the single measurement coverage, resulting in 33 333 squared metres coverage per measurement. To reflect theimpact of each point on the surrounding area, each point is considered a circle, its radius is easily calculated as $r = \sqrt{S/\pi} = \sqrt{33\,333/\pi} = 103$ metres, where S stands for the area. The circle has been drawn around each data point and their total area compared to the total area of Austria. The resulting coverage density of Austria is 1.68 %, densely covering only the centres of the large cities. Due to this fact, the regressions are realized on the filtered data to fit the area of Vienna with sufficient number of tests. The visualization of the performance map is therefore focused on the part of the Vienna centre.

The downlink speed distribution of the self-made measurements using H3 operator's network for both open area and city are shown in Figure 5-23 left and right. Neither of them resembles normal distribution. Additionally, most of the city measurements were tariff-limited (tariff limitation of 50 Mbps), creating a peak at the right side of the distribution.

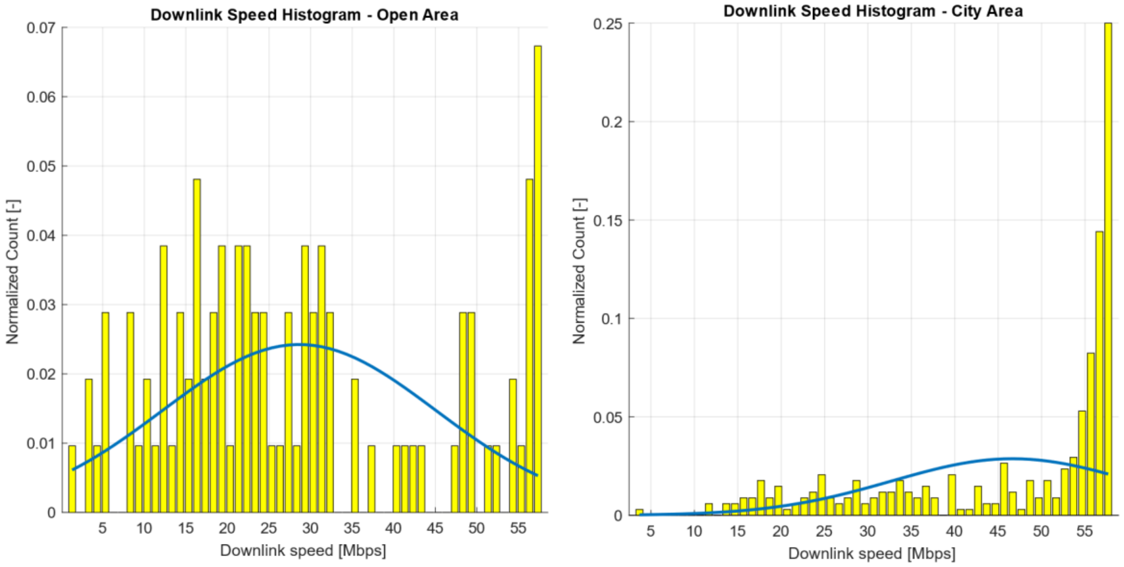


Figure 5-23 Downlink Speed Histograms - Open/City Area

Figure 5-24 depicts the uplink speed distributions for the two reference measurements, which again are not Gaussian. The measurements in the city area (right figure) were tariff limited to 20 Mbps.

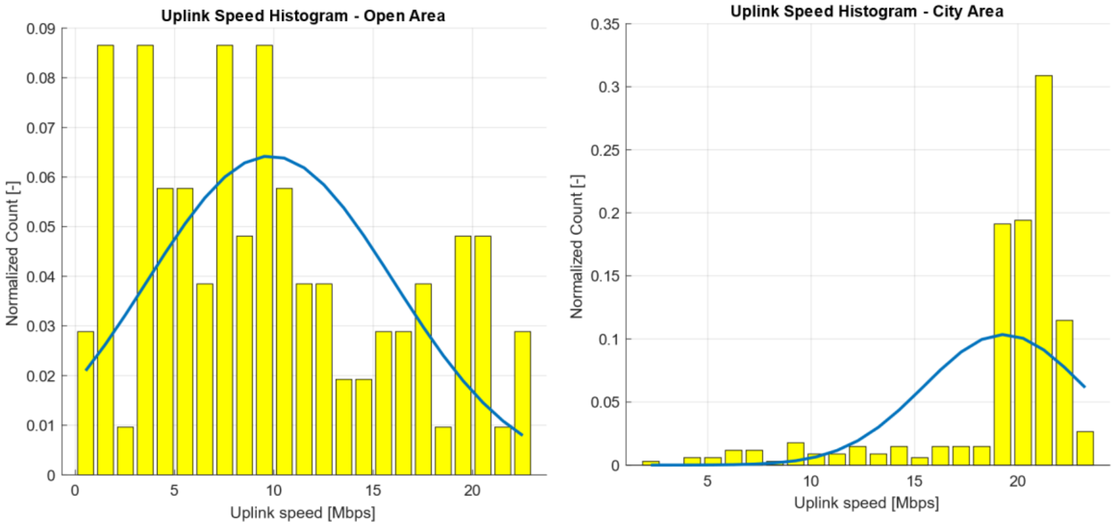


Figure 5-24 Uplink Speed Histograms - Open/City Area

Figure 5-25 shows the distribution of uplink (left) and downlink (right) results of the RTR measurements in H3 operator's network in the area of Vienna. The peaks in the values of tariff limitation (20 Mbps and 50 Mbps for downlink, 10 Mbps and 20 Mbps for uplink) are visible. These distributions are not Gaussian.

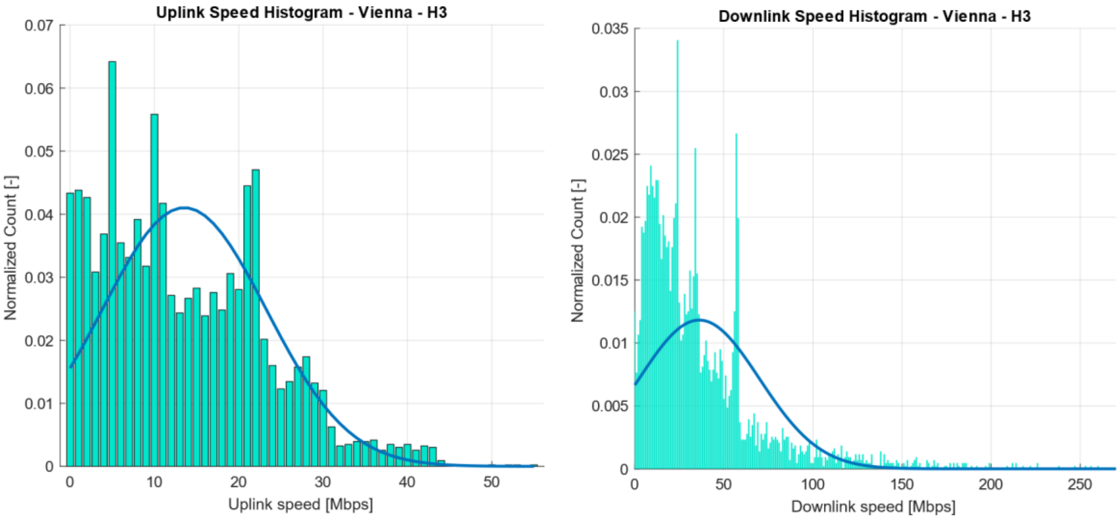


Figure 5-25 H3 - Uplink/Downlink Histogram, Vienna

The distribution of the uplink and downlink of the measurements within A1 network is shown in Figure 5-26. The distributions are non-Gaussian and tariff-limited peaks are also visible.

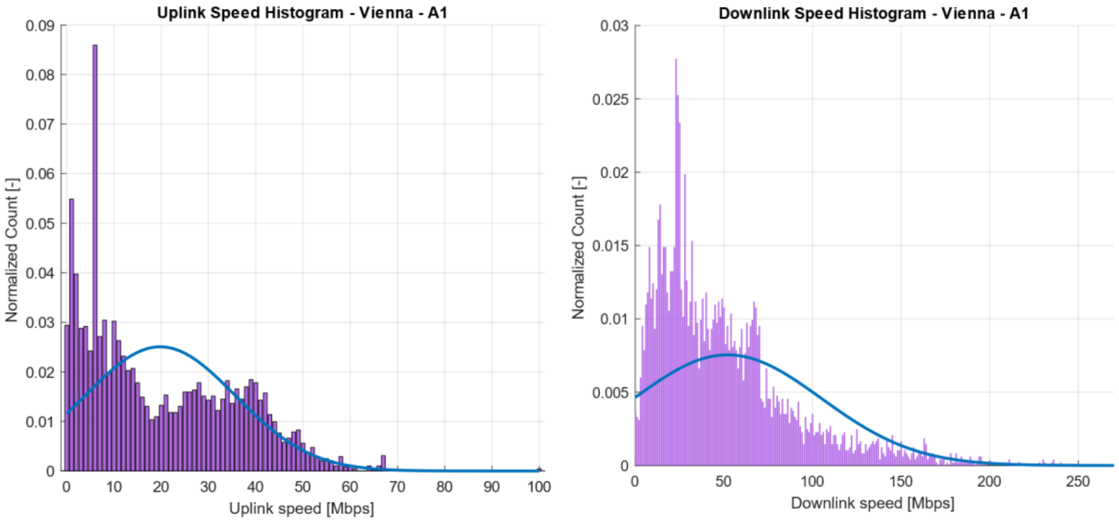


Figure 5-26 A1 - Uplink/Downlink Histogram, Vienna

The measurements of T-Mobile network are depicted in Figure 5-27. Also here, the distributions are not Gaussian.

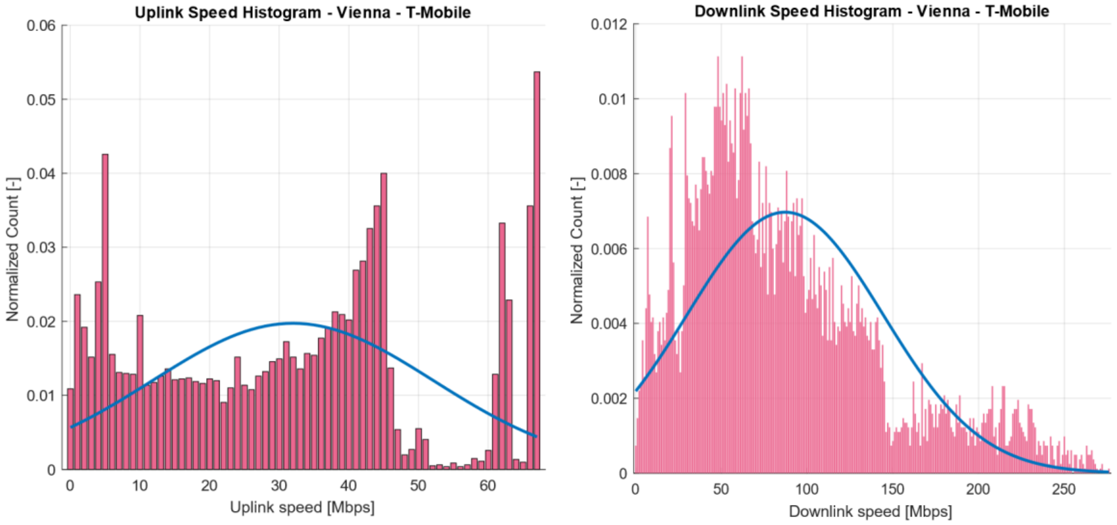


Figure 5-27 T-Mobile - Uplink/Downlink Histogram, Vienna

The distributions of RSRP for H3 and A1 are shown in Figure 5-28 left and right. In both figures, the fitted Gaussian distribution strongly resembles the empirical distributions of the measurement results.

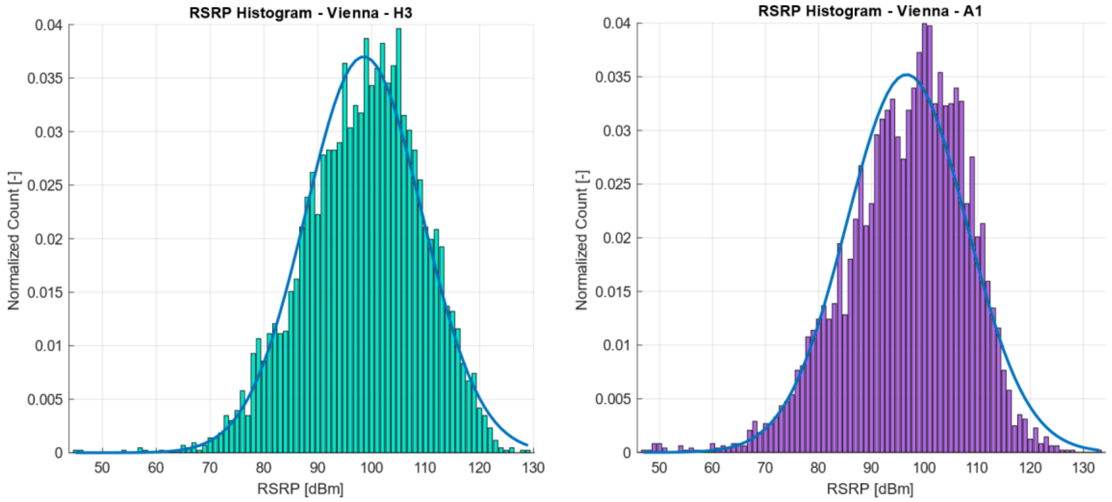


Figure 5-28 H3 and A1 - RSRP Histogram, Vienna

Figure 5-29 depicts the RSRP distribution of the RTR tests in Vienna realized in T-Mobile network. The resulting graph resembles Gaussian despite having an odd shape on the left slope of the distribution.

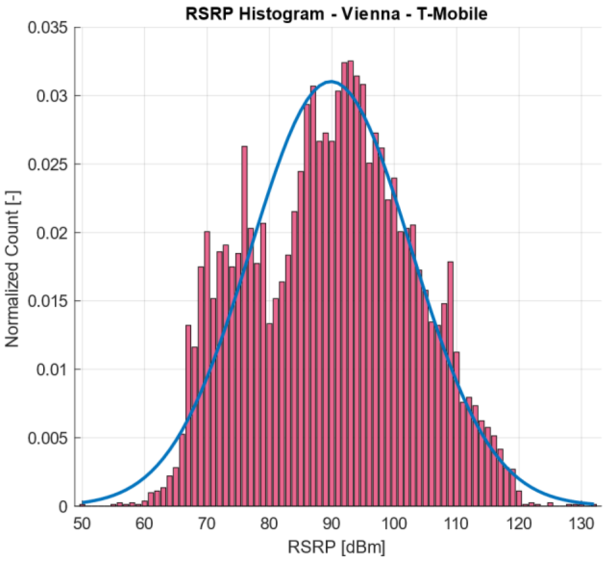


Figure 5-29 T-Mobile - RSRP Histogram, Vienna

The conclusion of this analysis is that in none of the networks, the upload and download distribution of the measurements is similar to the normal distribution. The RSRP distributions resemble Gaussian significantly. The non-Gaussian distributions of uplink and downlink may negatively affect the performance of GPR, which is designed to process the Gaussian-like data.

As the majority of the datapoints from the RTR database come from the users who performed only a signle measurement, the tariff limitation detection is impossible from the available data. As is clearly visible form the uplink and downlink distributions, all operators limit the maximum throughput within some tariffs, offering the unlimited connectivity in premium packages requiring a long-term subscription. The users with maximum speed limited from the operator's side then devaluate the results of the network performance estimation by reporting a false maximum throughput (limited by tariff).

As the non-Gaussian distributions of data reduce the potential performance of GPR, the tariff limitation factor devaluates the IDW's performance as well, as it considers the closest points on the grid with the highest weight. Two measurements next to each other, one with tariff limitation and one without it, create a strong fluctuation on the regression.

Below are presented the performance maps created isung IDW regression (power 3, infinity radius) for download and upload parameters. Figure 5-30 and Figure 5-31 depict the performance maps for downlink and uplink for A1, the performance maps of H3 are shown in Figure 5-32 for downlink and Figure 5-33 for uplink and Figure 5-34 with Figure 5-35 show the network performance prediction map for T-Mobile. The performance of GPR returned comparable results to the IDW (in some cases slightly larger, in others a little lower MAE).

The measured mean abosolute errors for downlink are presented in Table 5 along with mean measured value and relative MAE, which is calculated as MAE divided by the meam of all input samples. The table shows, that the relative error is up to the half of the mean value of the input samples. Compared to the RSRP values, for which the MAE for H3 is estimated as 7 dB and mean value -98.21 dB resulting in the relative MAE 7.13 %.

Table 5 Downlink MAE comparison

| **DL MAE** | MAE [Mbps] | Mean [Mbps] | Relative MAE [%] |
|---|---|---|---|
| A1 | 26.06 | 52.11 | 50.00 |
| H3 | 17.61 | 36.20 | 48.65 |
| T-Mobile | 18.92 | 87.14 | 21.71 |

Table 6 presents the error metrics for the uplink performance evaluation. The same as in downlink results, the relative MAE exceeds the bounds for which the regression presents the satisfying results.

Table 6 Uplink MAE comparison

| **UL MAE** | MAE [Mbps] | Mean [Mbps] | Relative MAE [%] |
|---|---|---|---|
| A1 | 6.70 | 19.78 | 33.87 |
| H3 | 9.23 | 13.54 | 68.17 |
| T-Mobile | 7.92 | 31.97 | 24.77 |

The reason why the regression, which works well with RSRP does not perform reliably for uplink and downlink is that those parameters are not reflected by elementary metrics such as signal strength. The downlink and uplink speeds depend on a number of additional factors apart from the received signal power such as the current load on the node, tariff limitation, current coding and modulation setting, potential handovers or performance utilization on the side of the end device (video streaming, online games etc.).

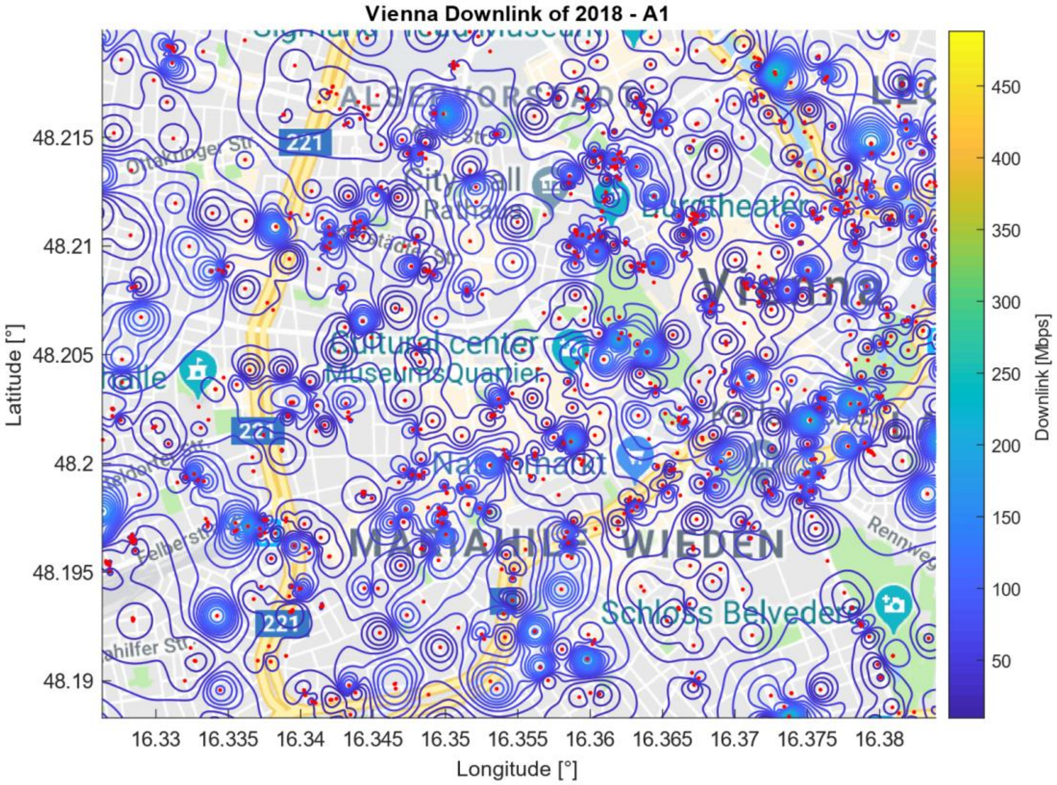


Figure 5-30 A1 - Downlink Map of 2018

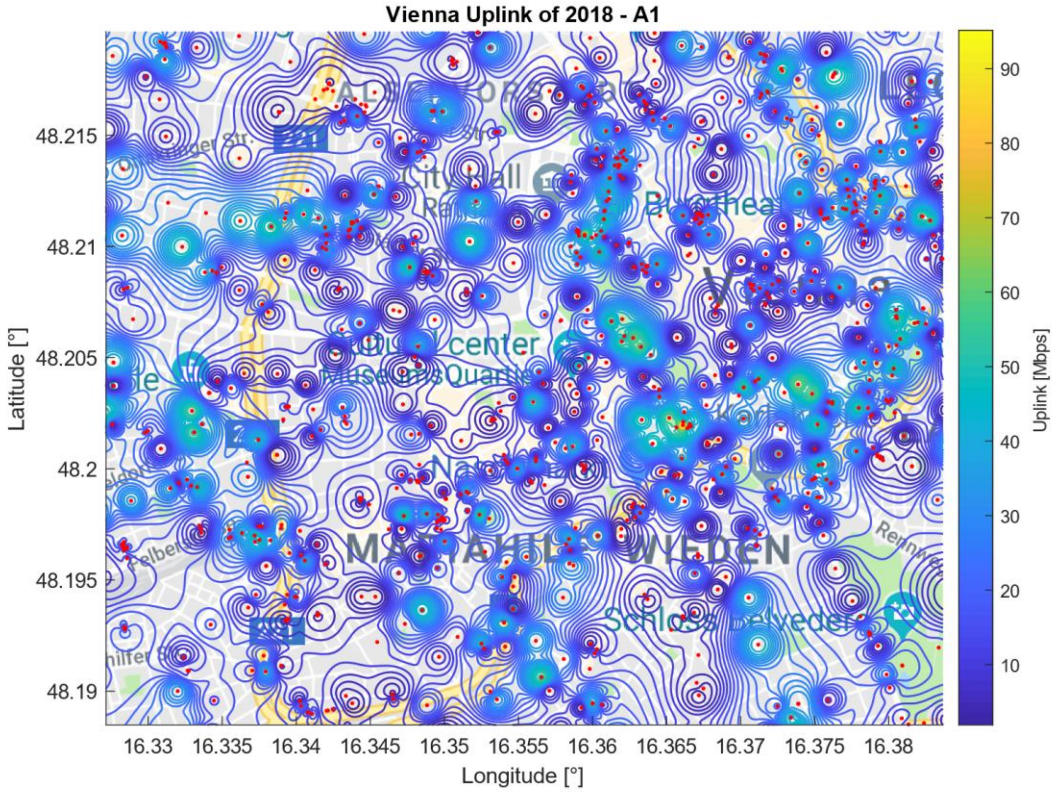


Figure 5-31 A1 - Uplink Map of 2018

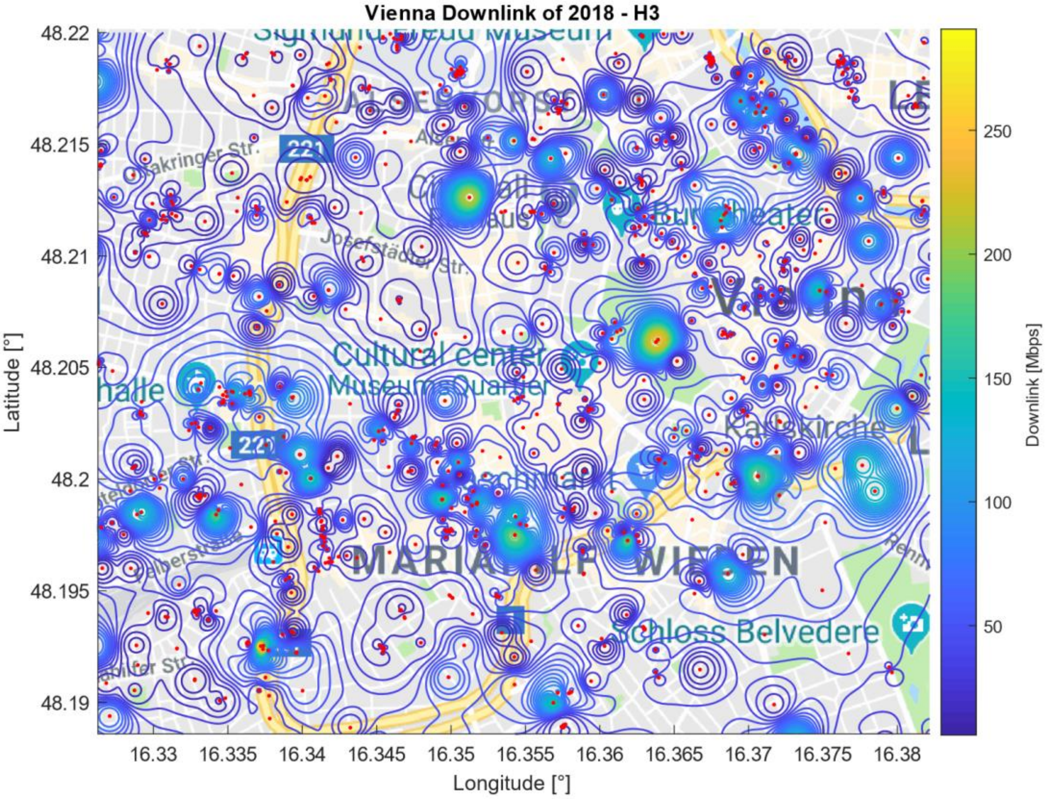


Figure 5-32 H3 - Downlink Map of 2018

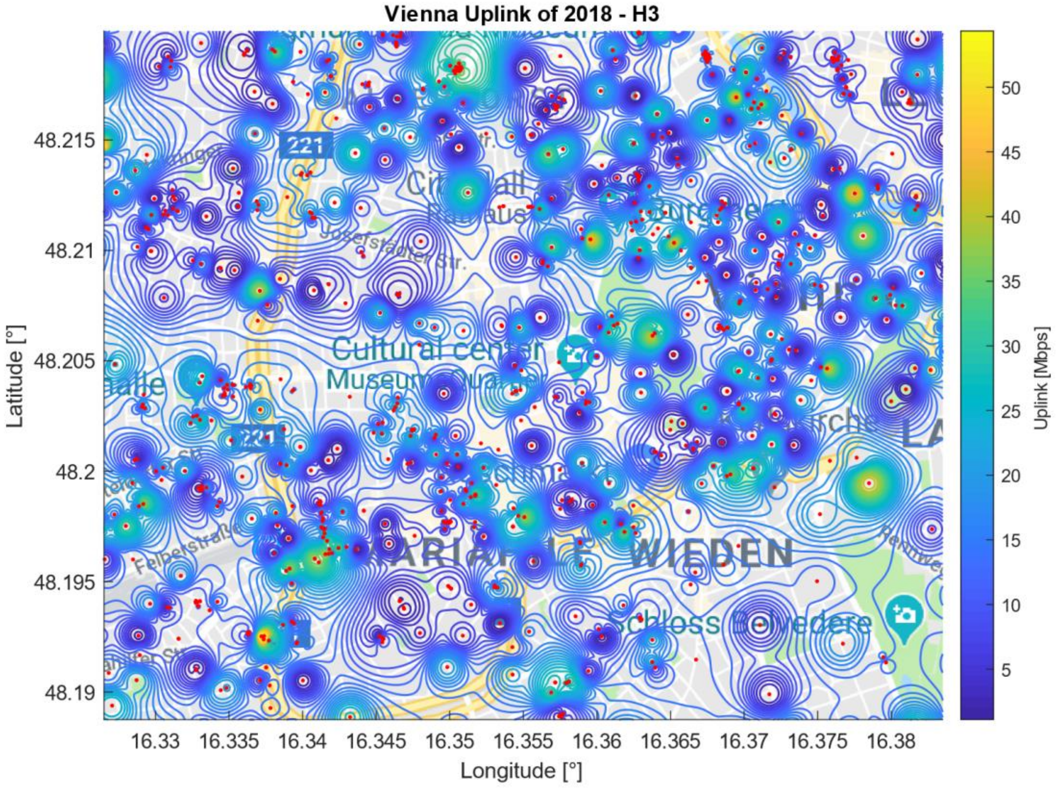


Figure 5-33 H3 - Uplink Map of 2018

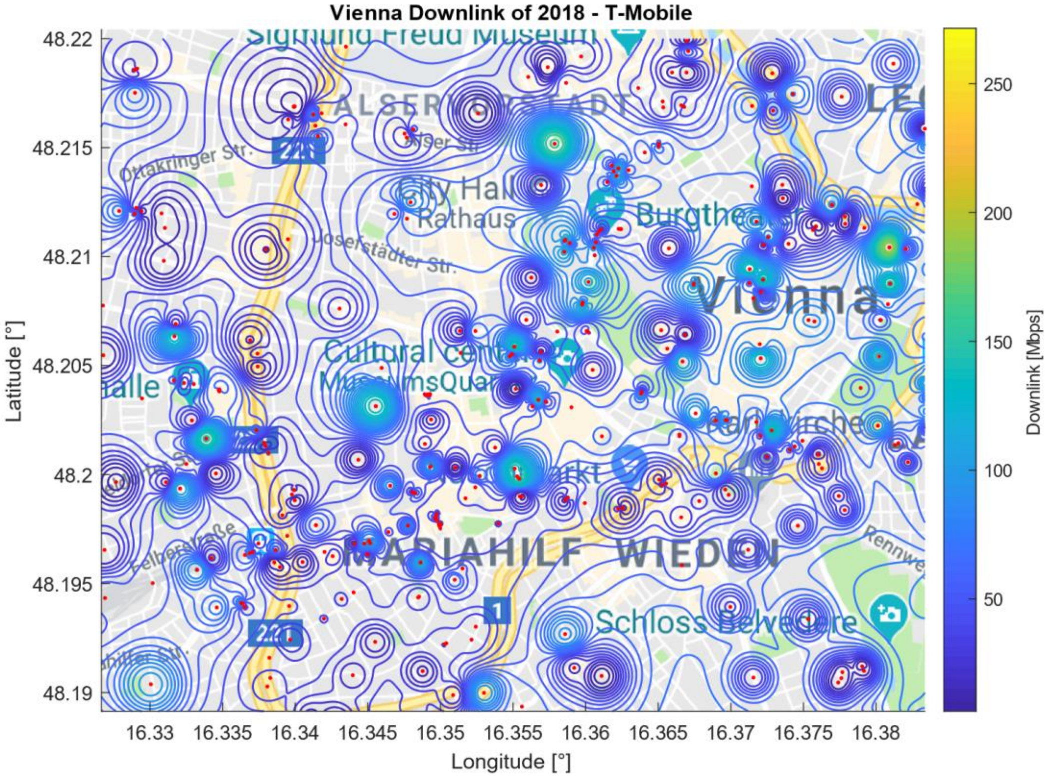


Figure 5-34 T-Mobile - Downlink Map of 2018

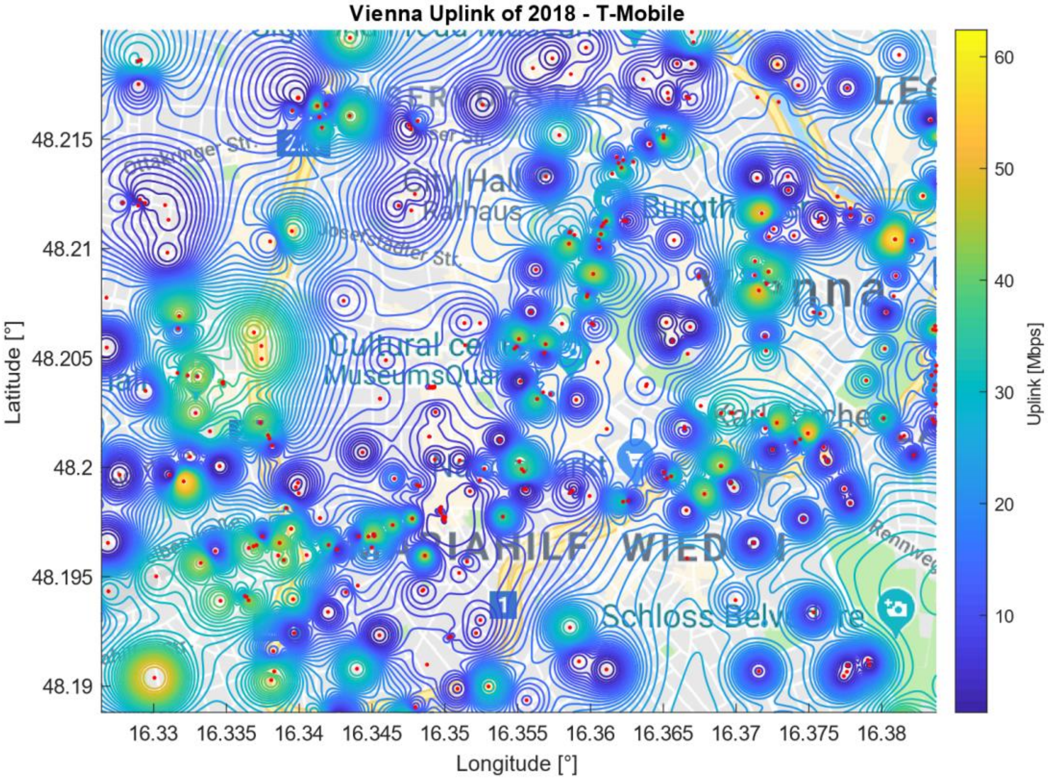


Figure 5-35 T-Mobile - Uplink Map of 2018

# FUTURE POTENTIAL & EXTENSION

Several possible extensions of this work may be utilized to obtain a better precision and reliability of the resulting regressions, leading to a better scalability and usability in performance optimization, resources allocation and eNodeB orientation or relocation.

Utilizing an additional source of data, which are merged with the current database to increase a number of data samples may greatly increase the operability of the created method. Utilizing for example a database from Tutela, who implements their measurements within other applications and games to periodically run lightweight tests, would quickly fill the map with relevant measurements.

To increase the reliability of downlink and uplink performance evaluation by identifying the tariff-limited users by utilizing a suitable technique (clustering, classification) may remove the inconsistencies they are causing from the performance maps and improve the method's performance. Tariff limited user could be not considered in performance evaluation or several parallel maps could be created, each for the specific group of users.

To perform the estimation of greater area (e.g. whole Austria) an area-dividing algorithm such as location-classifying algorithm or centroid-based clustering could be implemented. The performance mapping would then be realized based on the location, with the technique chosen based on the density of measurements.

The current method can be adjusted to localize the telecommunication nodes within the area, find the objects that dampen the signal and other facts about the current state of the network. By localizing the blind spots, the network operator can adjust the position or orientation of the current nodes or add the new ones to ensure better coverage of his network.

# SUMMARY

The performance optimization is an important topic in wireless networks, especially of interest for operators, regulators of the network and researchers. The main goal of this thesis is creation of a network performance parameters estimation approach, which creates performance maps for the chosen network parameters.  Based on the analysis of various regressions, the thesis proposes a set of optimal settings based on which maps may be created using measured points with known parameter value and coordinates.

First, the thesis introduces the reader to the topic of machine learning and big data, including “V” characteristics, storage options and brief history, followed by initial remarks on the thesis workflow. The challenges of working with crowdsource based data are listed and discussed. The considered data processing and regression methods are then introduced, and their basic parameters explained. As the method performance evaluation metric, MAE was introduced.

The network testing application RTR NetTest is introduced and discussed in detail, including its properties, interface and network performance testing process. The results and their visualization inside the application as well as in the web view is shown. The results of all tests, available as open data are introduced. The second source of data, the drive test measurements performed by NEMO Keysight system are introduced along with the supporting measurement tool.

In 1D analysis chapter, the NEMO measurement data are utilized to evaluate the performance of the considered regression methods and their applicability on the crowdsource-based data, as well as on the two-dimensional regression. Inverse distance weighting and Gaussian process regression were evaluated to be the best fitting methods for further application.

The reference measurements using H3 SIM card with unlimited data volume, downlink limitation to 50 Mbps and uplink limitation to 20 Mbps were performed in the city area and open area to evaluate the impact of the surroundings on the data. Along with the reference measurements, NEMO data were utilized in 2D analysis to find optimal parameters for each method. Adjusted Haversine formula was applied to transform the data coordinates into distances in meters to create regressions without the bias (as 1 degree of latitude and 1 degree of longitude do not represent the same distance). Minimum number of evenly distributed samples per area was derived, above which both regression methods show the same qualitative performance regarding MAE. The evaluation of methods established, that the optimal parameters for IDW are power parameter in the range from 3 to 4 and infinity radius, GPR performs the best with constant basis and rational quadratic covariance function. By evaluating the input data from all sources as discrete probability density functions it was discovered, that RSRP measurements resemble Gaussian distribution, whereas uplink and downlink have highly differing distributions, within which the tariff limitations are clearly visible.

The performance maps were created for the three biggest mobile network operators, predicting downlink, uplink and RSRP parameters. The relative MAE established, RSRP returns reliable performance maps, whereas uplink and downlink prediction is unreliable due to the corruption of data by tariff limitation and plenty other factors that are listed in the text. It was also established, that the current method while utilizing the existing

database is able to assess the parameters on 1.68 % area of Austria.

As the created RSRP prediction tool performs on the lowest margin of error with 30 measurements distributed evenly on the 1 square kilometre area, the performance maps can be created for the centres of the major cities in Austria. A set of proposals for the future extension of the method include extending the database by other sources of information or removing the tariff limitation bias by monitoring the limited number of users.

# REFERENCES

[1] “LOURIDAS, Panos a Christof EBERT. Machine Learning. IEEE Software[online]. 2016, 33(5), 110-115 [cit. 2018-12-10]. DOI: 10.1109/MS.2016.114. ISSN 0740-7459. Available at: http://ieeexplore.ieee.org/document/7548905/”.

[2] “MARR, Bernard. A Short History of Machine Learning -- Every Manager Should Read. Forbes. 2016”.

[3] “From the archive, 12 May 1997: Deep Blue win a giant step for computerkind. The Guardian. 2011”.

[4] “SAVOV, Vlad. The OpenAI Dota 2 bots just defeated a team of former pros. The Verge. 2018”.

[5] “GARETH, M. JAMES. Variance and bias for general loss functions. Kluwer Academic Publishers, 2003. DOI: https://doi.org/10.1023/A:1022899518027. ISSN 0885-6125.”.

[6] “M. Chi, Z. Sun, Y. Qin, J. Shen and J. A. Benediktsson, "A Novel Methodology to Label Urban Remote Sensing Images Based on Location-Based Social Media Photos," in Proceedings of the IEEE, vol. 105, no. 10, pp. 1926-1936, Oct. 2017,” doi: 10.1109/JPROC.2017.2730585.

[7] “Yuchi Tian, Kexin Pei, Suman Jana, and Baishakhi Ray. 2018. DeepTest: Automated Testing of Deep-Neural-Network-driven Autonomous Cars. In ICSE ’18: ICSE ’18: 40th International Conference on Software Engineering,” May 27-June 3, 2018, Gothenburg, Sweden. ACM, New York, NY, USA, 12 pages. Available at: https://doi.org/10.1145/3180155.3180220.

[8] “Leung, Kingsly and Christopher Leckie. “Unsupervised Anomaly Detection in Network Intrusion Detection Using Clusters.” ACSC (2005)”.

[9] “HILBERT, Martin a Priscila LÓPEZ. The World’s Technological Capacity to Store, Communicate, and Compute Information. American Association for the Advancement of Science, 2011. DOI: https://doi.org/10.1126/science.1200970. ISSN 0036-8075.”.

[10] “Big Data facts – How much data is out there? NodeGraph [online]. Available at: https://www.nodegraph.se/big-data-facts/?fbclid=IwAR14IwXTsCk _sXkgU7-29nGrYVav4XfCICjl8XvzWgJ_yvmEEci9x6yxa28”.

[11] “Dean, Jeffrey & Ghemawat, Sanjay. (2004). MapReduce: Simplified Data Processing on Large Clusters. Communications of the ACM. 51. 137-150. 10.1145/1327452.1327492.”.

[12] “LANDSET, Sara, Taghi M. KHOSHGOFTAAR, Aaron N. RICHTER a Tawfiq HASANIN. A survey of open source tools for machine learning with big data in the Hadoop ecosystem. Journal of Big Data [online]. 2015, 2,” DOI: 10.1186/s40537-015-0032-1. ISSN 2196-1115. Available at: http://www.journalofbigdata.com/content/2/1/24.

[13] “GANI, Abdullah, Aisha SIDDIQA, Shahaboddin SHAMSHIRBAND a Fariza HANUM. A survey on indexing techniques for big data: taxonomy and performance evaluation. Knowledge and Information Systems [online]. 2016,” DOI: 10.1007/s10115-015-0830-y. ISSN 0219-1377. Available at: http://link.springer.com/10.1007/s10115-015-0830-y.

[14] “HADI, Mohammed S., Ahmed Q. LAWEY, Taisir E.H. EL-GORASHI a Jaafar M.H. ELMIRGHANI. Big data analytics for wireless and wired network design: A survey. Computer Networks [online]. 2018,” DOI: 10.1016/j.comnet.2018.01.016. ISSN 13891286. Available at: https://linkinghub.elsevier.com/retrieve/pii/S1389128618300239.

[15] “STEPHENSON, Debbie. 7 Big Data Techniques That Create Business Value. THE DEAL ROOM. 2016”.

[16] “Singh, D. & Reddy, C.K. Journal of Big Data (2015) 2: 8. https://doi.org/10.1186/s40537-014-0008-6”.

[17] “Sayed Ali Ahmed, Elmustafa & Saeed, Rashid. (2014). A Survey of Big Data Cloud Computing Security. International Journal of Computer Science and Software Engineering. 3. 8”.

[18] “RTR-NetTest. RTR-NetTest [online]. Available at: https://www.netztest.at/en/”.

[19] “Traffic Summary. Vienna Internet Exchange [online]. Available at: https://www.vix.at/vix_statistics.html?L= 1&fbclid=IwAR37rEMANZgLs BFkuW8ff0Zr-tGD1JwFe8nzRuysf62R k0sZ5Jq18x5DtJw”.

[20] “TUTELA [online], Available at: https://www.tutela.com/”.

[21] “Pownuk, Andrzej and Kreinovich, Vladik,"Why Linear Interpolation?" (2017). Departmental Technical Reports (CS). 1098. http://digitalcommons.utep.edu/cs_techrep/1098”.

[22] “Interpolation. Scratchpixel 2.0 [online]. Available at: https://www.scratchapixel.com/lessons/mathematics-physics-for-computer-graphics/interpolation/bilinear-filtering”.

[23] “Linear Interpolation. Technopedia [online]. Available at: https://www.techopedia.com/definition/20366/linear-interpolation”.

[24] “HYNDMAN, Rob a George ATHANASOPOULOS. Forecasting: Principles and Practice: Chapter 7 Exponential smoothing. Otexts [online]. Available at: https://otexts.com/fpp2/expsmooth.html”.

[25] “George Y. Lu, David W. Wong, An adaptive inverse-distance weighting spatial interpolation technique, Computers & Geosciences, Volume 34, Issue 9, 2008, Pages 1044-1055, ISSN 0098-3004,” Available at: https://doi.org/10.1016/j.cageo.2007.07.010.

[26] “How IDW works. ArcMap [online]. Available at: http://desktop.arcgis.com/en/arcmap/10.3/tools/3d-analyst-toolbox/how-idw-works.htm”.

[27] “Chen, FW. & Liu, CW. Paddy Water Environ (2012) 10: 209., Available at: https://doi.org/10.1007/s10333-012-0319-1”.

[28] “M. Zhou, H. Guan, C. Li, G. Teng and L. Ma, "An improved IDW method for linear array 3D imaging sensor," 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Fort Worth, TX, 2017, pp. 3397-3400.,” doi: 10.1109/IGARSS.2017.8127727, Available at: https://ieeexplore.ieee.org/document/8127727.

[29] “Liaw, Andy & Wiener, Matthew. (2001). Classification and Regression by RandomForest. Forest. 23”.

[30] “HOW THE RANDOM FOREST ALGORITHM WORKS IN MACHINE LEARNING.

Dataaspirant [online]. May 22, 2017," available at: http://dataaspirant.com/2017/05/22/ random-forest-algorithm-machine-learing/ ?fbclid=IwAR3nJOQdp_Iey6RD _lrhrHYGB9voxy6j2pabO5dXHgcm6WFxco0FuPeN4bg.

[31] "DONGES, Niklas. The Random Forest Algorithm. Towards Data Science. 2018".

[32] "C. E. Rasmussen & C. K. I. Williams, Gaussian Processes for Machine Learning, the MIT Press, 2006, ISBN 026218253X. c 2006 Massachusetts Institute of Technology. www.GaussianProcess.org/gpml".

[33] "DONGBING GU a HUOSHENG HU. Spatial Gaussian Process Regression With Mobile Sensor Networks. IEEE Transactions on Neural Networks and Learning Systems [online]. 2012, 23(8), 1279-1290," DOI: 10.1109/TNNLS.2012.2200694. ISSN 2162-237X. Available at: http://ieeexplore.ieee.org/document/6218781/.

[34] "SONI, Devin. What is Bayes Rule?. In: Towards Data Science [online]. May 10, 2018, Available at: https://towardsdatascience.com/what-is-bayes-rule-bb6598d8a2fd".

[35] "Quiñonero-Candela, Joaquin, and Carl Edward Rasmussen. "A unifying view of sparse approximate Gaussian process regression." Journal of Machine Learning Research 6.Dec (2005): 1939-1959".

[36] "BROOKS-BARTLET, Jonny. Probability concepts explained: Bayesian inference for parameter estimation. Towards Data Science. 2018".

[37] "Fitgpr: Fit a Gaussian process regression (GPR) model. In: MathWorks: Documentation [online]. Available at: https://uk.mathworks.com/help/stats/fitrgp.html".

[38] "Median. Wolfram MathWorld [online]. Available at: http://mathworld.wolfram.com/Median.html".

[39] "Percentiles, Percentile Rank & Percentile Range: Definition & Examples. Statistics How To [online].," Available at: https://www.statisticshowto.datasciencecentral.com/ probability-and-statistics/percentiles-rank-range/.

[40] "Netzabdeckung. Drei AT [online]. Available at: https://www.drei.at/de/info/netzabdeckung/ ?fbclid=IwAR3z8fqAKN0yARIKszIl4o8nhX AZYZLlN5lJUnk84YZdPqLMaAjXIw2NVW8".

[41] "LTE CHECK. T-Mobile AT [online]. Available at: https://www.t-mobile.at/netz/ ?fbclid=IwAR0-XSfg4c_LMkPYytqLwpMqPrXT 40XwBwswUdkRSIOf8egeau8NmwkgCTs#tab5".

[42] "Mobile Netzabdeckung. A1 AT [online]. Available at: http://www.a1.net/hilfe-support/netzabdeckung/frontend/main.html?fbclid=IwAR2Q72Pva1LhkOAZt-d2gqRgkwSO-02GCUniGzTKcDzhVqmWUzF04WmhvZ4".

[43] "Upload/Download. Collinsdictionary [online]. Available at: https://www.collinsdictionary.com/dictionary/english/download".

[44] "Latency. Whatis.com [online]. Available at: https://whatis.techtarget.com/definition/latency".

[45] "MAMMEN, Stephen. Making Sense of Signal Strength/Signal Quality Readings for Cellular Modems. Industrial Networking Solutions Tips and Tricks. 2018".

[46] "FARISS, Todd. What is a Good Cell Phone Signal Strength? WILSON PRO. 2017".

[47] "Alladin [online]. alladin-IT, 2018 Available at: https://alladin.at/".

[48] "OPEN NETTEST [online]. Martes Specure International, 2018. Available at: https://www.martes-specure.com/".

[49] "Akos Net Test [online]. Slovenia: Agency for communication networks and services of the Republic of Slovenia, 2018. Available at: https://www.akostest.net/en/".

[50] "RATEL Nettest [online]. Serbia: RATEL, 2018. Available at: https://www.nettest.ratel.rs/en/index".

[51] "Checkmynet [online]. Luxemburg: checkmynet LU, 2018 Available at: https://checkmynet.lu/".

[52] "Nemo Walker Air In-Building Measurement Solution: KEYSIGHT Technologies [online]. [ref. 2019-03-10]. Available at: https://www.keysight.com/en/ pd-2767482-pn-NTD00000A/ nemo-walker-air?cc=AT&lc=ger".

[53] "Keysight Technologies," 5 July 2018. [Online]. Available: https://literature.cdn.keysight.com/litweb/pdf/5992-2051EN.pdf?id=2827649.

[54] "Distance on a sphere: The Haversine Formula. In: GeoNet: The Esri Community [online]. Oct 5, 2017. Available at: https://community.esri.com/ groups/coordinate-reference-systems/blog/2017/10/05/ haversine-formula".

[55] "Distance on an ellipsoid: Vincenty's Formulae. In: GeoNet: The Esri Community [online]. Oct 10, 2017. Available at: https://community.esri.com/groups/ coordinate-reference-systems/blog/ 2017/10/11/ vincenty-formula".

[56] "V. Lyandres, S. Briskin, On an approach to moving-average filtering, Signal Processing, Volume 34, Issue 2, 1993, Pages 163-178, ISSN 0165-1684, Available at: https://doi.org/10.1016/0165-1684(93)90160-C".

[57] "BROWNLEE, Jason. What is the Difference Between a Parameter and a Hyperparameter?. Machine Learning Mastery: Machine Learning Process [online]. July 26, 2017," Available at: https://machinelearningmastery.com/ difference-between-a-parameter-and-a-hyperparameter/ ?fbclid=IwAR3fsAWQgDR8iY9902IHptTrOK5UBo97He8cie8VkbYebmUJIMvL_mCH08c.

[58] "BAR-YEHUDA, Zohar. Plot_google_map. Github.com. Available at: https://github.com/zoharby/plot_google_map".

[59] "SOUZA, Cesar. Kernel Functions for Machine Learning Applications [online]. Mach 17, 2010.".

[60] "DUVENAUD, David. The Kernel Cookbook: Advice on Covariance functions [online]. Available at: https://www.cs.toronto.edu/~duvenaud/cookbook/".

[61] "Anderson-Darling Test. Engineering Statistics Handbook [online]. Available at: https://www.itl.nist.gov/div898/handbook/eda/section3/eda35e.htm".

[62] "SERGEEV, Dmitriy. Open Machine Learning Course. Topic 9. Part 1. Time series analysis in Python. Medium. 2018".

[63] "SERGEEV, Dmitriy. Open Machine Learning Course. Topic 9. Part 1. Time series analysis in Python. Medium. 2018".