User Equipment Characterization using Machine Learning

XIN ZHOU
VANESSA FAKHOURY
MASTER´S THESIS
DEPARTMENT OF ELECTRICAL AND INFORMATION TECHNOLOGY
FACULTY OF ENGINEERING | LTH | LUND UNIVERSITY

LUND UNIVERSITY

DEPARTMENT OF ELECTRICAL AND INFORMATION TECHNOLOGY
FACULTY OF ENGINEERING | LTH

User Equipment Characterization using Machine Learning

by

Xin Zhou xi0138zh-s@student.lu.se
Vanessa Fakhoury va5173fa-s@student.lu.se

EITM02
Master's Thesis (30 credits ECTS)
June 2019
Supervisor: Harsh Tataria
Examiner: Fredrik Rusek

Abstract

With the ever increasing demand for higher data rates and reliability, efficient management of cellular networks remains a challenge. Among other technologies, fifth generation systems are expected to tackle this challenge using large electronically controllable antenna arrays operating in the time division duplex mode. In such systems, spatial beamforming is implemented with accurate channel estimates at the cellular base station (BS). As a consequence of the high channel dimensions, large amounts of data are being collected at the BS which can be further utilized in order to optimize resource allocation, and to observe trends to facilitate more efficient beamforming to user equipments (UEs). To this end, machine learning methods play a vital role to identify useful patterns in data. In this thesis, four machine learning models have been built to categorize whether an UE is moving at a given velocity, or remaining stationary. In particular, binary neural network, multiclass neural network, support vector machine and logistic regression techniques are implemented and analyzed. The uplink sounding reference signal (SRS) channel estimates values are used as input data to the machine learning techniques. The SRS data are generated from a lab simulator at Ericsson AB, Lund. A binary neural network is first built to classify if the UE is moving or remaining stationary. Furthermore, the multiclass neural network is extended to classify movement of the UE at different speeds of 30 km/h or 100 km/h. Further to this, a support vector machine and logistic regression are implemented to compare performance and computational complexity of such approaches, relative to a binary neural network. The obtained results show that the binary neural network has the highest classification accuracy (98%) compared to the support vector machine (95%) and logistic regression (93.8%). For binary classification in this thesis, a larger amount of samples are input into the neural network therefore achieving the highest accuracy. Additionally, the multiclass neural network showed an accuracy of 89.2%. The accuracy of the machine learning algorithms depend on the problem scenario and the size of the dataset.

Acknowledgements

We would like to express our gratitude to our supervisors Jianmin Hou and Per Fryking for giving us the opportunity to do this master thesis at Ericsson AB in Lund, providing us with valuable suggestions and advice during the entire work. A big thank you to our university supervisor Harsh Tataria for his productive feedback and guidance.

Finally we would like to thank our family and friends for all the support during our study in Sweden at Lund University. We couldn't have done it without you. This master thesis concludes our education in the international programme of Wireless Communications at Lund University. These two years have been a truly great journey.

Xin Zhou & Vanessa Fakhoury

Contents

List of Figures

List of Tables

Preface

This thesis work was done at Ericsson AB in Lund, Sweden for approximately 5 months. In this project, both authors have completed a Stanford Machine Learning online course including the assigned programming exercises. Both authors engaged in all of the thesis work. Some techniques were researched separately and tasks were split, but problems and theories were discussed together throughout the entire work. Finally, the project work was filed as a patent invention with Ericsson AB, Lund.

List of Abbreviation

3D	Three Dimensional
4G	4th Generation
5G	5th Generation
AWGN	Additive White Gaussian Noise
BS	Base Station
CSI	Channel State Information
CNN	Convolutional Neural Network
eNB	evolved-NodeB
FDD	Frequency Division Duplex
FFT	Fast Fourier Transform
GUI	Graphical User Interface
LTE	Long-Term Evolution
MIMO	Multiple Input Multiple Output
ML	Machine Learning
RBF	Radial Basis Function
RF	Radio
RSS	Received Signal Strength
SRS	Sounding Reference Signal
SVM	Support Vector Machine
TDD	Time Division Duplex
UE	User Equipment

1

Introduction

1.1 Background and motivation

Rapid developments in communication systems sustain a high demand for data rates, a large number of terminals, as well as massive number of antennas at the base station. In order to deal with these challenges, a range of evolution in the technology have been developed, which have led up to the 5th generation (5G) systems. Looking back at the 4th generation (4G) of cellular systems, long-term evolution (LTE) standardized base station evolved-NodeB (eNodeB)'s to have multiple antenna elements for signal transmission and reception. This facilitates service to multiple user equipments (UEs) by distributing the radio resources in time, frequency and power domains. Since 5G base stations are envisioned to scale up the number of antennas, they aim to exploit the spatial domain for leveraging the multiplexing gains of the system. Unlike previously, this allows for the system to simultaneously serve multiple UEs within the same time-frequency resource.

Beamforming, leveraging accurate knowledge of the propagation channel at eNodeB, has proven to be the most efficient way to exploit the spatial domain [4]. In LTE, the eNodeB process sounding reference signal (SRS) which are periodically (or aperiodically) sent by UEs to estimate the uplink channel state. The SRS specific to each UE changes with the channel condition, as the propagating waveform interacts with different objects in the environment, causing large and small-scale fluctuations on the waveform envelope across the movement trajectory. If UEs are relatively stationary, their SRS tends to stay fairly constant due to marginal deviations in the channel conditions. Due to many antennas are receiving the SRS from each UE simultaneously, the SRS dimensions increase. Critically, the updating of high dimensional SRS leads to an increase in computational complexity. This introduces the need for intelligent processing to categorize the UEs into stationary or moving in order to effectively optimize the resources for beamforming.

1.2 Purpose and Aims

Managing wireless networks that grow in complexity is a difficult task, due to the processing requirements for large amounts of data. Machine learning is a compelling tool for managing vast data, as it can extract valuable features from raw data to generate insightful predictions [1]. The aims of the thesis are to investigate whether machine learning methods can cope with the large amounts of high dimensional

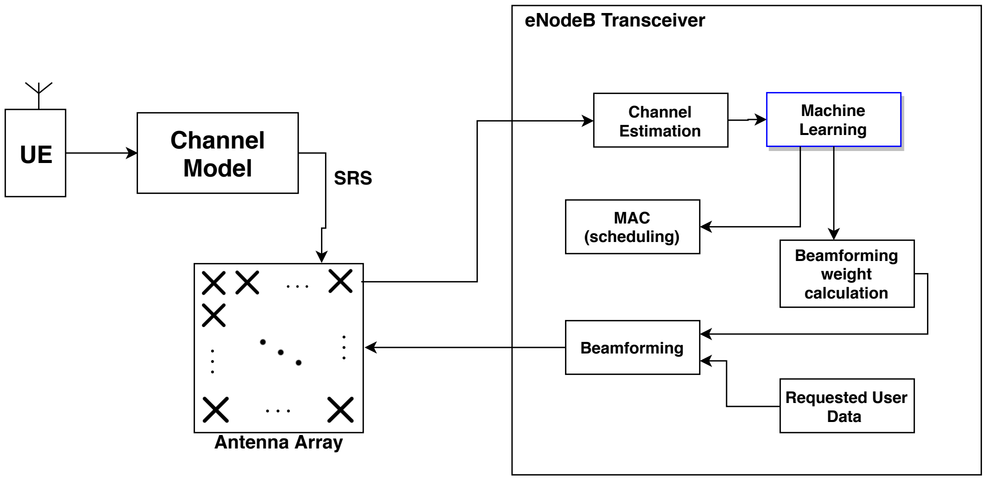

Figure 1.1: A mobile UE transmits SRS to the multi-antenna eNodeB, which will be trained by machine learning to adjust the beamforming weights

measurement data of SRS in an intelligent manner at the eNodeB. Furthermore, it is noteworthy to study whether algorithms can be trained by using SRS to determine if the UE is stationary or moving. In addition, whether different speeds of UE can be classified by machine learning algorithm. As such, knowing speed characterization of UE could result in a reduction in computations for beamforming.

A typical recent cellular eNodeB consists of several processing blocks as shown in Figure 1.1. The goal is to create an efficient method for the machine learning block highlighted in blue, which helps with the computation and efficient adaption of the beamforming weights to reduce the eNodeB processing complexity.

This project's scenario includes a mobile UE transmitting SRS through different channel conditions to the multiple input multiple output (MIMO) eNodeB transceiver. The eNodeB uses uplink SRS for channel-state estimation to support downlink channel-dependent scheduling and beamforming weight calculations which adjusts antenna beamforming to transmit the requested user data [7].

1.3 Methodology

To find a solution of the above mentioned problem scenario, the thesis work started with a literature study. Both authors completed an online course for machine learning offered by Andrew Ng in Stanford University, where several machine learning methods were introduced and programming exercises were provided. After investigation on what kind of data can be collected and which scenarios should be tested, channel estimation logs were collected from Ericsson Lund's local lab simulator. Data was extracted, filtered and converted accordingly and fed into neural network models in MATLAB. Additionally, more algorithms, such as support vector machine (SVM) and logistic regression were tested and investigated for comparison purposes.

1.4 Literature Review

In literature, researchers have tackled similar problems with different approaches, such as machine learning algorithms or wireless communication technologies.

The work in [8] introduced two algorithms, spectral analysis method and time-based spectrum spreading method, to estimate the speed of UEs in LTE system. Despite successful separation of UEs into speed categories through the established algorithms, this work does not implement machine learning method in any form, but instead using a database with reference curves.

A lot of researches have delivered substantial results with machine learning for UE positioning [2]. Channel charting, as a machine learning algorithm, was used in localization of UEs with a multi-antenna base station in [2]. Similarly, other machine learning approaches such as deep convolutional neural networks (CNNs) and the nearest neighbor regression were investigated in massive MIMO fingerprint-based positioning and mobile location MIMO communication systems [5][6].

Machine learning based beamforming in [9] proposes an algorithm which recommends the user to choose between two beamforming schemes, maximum ratio transmission and zero forcing, taking channel vectors and transmit power as the algorithm input. Radio frequency (RF) based three dimensional (3D) skeletons [10] uses the environment's RF signals and CNN to localize and track human bodies motions. Moreover, the project in [11] shows how deep learning estimates wireless channel and direction-of-arrival for massive MIMO system, where offline and online learning procedures are used to achieve better estimation performances.

With the aforementioned discussion in mind, previous researches have only covered speed of UEs with wireless communication technologies or the localization of UE using machine learning. While this project using machine learning methods to do movement classification of UE into moving or stationary and speed classification with channel estimation as input.

2

Technical Background

In this section, the concept of massive MIMO will be introduced. Furthermore, the technical knowledge and background of the wireless communication system will be given. Wireless channel properties, channel estimation and beamforming will be covered for better understanding of the thesis. Moreover, the machine learning algorithms used in this thesis will be explained.

2.1 Wireless Channels

The performance of a wireless communication system heavily depends on its channel conditions. Transmitted signals will likely be affected by physical circumstances in terms of amplitude and phase variations, therefore it is of great importance to take these changes into consideration. These influences on the wireless communication can be grouped into two types, small- and large-scale fading, and will be discussed further in the following sections.

2.1.1 Small-scale fading

Small-scale fading is also known as fast fading and due to mobility of the UE over time or movement of scatters when the UE is fixed, the channel characteristics and propagation between transmitter and receiver will vary. Multipath propagation is known as the receiving of radio signals by two or more different paths due to reflections and refractions [41]. These propagation mechanisms can either cause constructive or destructive interference, along with phase shifts and rapid changes in amplitude of the transmitting signal.

In a wireless communication environment, each signal path has its own time delay, attenuation and Doppler shift. Taking into consideration the Doppler shift is critical for accurate modelling of wireless channels when a UE is moving, due to the fact that the frequency error is proportional to the speed of the UE. Additionally, the coherence time, or time duration when the wireless channel variations are negligible, can be estimated from the Doppler shift frequency [42].

2.1.2 Large-scale fading

Large-scale fading, also known as shadowing, represents the impact on average signal-power attenuation, causing phase variations and a large path loss exponent due to movement over large areas [28]. Shadowing is the result of terrain structures obstructing the signal travelling from transmitter to a receiver, where obstacles include mountains, hills, or large buildings. It is therefore also often referred to as slow fading. Path loss is defined as the attenuation of a signal due to distance between transmitter and receiver [43], as well as height and location of antennas. A large path loss exponent can be an effect of refraction, diffraction, reflection and absorption and is influenced by environment, propagation medium and more [44].

2.2 Introduction of Massive MIMO

Massive MIMO, best described as large scale antenna array system, offers big advantages over conventional multiple antenna systems on both UEs and base stations. It uses a large number of electronically controlled antennas to serve a multiplicity of terminals within the same time-frequency bins. Massive MIMO extends conventional multiuser MIMO system by an order of magnitude, with a focus on improvements in throughput, spectral efficiency, and energy efficiency [13].

2.2.1 Concept and Properties

In an LTE system, uplink is defined as the communication link from the UEs to the eNodeB, while downlink is known as the communication link in the opposite direction. There are several key characteristics of massive MIMO systems, one of which is that the system process is a fully digital baseband chain, enabling coherent signal processing from all antennas at the base station as well as the ability to measure channel responses on both uplink and downlink to quickly respond to the changes in the channel [14].

Time division duplex (TDD) is an important feature utilized in the massive MIMO system, implying that transmissions take place in different time slots which do not overlap. Channel reciprocity is assumed between uplink and downlink during TDD, due to the fact that the channels are identical when channel conditions are constant.

Further advantages of massive MIMO include channel hardening, which is known as the scenario when the number of antennas in a massive MIMO base station are increased and the variations of channel gain in both frequency and time decrease [15]. Channel hardening removes the effects of fast fading which has been discussed in Section 2.1.1. The link between the UE and eNodeB becomes a scalar channel with a stabilized gain, which is set to a deterministic constant, independent of frequency [14].

Favorable propagation is another one of the key advantages of massive MIMO and is defined as the mutual orthogonality of the vector-valued channels between the base stations and terminals [45]. This massive MIMO radio channel property is particularly favorable when maximizing sum-capacity. It helps to achieve optimal performance through linear processing. During uplink, interference can be cancelled out by a linear detector and matched filter, and during downlink, the eNodeB can perform simultaneous beamforming to multiple UEs without mutual interferences [45].

2.2.2 Propagation Channel Modeling

In order to efficiently design and evaluate the performance of massive MIMO, it is necessary to develop an accurate and flexible channel model. The 3GPP standard covers three different multipath propagation channel models; Extended Pedestrian (EPA), vehicular (EVA) and Typical Urban Mode (ETU). Each model presents a low, medium and high doppler spread environment, for which the delay profiles can be found in [12]. When looking at static propagation scenarios, an Additive White Gaussian Noise (AWGN) environment is considered, where no fading or multipaths are present [12].

2.3 Channel Estimation

A communication channel refers to the transmission medium between a transmitter and a receiver through either a physical wire or a radio channel. In wireless communication systems, the signal is transmitted by radio waves, where the channel estimation describes the channel property when a transmitted signal propagates from the transmitter to the receiver or vice versa.

Channel estimation is crucial for reliable transmission and can be categorized into two methods. The data-aided channel estimation method is based on known data for both transmitter and receiver, such as a training sequence or pilot, whereas during the blind approach, the estimation is based on a received signal sequence, without knowing transmitted signal beforehand[51]. In the data-aided method, the reference signal is transmitted to the UE during downlink, where the it can then estimate the channel and feed back the Channel State Information (CSI) to the transmitter. CSI includes the knowledge of the channel and can range from a general statistical model of the set of channel properties, to specific channel frequency response estimations and their accuracy [49].

In order to perform channel estimation, resources are reserved for transmitting pilot sequences. In Frequency Division Duplex (FDD) transmission occurs simultaneously in time but in different frequency bands. This results in the requirement of channel estimations in both uplink and downlink, which is estimated at the receiver in FDD and sent back to the transmitter (eNodeB). However, in TDD systems, channel reciprocity is assumed and therefore the downlink channel quality can be obtained from its uplink equivalent during the coherence time [6]. This means that in order to minimize pilot overhead, the channel estimation is only estimated by transmitting SRS during uplink, which is the method this thesis focuses on.

The uplink channel knowledge is later useful for scheduling, or assigning communication resources to users which are not experiencing fading dips at that frequency and time, to increase spectral efficiency.Considering the scenario when a UE is moving, the received signal strength (RSS) varies, exposing a risk of data loss due to the fact that the base station cannot attempt communication when the UE is in a fading dip, if the transmission occurs at that frequency over a single pair of antennas. In massive MIMO systems for example, due to the large number of antennas, the channel estimation quality is improved as the cost for pilot overhead is proportional to the number of terminals [31][32].

2.4 Beamforming

With the increased network capacity and coverage demands, massive MIMO employs what is called beamforming. Highly directional signal beams are transmitted to particular users instead of being broadcasted through the entire cell, therefore reducing interference across cells and approaching a higher throughput and higher data-rate over a longer distance [33]. It is a signal processing technique, working at particular angles by combining elements in a phased array such that some users experience constructive, and others destructive interferences [50]. Individual data packets can be sent in many different directions with different arrival time in a precisely coordinated pattern to reach the intended users, enabling the antennas on a massive MIMO array to transceive a lot of information at once.

Massive MIMO systems use smart antenna arrays, or digitally controlled antennas, consisting of signal processing algorithms which keep track of spatial signal identifiers, such as direction of arrival, to compute the beamforming vectors for transmission [34]. These vectors identify the required signal sent from mobile stations and monitor it. Smart antennas are used in acoustic signal processing, track-and-scan radars, radio telescopes and more, but our focus stays on their use in the LTE wireless communication systems.

There are various different beamforming techniques for massive MIMO systems in LTE and LTE-Advanced using smart antennas such as adaptive beamforming, analogue or digital beamforming and switched beamforming. Switched beamforming relies on fixed beam networks, while adaptive networks adjust their beams according to the processed weight vectors [34]. In this project, scenarios using switched beamforming were simulated with the help of a Butler matrix, which is a matrix including hybrid couplers, phase shifters and crossovers. A more detailed explanation of the scenario using the beams of the Butler matrix will be discussed in Chapter 3.

2.5 Machine Learning

Machine Learning (ML) is the scientific study of algorithms and mathematical models that computers use to perform specific task efficiently without explicit instructions, relying on patterns and inference instead[52].

In this section, three different machine learning algorithms are described, namely, neural network, support vector machine and logistic regression.

2.5.1 Neural Network

A neural network is inspired by the biological neuron system and consists of several processing layers: an input layer, one or more hidden layers, and an output layer [37]. The layers are interconnected by neurons with outputs of each layer acting as inputs of the next layer, where the layers are connected by weights. The basic neural network model can be described as a series of functional transformations, that is, all the neurons in the previous layer apply their transformations to the information they receive and send it to the next layer. An example of a simple three layer neural network with D input variables, M hidden neurons and K output variables is shown in Figure 2.1. The hidden layer can be constructed by input variables x_1, $x_2 ... x_D$ in the form

$$a_j^{(2)} = g(\sum_{i=1}^{D} w_{ji}^{(1)} x_i + w_{j0}^{(1)}) \tag{2.1}$$

where j=$1, 2, \ldots, M$ and the superscript (1) indicates the parameters in the first layer and (2) representing the parameters in the second layer. The parameter $w_{ji}^{(1)}$ refer to the weights from input layer to hidden layer and the parameter $w_{j0}^{(1)}$ act as bias units which are often set to be 1. The function g(.) is called the activation function and is generally chosen to be a sigmoid function. The sigmoid function maps the resulting values between 0 and 1 measuring the probability of the categories as an output of the neural network. It is a special case of the logistic function as shown in Figure 2.4. The details of logistic function will be explained in the next section. Following equation 2.1, the output values of the hidden layer are again linearly combined as the input of an activation function in the output layer

$$h_k = g(\sum_{j=1}^{M} w_{kj}^{(2)} a_j^{(2)} + w_{k0}^{(2)}) \tag{2.2}$$

where $k = 1, 2, ..., K$ and again $w_{k0}^{(2)}$ are bias units [24].

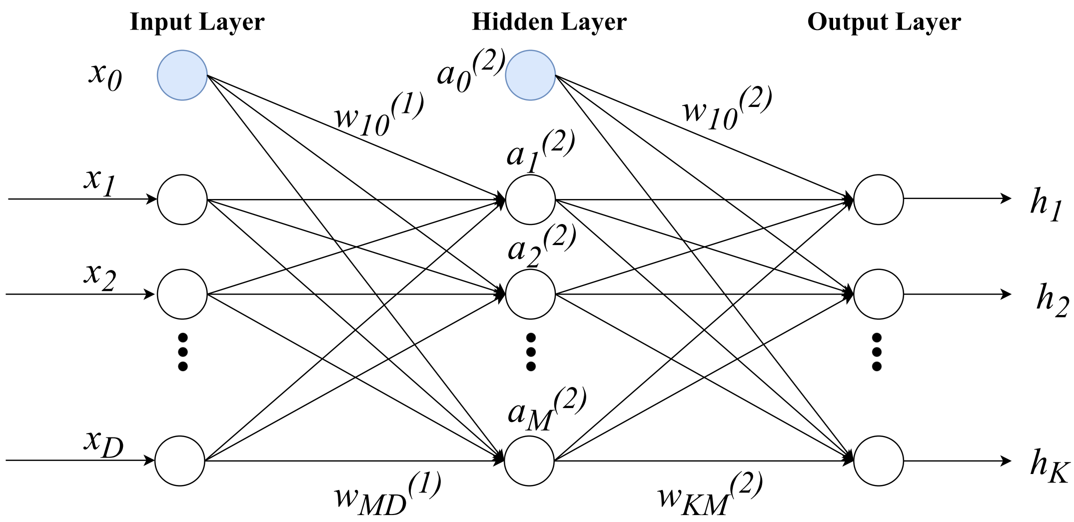

Figure 2.1: Diagram of a three-layer neural network with D input neurons, M neurons in hidden layer and K output neurons

The goal of the neural network is to optimize the vectors $w_{ij}^{(1)}$ and $w_{kj}^{(2)}$ such that the cost function J reaches its minimum. The cost function indicates how close is the predicted values are with the labelled values. The lowest possible version of the cost function is therefore desired.

2.5.2 Support Vector Machine

Support vector machines (SVMs) is a machine learning algorithm suitable for classification problems. An important property of SVM is that any local solution is also a global optimum [24]. It was originally designed for binary classification, however how to effectively extend it for multiclass classification is still an ongoing research issue, as it is computationally more expensive to solve multiclass problems using SVM

[18]. The main idea of the SVM is to construct a hyperplane in a high-dimensional feature surface which is mapped by input vectors so that the training data can be separated without errors [19], i.e. the hyperplane is k-1 dimensional in a k dimensional feature surface. In this thesis, the binary classification using SVM is mainly discussed.

Linear SVM

Figure 2.2 shows a two-class classification example using linear hyperplane. The red colored circles and blue solid filled triangles denote data points from two different classes. An optimal hyperplane is defined as the linear decision function with maximal margin in Euclidean space between the vectors of the two classes. The input vectors on the dotted lines in Figure 2.2 are called support vectors which define the margin of the hyperplane [19]. Once the model is trained, a significant number of samples can be discarded, and only the support vectors remain [24]. Any new data will be categorized into the appropriate sides of the hyperplane through usage of the model.

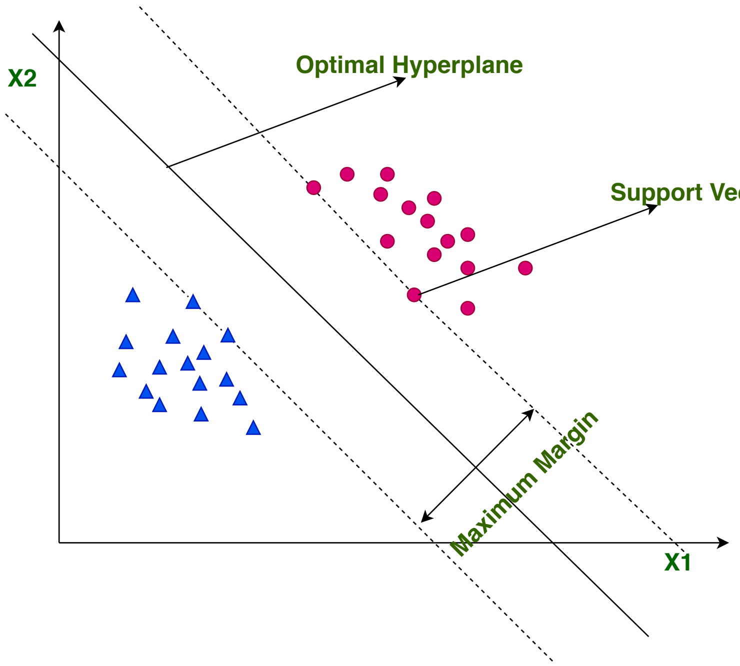

Figure 2.2: The hyperplane (black line) representation in SVM. The red colored circles and blue solid filled triangles denote data points from two different classes. The data points on the dotted lines represent support vectors.

Non-linear SVM

In the leftmost graph of Figure 2.3, non-linear separable data can be seen. With a linear hyperplane, it is not possible to separable the two-class datasets. In this case, kernel function can be used to map the data into high dimensional space so that a linear hyperplane can be found and nonlinear decision boundary can be learned.

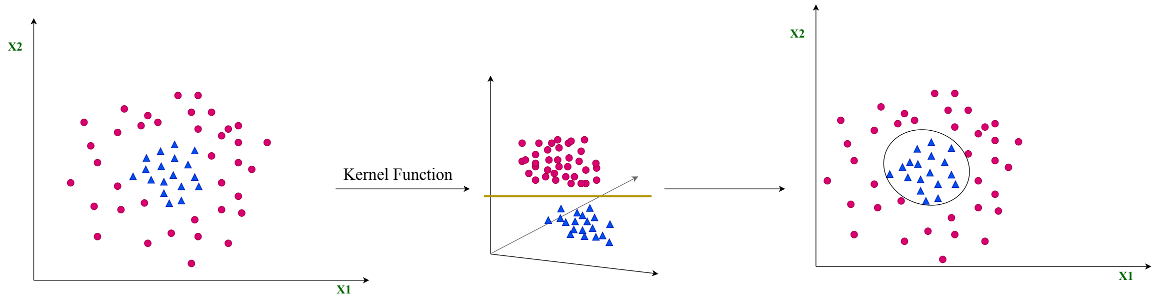

Figure 2.3: The process of data processing in order to find non-linear hyperplane by kernel function [25].

2.5.3 Logistic Regression

Logistic regression is a widely used machine learning algorithm which uses a logistic function to model a binary classification variable, although it can be extended to deal with multiclass classification problems. The logistic function is a "S" shape defined by the equation

$$f(x) = \frac{L}{1 + e^{-k(x-x_0)}} \tag{2.3}$$

where L is the maximum value of the curve, k is the curve's steepness and x_0 is the midpoint of the x-value [27]. The standard logistic function is the logistic function with parameter $L = 1$, $k = 1$, $x_0 = 0$ which is shown in Figure 2.4. The binary logistic function $f(x)$ has two possible results "1" or "0". The input x can be a variable or a linear combination of several variables with weights. Thus, determination of x is the key of the logistic regression algorithm.

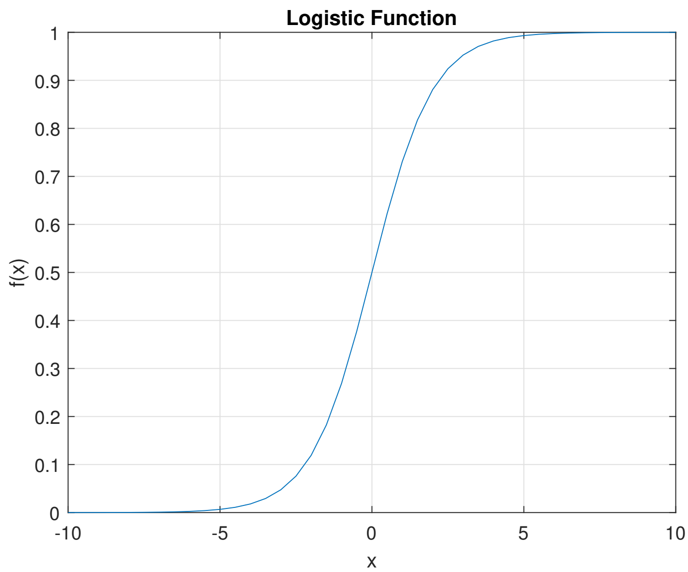

Figure 2.4: Logistic function with $L = 1$, $k = 1$, $x_0 = 0$

3

UE Classification with Machine Learning

This chapter presents the scenario of the considered wireless communication system. Moreover, the preparation of the collected data before the machine learning input is described. Additionally, the methodology of different machine learning algorithms is explained.

3.1 System and Data Description

A wireless communication system is simulated in the lab in Ericsson Lund. The communication system employed in this thesis can be described as follows: A single cellular base station eNodeB communicates with a single antenna UE. The channel estimation measures the uplink channel, based on SRS which is transmitted during uplink from UE to eNodeB.

The 32 x 16 Butler Matrix is a beamforming network implemented at the base station of the simulation lab as shown in Figure 3.1. The Butler Matrix is used to activate antenna beams in specific directions with magnitude and fixed phase in one plane [38]. Since the 32 x 16 Butler Matrix is an extension of a 4 x 4 Butler Matrix, the structure of the latter will be explained.

As shown in Figure 3.2, the Butler Matrix consists of four input ports labelled I1 to I4 and four output ports O1 to O4, which are connected to four antenna elements. The Butler Matrix feeds power to the antenna elements through 3dB hybrid couplers and 45 degree phase shifters such that the beams of the radio transmission are aimed in the desired directions [47]. Since the number of input ports is equal to the number of output ports, this Butler matrix is the analog version of the Fast Fourier Transform (FFT) [48]. Additionally, the number of antenna ports correspond to number of beam patterns and due to reciprocity, the transmitted beam pattern is considered to be the same as the as the received beam pattern. More details of the Butler Matrix can be found in [40].

The layout of the simulation including two eNodeB generated beams and one UE is shown in Figure 3.3. The beams are placed 2 kilometers apart, due to the fact that in this thesis, the speed of the UE, which is set manually, can be up to 100 kilometer per hour. The UE is either considered as stationary, or moving in a configurable elliptical route with a certain speed back and forth between two transmission beams.

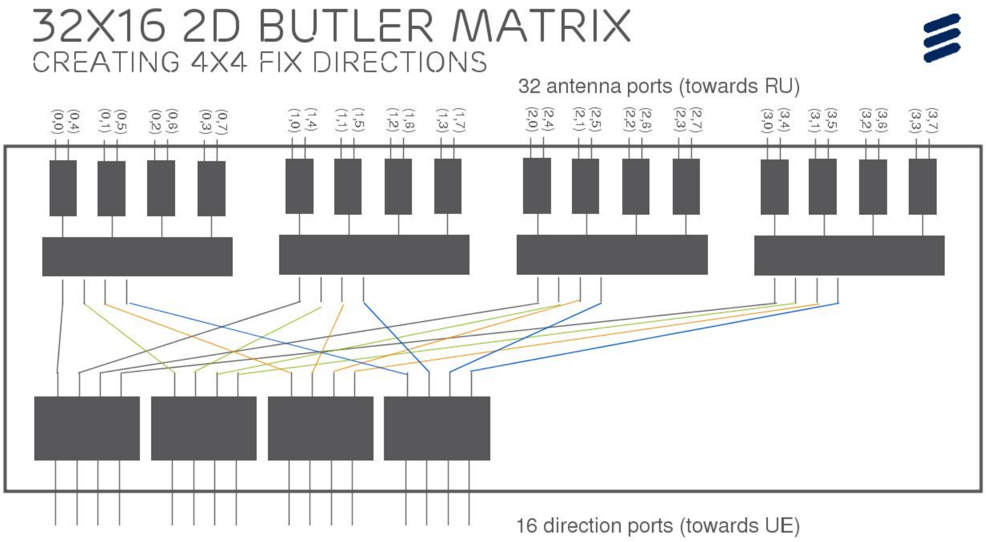

Figure 3.1: Block diagram of the 32 ×16 Butler Matrix

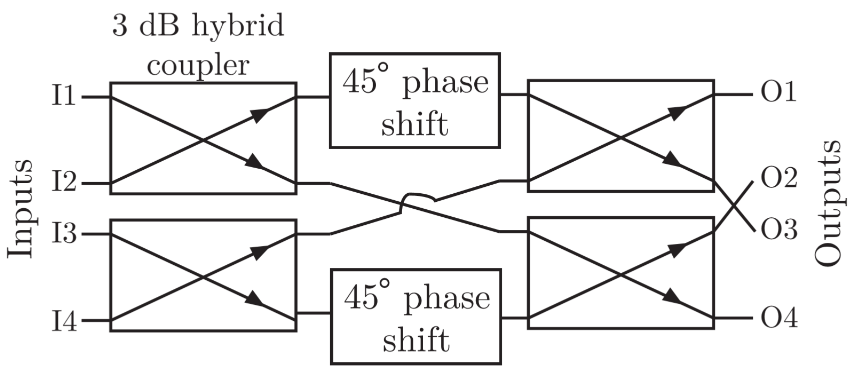

Figure 3.2: Block diagram of a 4 ×4 Butler Matrix

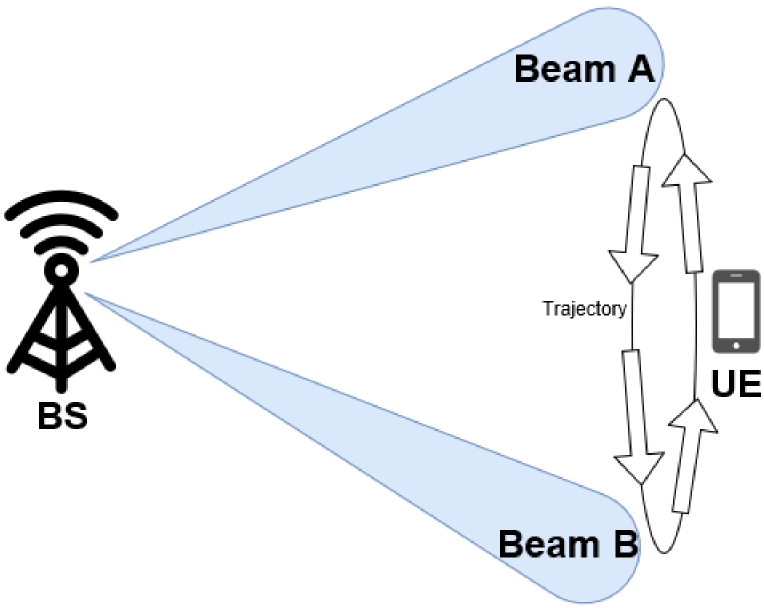

Figure 3.3: Scenario layout with two beams generated from base station and a moving UE

In the lab simulator, the base station measures the channel estimates in 64 amplitude and phase observations every 20 milliseconds, which is a recurrence set by the 3GPP standard [12]. The observation with maximum amplitude indicates the strongest reference signal strength from the UE. Due to the power control implemented in the lab, normalization is performed by dividing the sum of the amplitudes of all observations to compare their signal strength. An example of the amplitudes of the channel estimates in each observation is shown in Figure 3.4, where the observation 33 has the highest normalized absolute voltage. For each time instant, maximum amplitude of the channel estimation is selected as one sample, which will be explained in more detail later in the text.

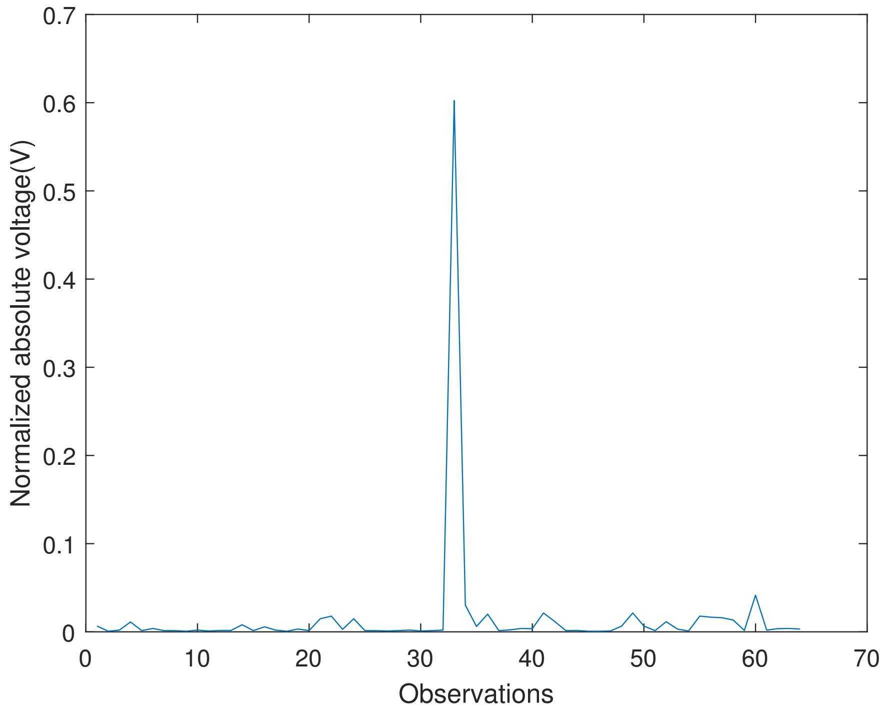

Figure 3.4: The channel estimation strength in 64 observations

Since the channel estimation is performed frequently, the values will change faster with different time instant if the UE is moving with high speed; When the UE is stationary or at a low speed, the channel estimates will change slowly. Figure 3.5 and 3.6 show an example of how the samples change at different time instances. Figure 3.5 indicates the samples which are measured when the UE is set to be stationary while Figure 3.6 presents the samples which are estimated when the UE is moving with a speed of 100km/h between two beams as the scenario explained previously.

In order to generate more training examples, the data is sampled by a rolling-window as shown in Figure 3.7, each window is a training example of the neural network, the window size determines how long the UE is classified to be moving or stationary. To be able to train the data in the machine learning algorithms, each training example is labeled according to the speed statement of UE.

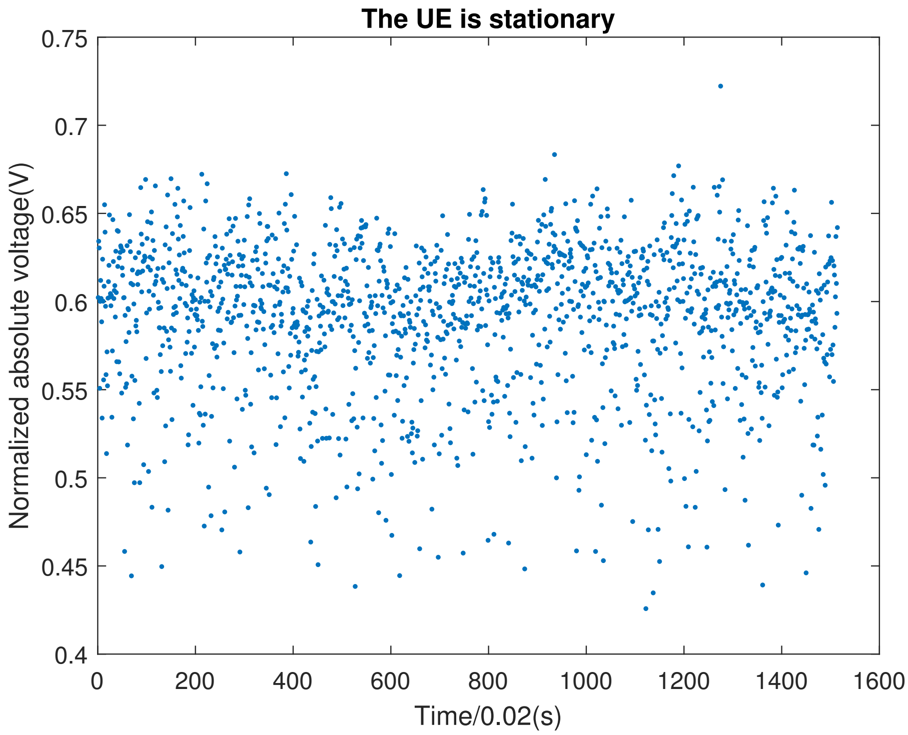

Figure 3.5: The value of the samples vs time/0.02 when the UE is stationary

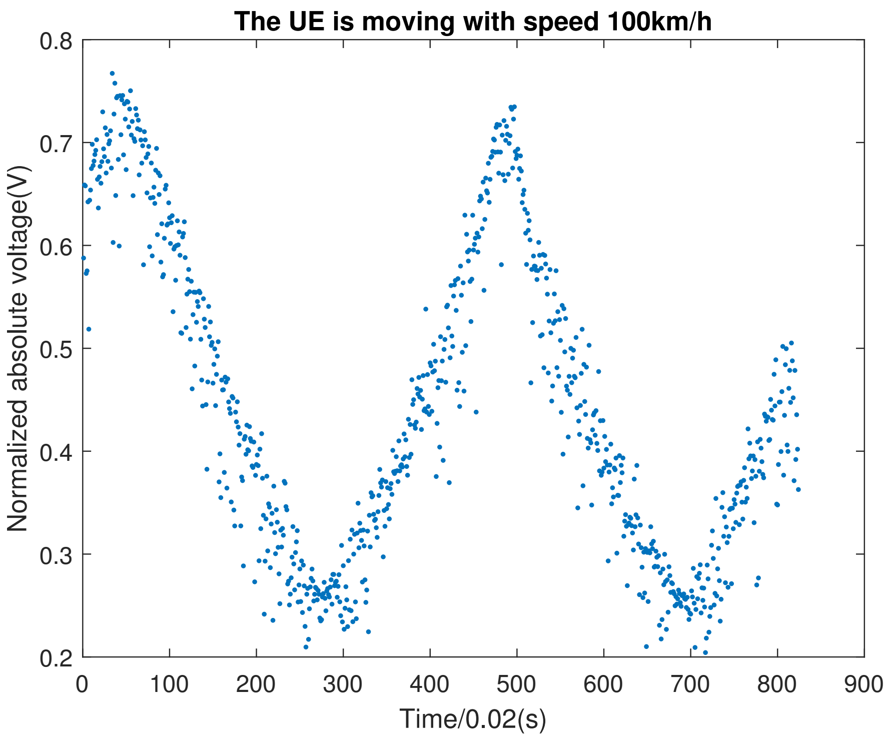

Figure 3.6: The value of the samples vs time/0.02 when the UE is moving with speed 100 km/h

Additionally, the number of training examples with different labels are equally distributed among the number of input training examples. As there are more than one UE in the real world scenario, transmitting the channel estimation can cause interference to other UEs and consume uplink physical resources. By decreasing the frequency of the samples, the resources consumed and interferences can be reduced. To determine the frequency of the samples and the window size, different combinations of them are implemented, and the best pair is chosen such that the machine learning algorithms have a relatively high accuracy for both training datasets and new input datasets.

Figure 3.7: Illustration of data sampling

In this thesis, all of the three machine learning models: neural network, support vector machine and logistic regression are implemented in MATLAB. The Deep Learning Toolbox and the Optimization Toolbox are used in neural network and logistic regression respectively. The neural network model is constructed in the neural network start Graphical User Interface (GUI) and the pattern recognition and classification as machine learning techniques are chosen for designing neural networks.

3.2 Neural Networks

The neural network machine learning algorithm is used to predict the velocity of an UE. The total number of collected training examples are arbitrarily divided into 70% training, 15% cross validation and 15% test dataset. The training dataset is fed into the neural network to train the model by updating the weights of connections between the neurons of the model. The validation dataset provides the unbiased evaluation of a fitted model. Training of the model will stop when the error on validation set increases, which is a sign of overfitting of the training dataset. Overfitting occurs when a model produces relatively small error on training dataset but much larger error when new data input in the model. The test dataset is used to provide an unbiased evaluation of the final model fit on the training dataset [17].

By training channel estimation with a suitable sample rate and window size, the characteristic of the UE can be decided to be stationary or moving by a binary neural network, furthermore, the velocity can be decided to be 100km/h, 30km/h or stationary by a multiclass neural network. In the Deep Learning Toolbox, the number of neural network layers is fixed but the number of neurons in each layer can be decided according to the features of data available.

3.2.1 Binary Neural Network

The structure of the binary neural network is shown in Figure 3.8. The input layer combines 50 input neurons, while the hidden layer is composed of 10 neurons and one neuron represents the output layer. The training examples are labelled by 1s and 0s at input of the neural network. The value of the output layer is either a 1 or a 0, which determines that the UE is moving or stationary. The sigmoid (logistic) function is implemented as activation function between each two layers. The data is sampled at a rate of 1/6 , which means that sampling occurs at every 6th continuous data value, representing one input neuron of neural network as shown in Figure 3.8, the time period of UE to be classified is 6 seconds which can be calculated as multiplication of the reciprocal of the sample rate 1/6, number of input features 50 and time between each two channel estimation measurements, 20 milliseconds.

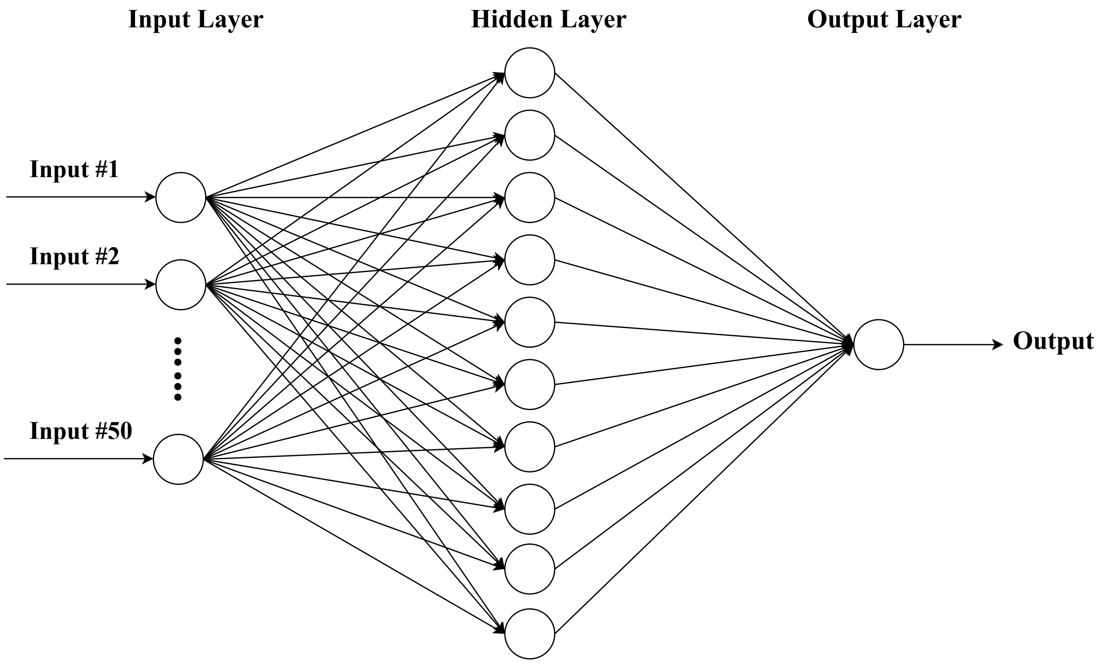

Figure 3.8: The structure of binary neural network

3.2.2 Multiclass Neural Network

The structure of the multiclass neural network is shown in Figure 3.9, The number of neurons at input layer, hidden layer and output layer are 80, 10 and 3 respectively. The sample examples are labelled by a 3-bit binary column vector, where one of the 3 bits is flagged as 1 and the rest as 0 according to the speed categories: stationary, 30km/h or 100km/h as shown in Table 3.1 below. When training the multiclass model, three labelled datasets are sampled by the same sample frequency as the inputs of the multiclass neural network. After attempting a combination of different sample rates and window sizes, 1/8 is chosen as the most effective sample rate.The time period of UE to be classified is 12.8 seconds. The output of the neural network should be as close as one of the three labels as possible, e.g. if the value of first neuron appears to be close to the 1 and the rest close to 0, the UE is categorized as stationary.

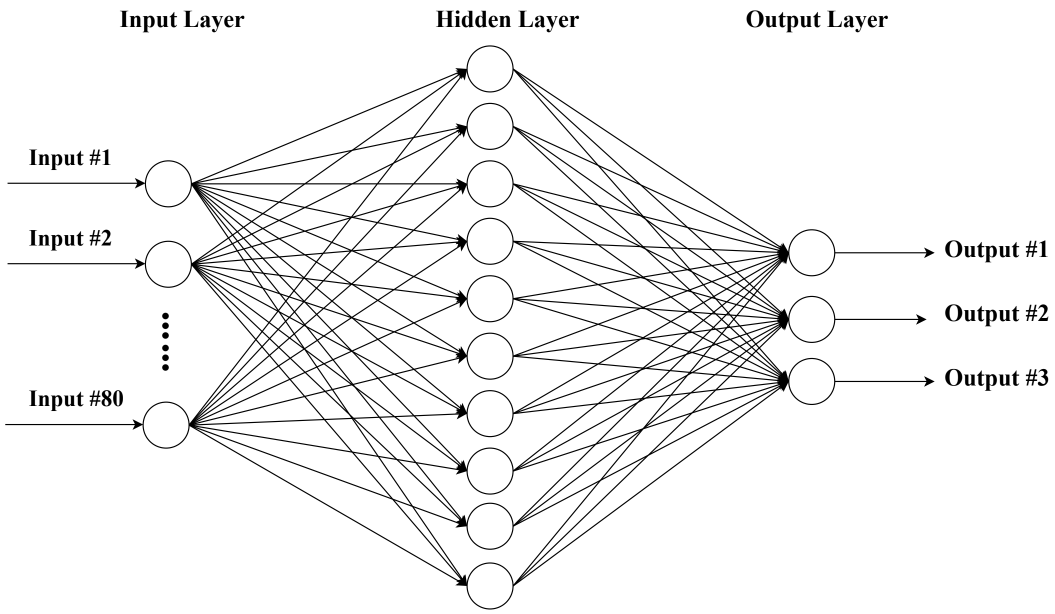

Figure 3.9: The structure of multiclass neural network

	Stationary	30 km/h	100 km/h
1st bit of the label	1	0	0
2nd bit of the label	0	1	0
3rd bit of the label	0	0	1

Table 3.1: The sample example is labelled according to the speed

3.3 Support Vector Machine

The Radial Basis Function (RBF) kernel, also known as Gaussian kernel, is implemented as the only kernel function in this thesis. There are some choices of kernel functions provided in [26], and a kernel function can also be set manually.

In contrast to a neural network, inputting a large number of training examples (e.g. 50000+) is very time consuming while training the SVM model. To refrain from this, the large training examples are all labelled, but only 1000 of them are randomly chosen as input data. The 1000 training examples are divided into 80% training dataset and 20% test dataset. The training dataset is used to train the machine learning model and the test dataset is implemented to measure the accuracy. Different window sizes and sample rates were investigated, the accuracies calculated and visualized with window sizes for sample rate equal to 1. Figure 3.10 below shows that with the window size increasing from 2 to 92, at a size larger than 22, the accuracy is substantially stable, ranging between 93% and 97%. Figure 3.11 indicates the accuracy with window sizes between 2 and 2500, where the accuracy can reach 100% when the window size is larger than 600. Since it is not necessary to reach 100% precision, it needs to be taken into consideration that the larger window size causes a larger time period needed to classify the UE. In this thesis, a window size of 5 was selected and a sample rate of 1/4 was chosen to get relatively high accuracy for both, the test dataset and new input data. So the time period of the UE to be classified is only 0.4 seconds.

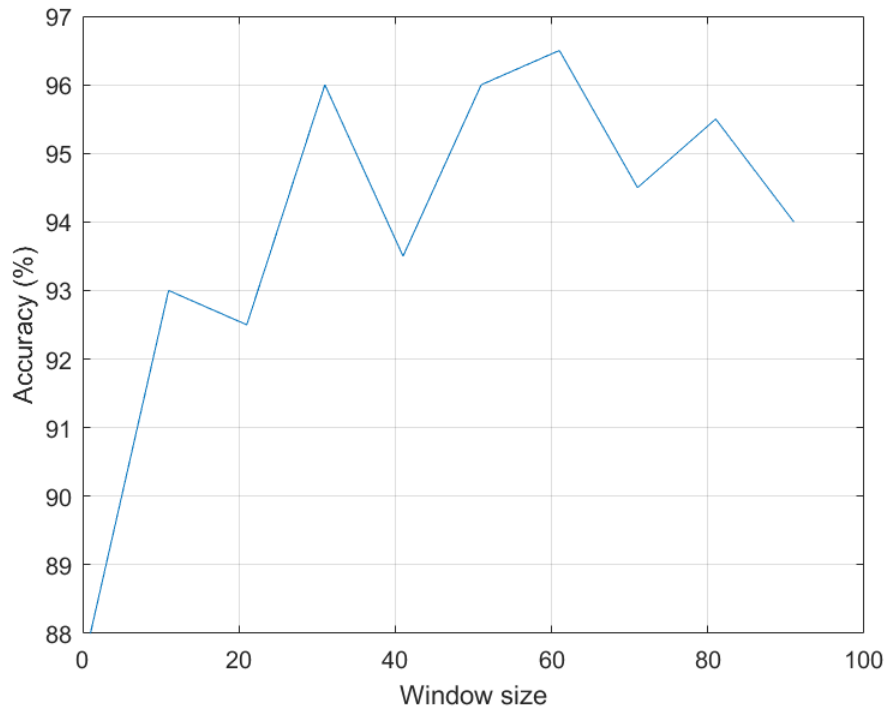

Figure 3.10: Investigation of accuracy with window size from 2 to 92 for SVM method

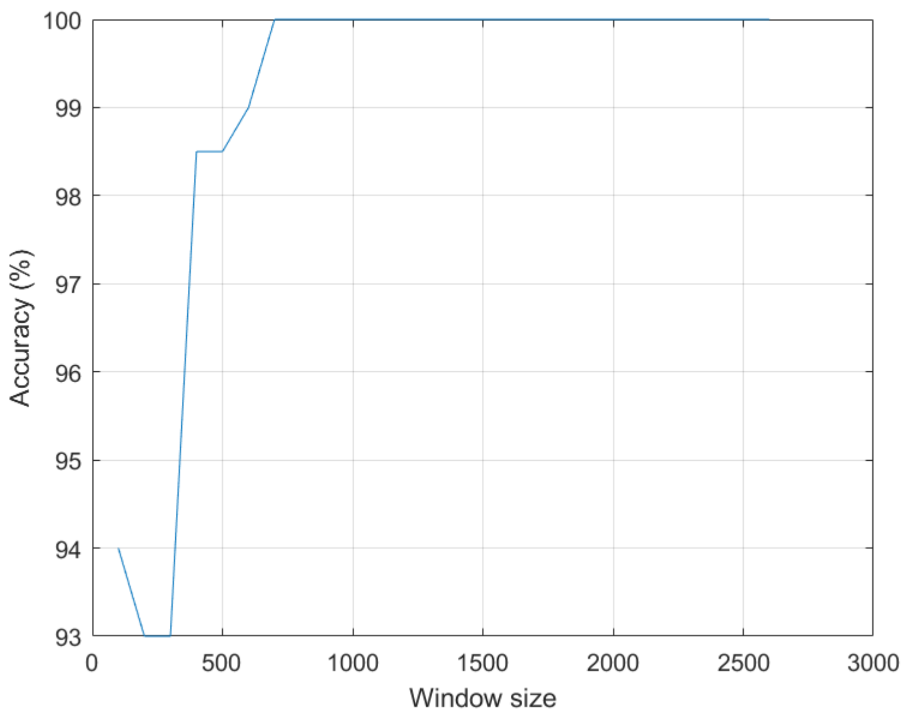

Figure 3.11: Investigation of accuracy with window size from 2 to around 2500 for SVM method

3.4 Logistic Regression

In the logistic regression algorithm, the input features are linearly combined with the weights as a variable input to the logistic function. The idea of the logistic regression is to generate the weight of the features by minimizing the cost function as much as possible.

First, the training examples are labelled and separated into 80% training dataset and 20% test dataset. The training dataset is implemented to calculate the weights of the features, whereas test dataset acts as new input data using the weights to predict the moving or stationary category. Then, the algorithm compares the test dataset with the known labelled categories to estimate the accuracy of the algorithm. Similarly to SVM, different window sizes and sample rates are implemented, for sample rate equal to 1, the accuracy is calculated and visualized with different window sizes as shown in Figure 3.12, the accuracy increases within window size 20 and close to 95% with window size increases until 92, the accuracy can reach 100% when window size is between 1000 and 1500, as shown in Figure 3.13. Again, due to the fact that a larger window size is equivalent to larger time period of UE to be classified, and 95% accuracy is sufficient, the window size is chosen to be 5 and the sample rate is 1/6 in this machine learning algorithm and scenario. Thus, the time period in which the UE is classified is 0.6 seconds.

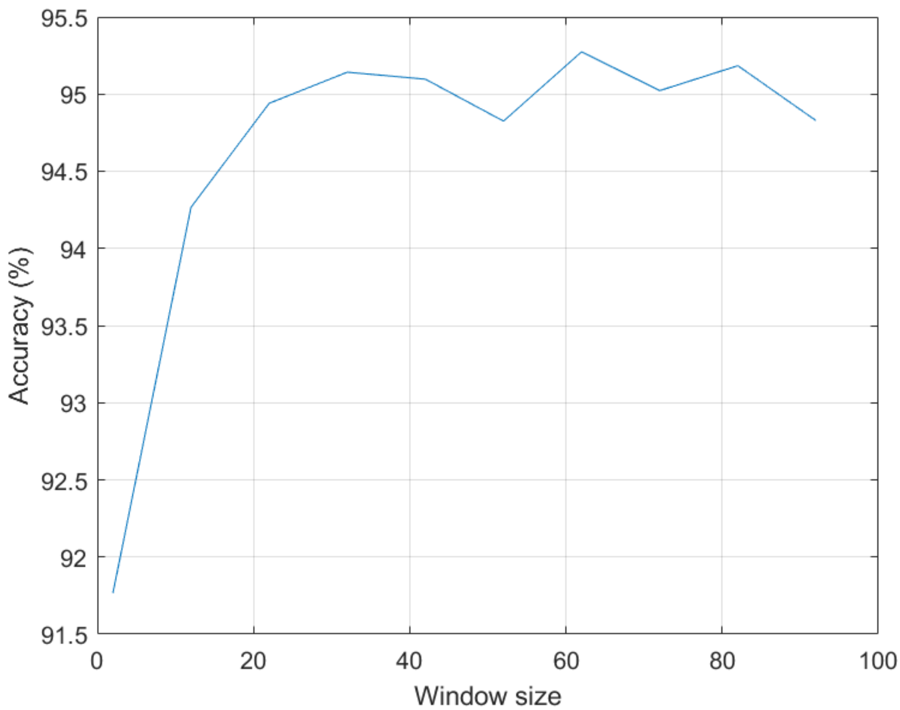

Figure 3.12: Investigation of accuracy with window size from 2 to 92 for for logistic regression method

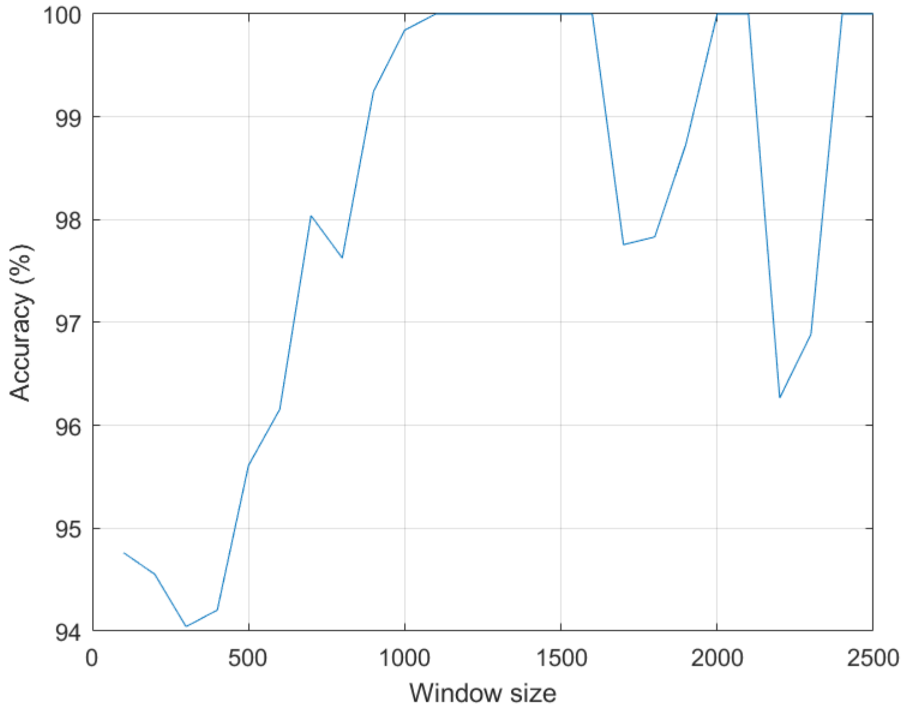

Figure 3.13: Investigation of accuracy with window size from 2 to around 2500 for for logistic regression method

4

Simulation results and analysis

This chapter presents the simulation results of the implemented machine learning algorithms. All simulations are done by keeping the base station beams fixed and arbitrarily locating the UE to measure the channel estimations for the stationary case or by setting the UE route in a random ellipse with fixed speed to simulate movement.

4.1 Neural Network

4.1.1 Binary Neural Network

The output of the binary neural network is a number between 0 and 1, where the threshold value 0.5 is used to classify the moving state. If the output of the binary neural network is larger or equal to 0.5, the decision is made to be 1 which means that the UE is moving; In contrast, the UE is classified as stationary when the output is lower than the 0.5 threshold. Since channel estimation is sampled every 20 milliseconds, as per 3GPP standard, within a time period of 6 seconds the velocity of UE can be classified in the binary neural network, which was discussed with more details in Section 3.2.1.

The confusion matrix for the training dataset, cross validation dataset, test dataset and all dataset are shown in Figure 4.1. In machine learning technology, a confusion matrix allows the visualization of the algorithm performance for the case of statistical classification. Two dimensions: target class and output class, as well as number of data examples in both dimensions construct up this table. Correlation between the two dimensions is defined as: true positive, true negative, false positive or false negative as shown in Table 4.1. The true positive and true negative indicate that the output values are identical with the target values, so the more true positives and true negatives exist, the better. The confusion matrices are constructed for the training, validation, and test set and then for all combined, as can be seen in the figures in Section 4.1 below. In Figure 4.1, each component of the confusion matrix shows the percentage and number of training examples, the 'All Confusion Matrix' shows that 1964 training examples are correctly classified to be moving, which occupies 49.5% of the total; 1952 training examples are classified to be stationary without error, which are 49.3% of the total. Thus, the accuracy of the binary neural network model is the sum of the two correlations which is 98.8%.

Furthermore, 200 labelled data examples were arbitrarily chosen as the new input

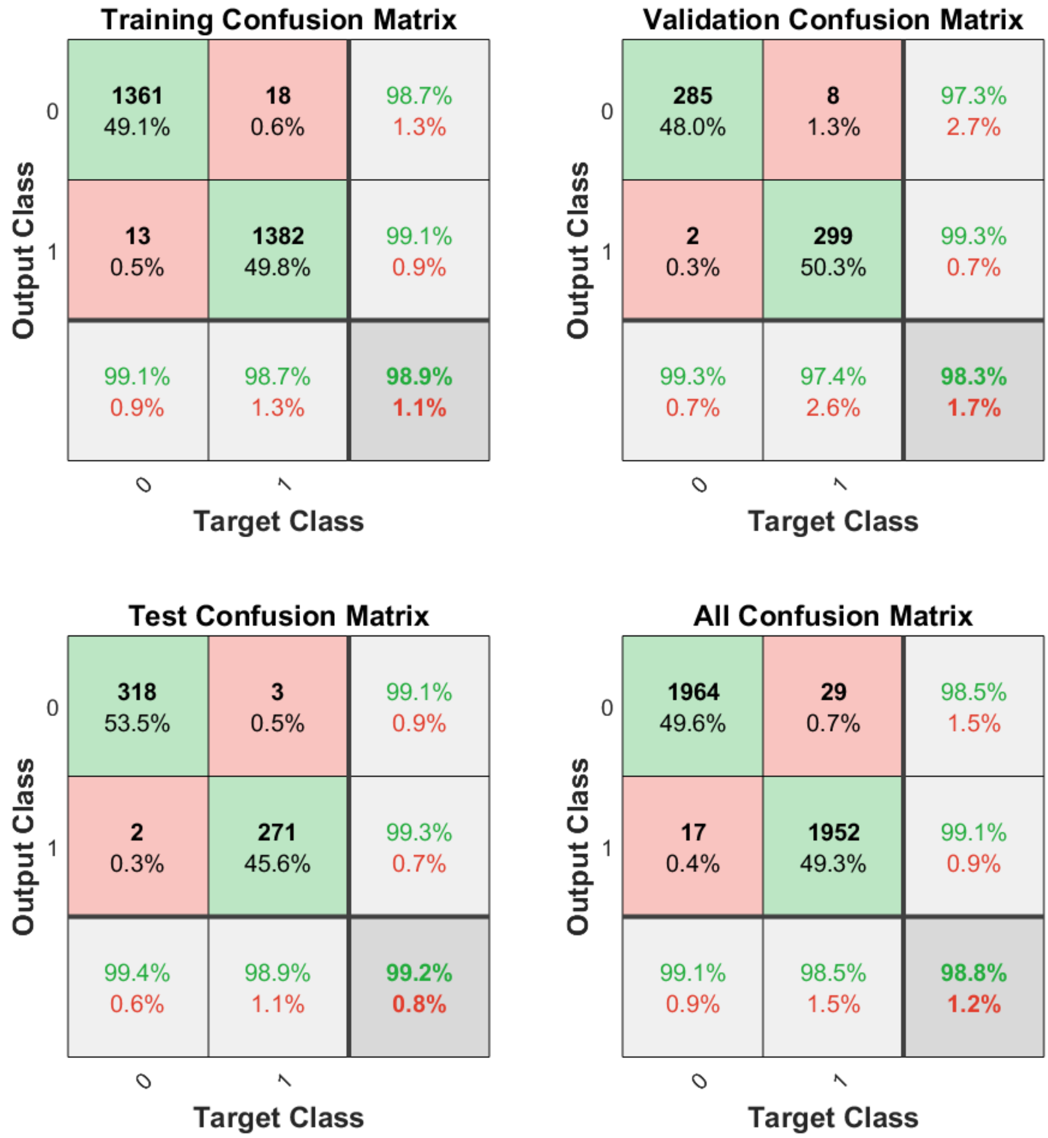

Figure 4.1: Confusion matrix of the binary neural network

dataset, with both moving and stationary properties and implemented in the trained model. The result of using a new dataset with the algorithm is shown in Figure 4.2 below. As presented in its legend, the black points indicate the UE characterization: where 1 represents the UE to be moving with speed 100 kilometer per hour in a 6-second time period, 0 when the UE is stationary, the red squares are the predicted results of the binary neural network. Comparing the labelled and predicted UE characterizations in the binary neural network, it is evident that their correlation for the 200 new input examples is 100%.

4.1.2 Multiclass Neural Network

The output of the multiclass neural network is a 3-bit vector, where the index of the maximum bit is selected as the result. The output is analyzed as follows. If the

		Target class	
		Stationary	Moving
Output class	Stationary	True Negatives	False Negatives
	Moving	False Positives	True Positives

Table 4.1: The confusion matrix and correlation between two classes for binary neural network

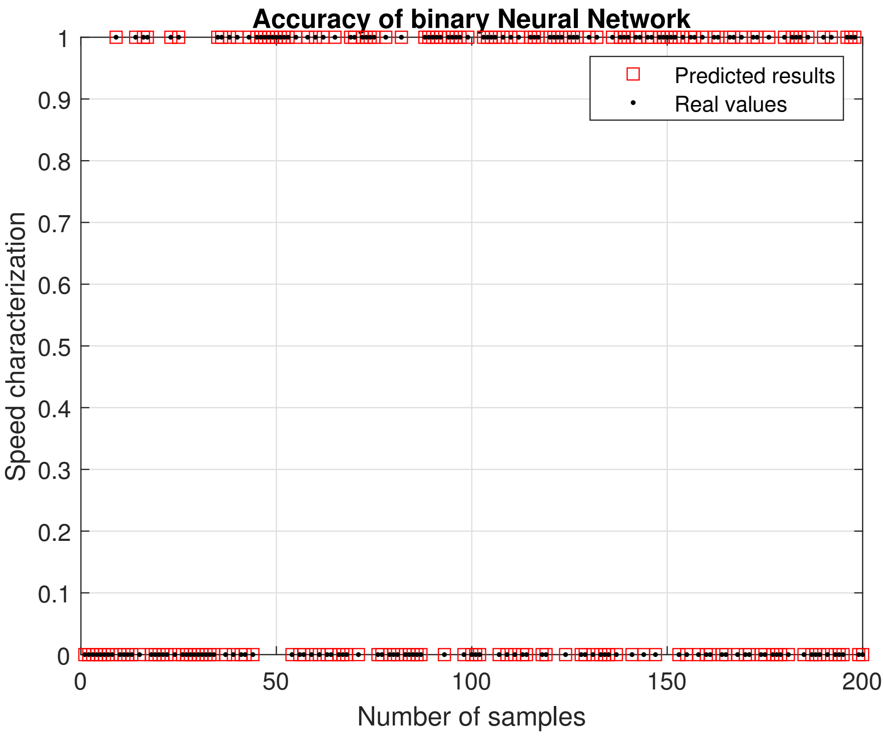

Figure 4.2: Accuracy of the binary neural network with 200 random labelled examples

first bit is the maximum, the UE is decided to be stationary; if the second bit is the maximum, the UE is categorized as having a speed of 30km/h; when the third bit is maximum, then the UE's set to be moving with a speed of 100km/h.

The confusion matrices of the multiclass neural network are shown in Figure 4.3, the first, second and third bit of the output vector are indicated as 1, 2, 3 respectively. The concept of the construction is the same as in the previous section, likewise there are two dimensions: target class and output class, where each class includes three categories, instead of two like in the binary case. The correlation between the two dimensions can be seen in Table 4.2, the diagonal cells give the number and percentage of correct classifications and the off diagonal cells represent

		Target class		
		Stationary	30km/h	100km/h
Output class	Stationary	True Negatives	False Negatives	False Negatives
	30km/h	False Positives	True Positives	False Positives
	100km/h	False Positives	False Positives	True Positives

Table 4.2: Confusion matrix and correlation between two classes for multiclass neural network

the misclassification. The result of the 'All Confusion Matrix' in Figure 4.3 shows that the accuracy of the multiclass neural network is 89.2%.

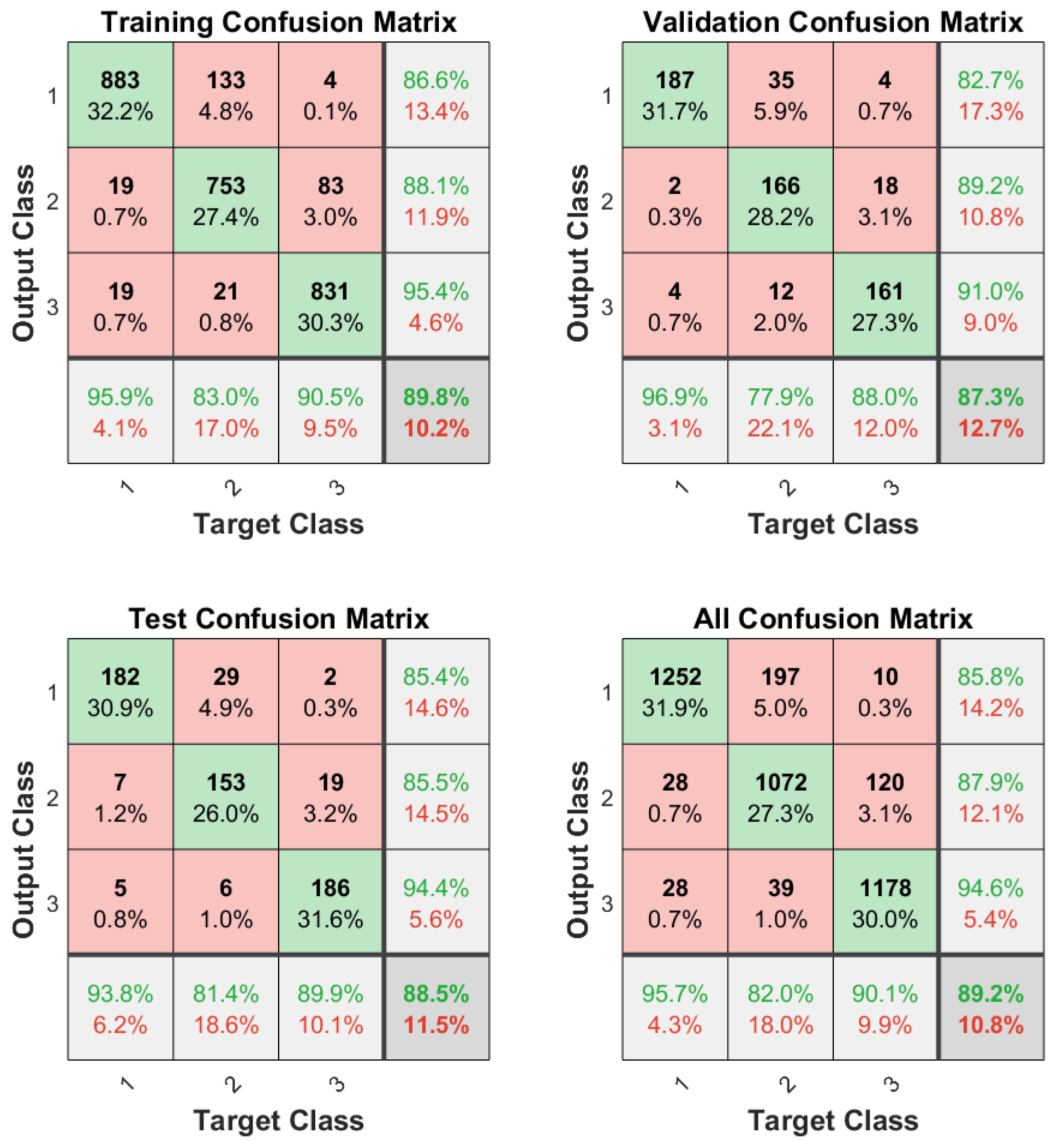

Figure 4.3: The confusion matrix of the multiclass neural network

In addition, 200 labelled training examples are randomly selected with three speed characterizations to input in the model. The Figure 4.4 shows the accuracy of inputting the new data set into the existing trained algorithm, where the horizontal-axis represents the number of training examples and the vertical-axis indicates speed characterization that indicates which bit of the output vector is maximum. The black points in the figure represent the values according to the original known values; the red squares indicate the predicted results of the multiclass neural network; the blue diamonds mark incorrectly predicted training examples. There are 7 incorrectly predicted training examples out of the total 200, resulting in an accuracy of the multiclass neural network for new input data of 96.5%.

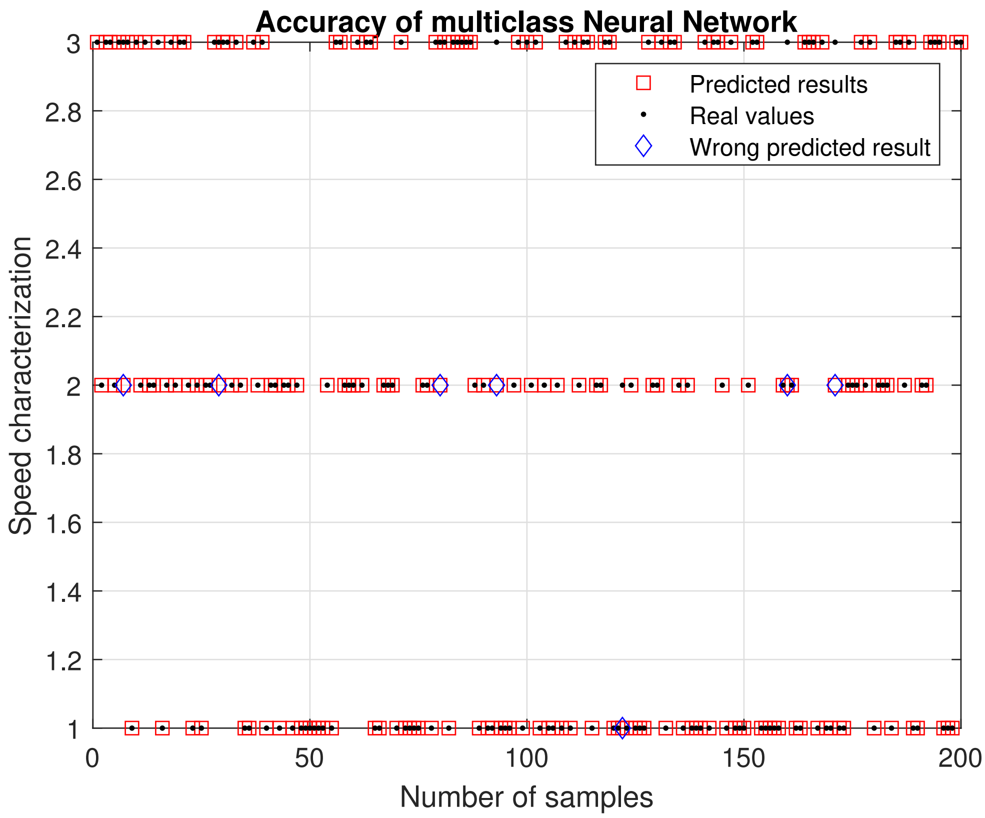

Figure 4.4: The accuracy of the multiclass neural network with 200 new data examples

4.2 SVM

The Figure 4.5 illustrates the development of the optimal objective function found within a given number of function evaluations. The objective function is a loss function in an optimization framework where the goal is to find the most efficient set of parameters which minimize this loss function. The fewer function evaluations needed to minimize the objective function, the better. The figure shows that the objective function value is reduced significantly from 0.5 to around 0.07 within 4 function evaluations, and close to 0.07 with number of function evaluations increases.

The binary SVM classification algorithm searches for an optimal hyperplane, the goal is to optimize the hyperparameter to achieve optimal classification between two classes. A way to do hyperparameter optimization of the objective function is using Bayesian optimization as shown in the 3D plot in Figure 4.6 below. The red surface is the objective function model which shows the performance at each location and the blue points indicate where the parameter space has been sampled. For channel estimation with window size 5 and sample rate 1/4, the accuracy of the binary SVM model is 95%. In this binary SVM classification model, the accuracy of 200 labelled new input data is 100% as can be seen in Figure 4.7.

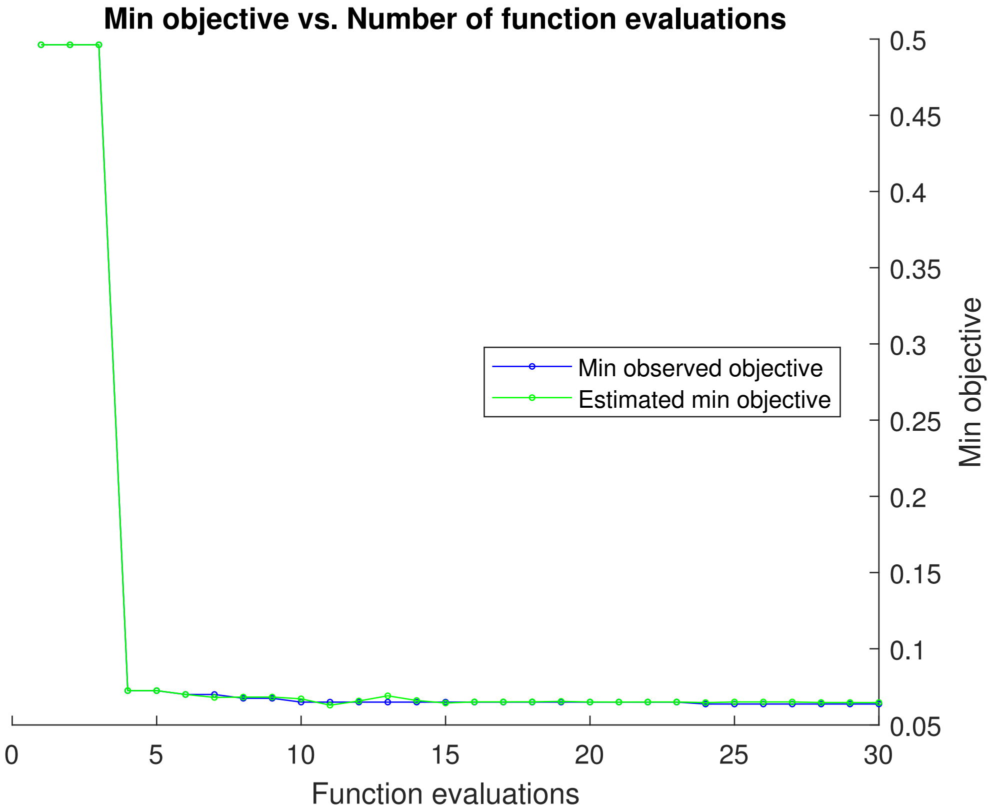

Figure 4.5: The development of the optimal objective function value vs number of function evaluations for observed value and estimated value

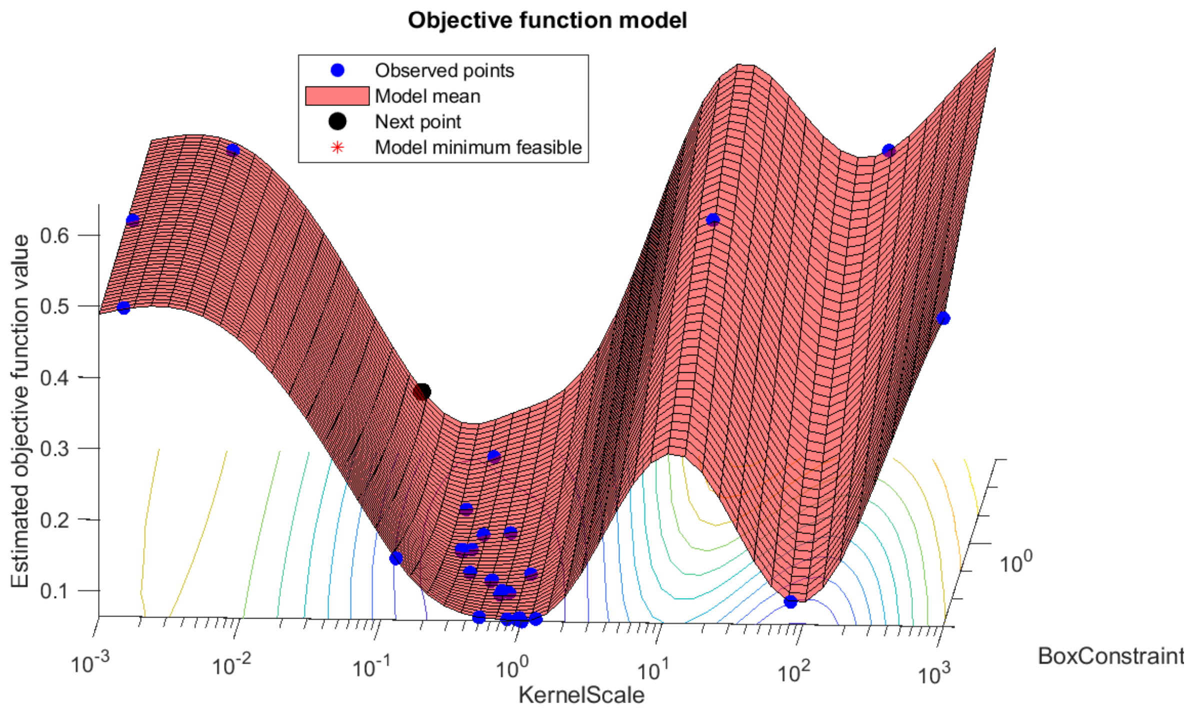

Figure 4.6: The process of hyperparameter optimization in SVM algorithm

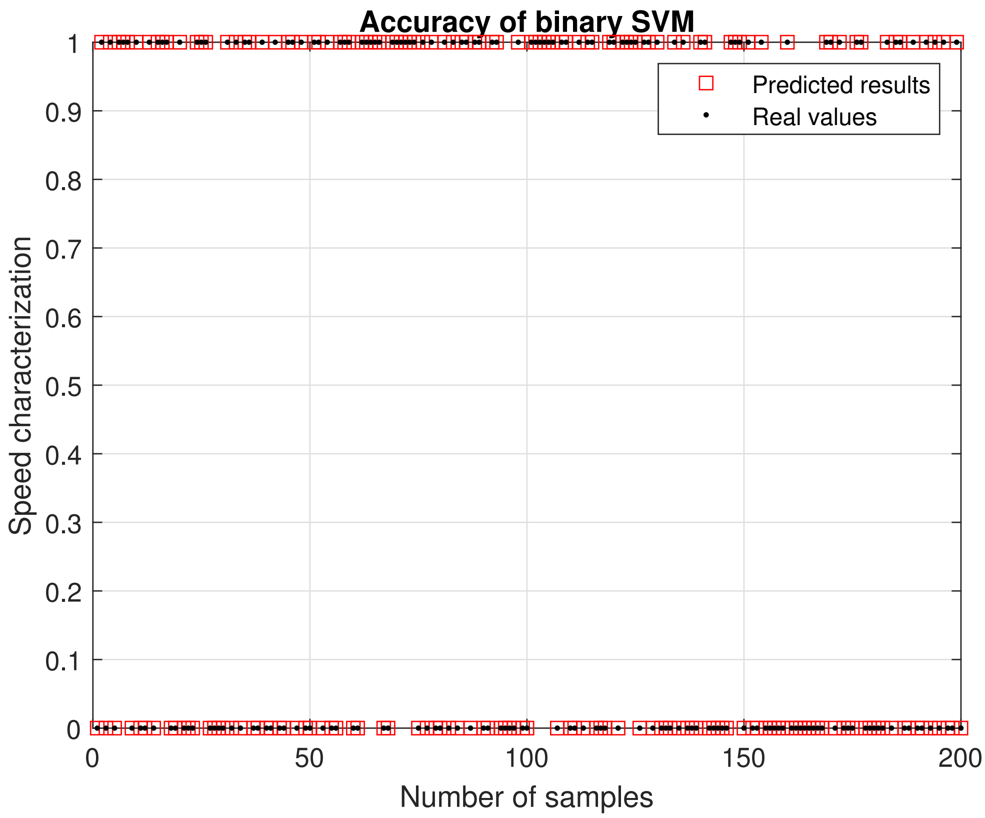

Figure 4.7: The accuracy of SVM with 200 random samples

4.3 Logistic Regression

The accuracy of the logistic regression model is calculated by a labelled test dataset which is 93.8% for input channel estimation with a window size of 5 and a sample rate 1/6. Furthermore, 200 labeled new input examples are tested to estimate the accuracy of new data which, is shown in Figure 4.8, where the red squares represent the predicted results and the black points indicate the labelled values. It is evident that all the black points and the red squares overlap, indicating that the accuracy of the 200 new input dataset is 100%.

4.4 Computational Complexity

Regarding the results from Section 4.1 to 4.3, one can see that all of the three different algorithms can classify the movement of the UE with more than 85% accuracy. Selecting a suitable algorithm according to the property of the data is essential to achieve a higher calculation efficiency. In the following section, the three machine learning algorithms will be analyzed in terms of their computational complexity.

The complexity of a neural network depends on its structure. The larger the number of neurons in each layer, the higher the computational complexity. Considering a feed forward neural network, its computational complexity is lower bounded by $O((\frac{M-1}{D})^D)$, where M indicates the number of neurons in the hidden layer and D represents the number of inputs in the network [39].

Considering a SVM with an RBF kernel, its complexity can be approximated by $O(d^2)$, where d is the number of input dimensions of the algorithm[30]. Using gradient descent, the complexity of logistic regression is $O(d)$ [36].

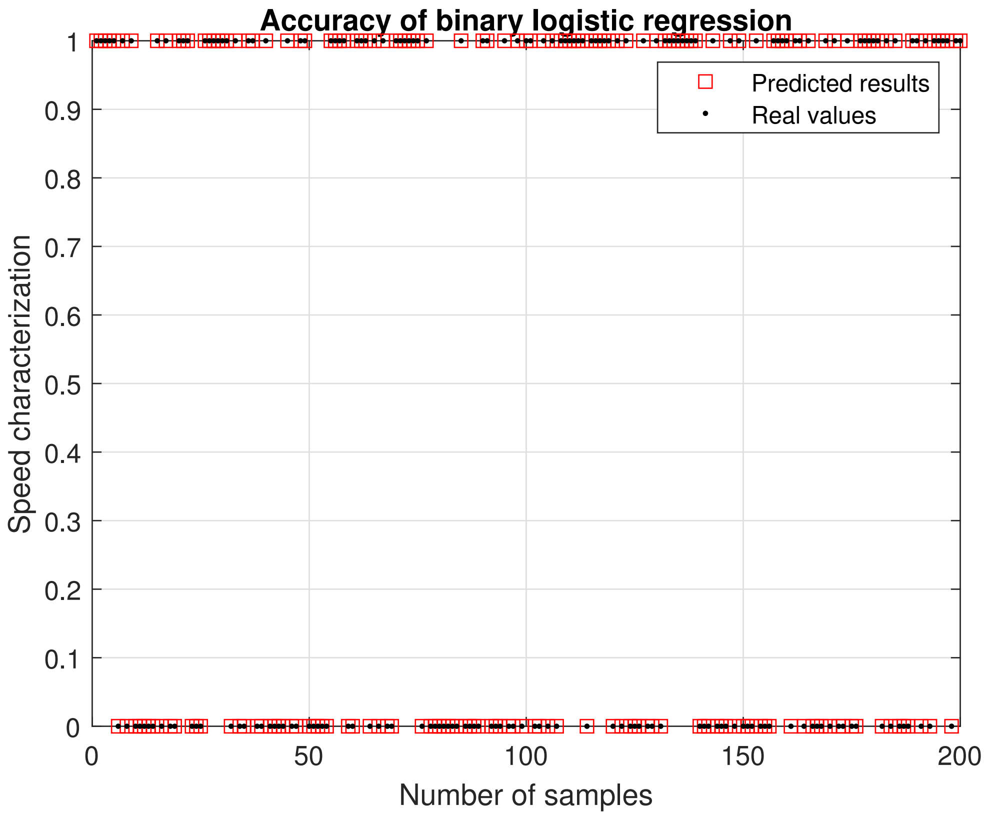

Figure 4.8: The accuracy of the logistic regression with 200 new input data examples

5

Conclusion

In this thesis, the machine learning models of neural networks, SVM and logistic regression have been successfully built and discussed in Section 3.2 to Section 3.4. The neural network algorithms meet the requirements of the thesis, which is stated in Section 1.2 (Purpose and Aims). The binary neural network can successfully classify whether the UE is moving or stationary with 98 % accuracy, whereas and the multiclass neural network can perform speed classification of the UE with 89.2% accuracy. The SVM and logistic regression algorithm were constructed as comparisons to the neural network and are able to perform binary speed classification of the UE with slightly lower accuracy which are 95% and 93.8% respectively. As a summary, the objectives of the thesis have been met and the initial questions have been investigated and answered.

5.1 Future Work

If we were to continue to work on this project, the first step would be to build an SVM and a logistic regression model of multi-class classification and study their performances compared to the neural network. Furthermore the improvement would be to design the structure of neural network with low computational complexity using Python, enabling the customizing of the number of layers and more. Additionally, improvements to the wireless system would be using actual collected data logs from a base station to adjust the parameters of the machine learning models to suit real world environments. It would help refine the parameters and accuracy of the model with values that have been collected in non-ideal channel conditions and with interferences. Finally, using multiple UEs in the scenario would improve its resemblance to real-life.

Bibliography

[1] M. Kibria, K. Nguyen, G. Villardi, O. Zhao, K. Ishizu and F. Kojima, "Big Data Analytics, Machine Learning, and Artificial Intelligence in Next-Generation Wireless Networks", IEEE Access, vol. 6, pp. 32328-32338, 2018. Available: 10.1109/access.2018.2837692.

[2] C. Studer, S. Medjkouh, E. Gonultas, T. Goldstein and O. Tirkkonen, "Channel Charting: Locating Users Within the Radio Environment Using Channel State Information", IEEE Access, vol. 6, pp. 47682-47698, 2018. Available: 10.1109/access.2018.2866979.

[3] A. Hatahet, A. Jain, " Beamforming in LTE FDD for Multiple Antenna Systems", M.S. thesis, Dept. Elect. Eng., Lund Univ., Lund, Sweden, 2018.

[4] M. Shafi, J. Zhang, H.Tataria, A. F. Molisch, S. Sun, et al., "Microwave vs. Millimeter-Wave Propagation Channels: Key Differences and Impact on 5G Cellular Systems", IEEE Communications Magazine, vol. 56, no. 12, pp. 14-20, Dec 2018. Available: https://ieeexplore.ieee.org/document/8570034.

[5] J. Vieira, E. Leitinger, M. Sarajlic, X. Li, and F. Tufvesson. Deep Convolutional Neural Networks for Massive MIMO Fingerprint-Based Positioning. Accepted in the IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), 2017.

[6] Li, J. , Wang, L. , Brault, J. J. , & Conan, J. . (2007). Mobile Location in MIMO Communication Systems by Using Learning Machine. Conference on Electrical & Computer Engineering. IEEE

[7] E. Dahlman, S. Parkvall and J. Skold, 4G, LTE Evolution and the Road to 5G, 3rd ed. Elsevier Science and Technology Books, Inc., 2016, p. 185, 564.

[8] M. Haddad D. Herculea, E. Altman, N. B. Rached, V. Capdevielle, et al., "Mobility State Estimation in LTE", IEEE Wireless Communications and Networking Conference, April 2016. Doha, Qatar. Available: https://hal.inria.fr/hal-01291728/document.

[9] H. J. Kwon, J. H. Lee and W. Choi, "Machine Learning-Based Beamforming in Two-User MISO Interference Channels," 2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIC), Okinawa, Japan, 2019, pp. 496-499.

[10] Zhao, M., Tian, Y., Zhao, H., Abu Alsheikh, M., Li, T., Hristov, R., Kabelac, Z., Katabi, D. and Torralba, A. RF-Based3DSkeletons. SIGCOMM'18 Budapest, Hungary: Massachusetts Institute of Technology. 2018

[11] H. Huang, J. Yang, H. Huang, Y. Song and G. Gui, "Deep Learning for Super-Resolution Channel Estimation and DOA Estimation Based Massive MIMO System," in IEEE Transactions on Vehicular Technology, vol. 67, no. 9, pp. 8549-8560, Sept. 2018.

[12] "3GPP TS 36.104 V8.2.0", Qtc.jp, 2008. [Online]. Available: http://www.qtc.jp/3GPP/Specs/36104-820.pdf. [Accessed: 14- May- 2019].

[13] E. Larsson, O. Edfors, F. Tufvesson, T. Marzetta, "Massive MIMO for next generation wireless systems" IEEE Communications Magazine. 2014;52(2):186-195.

[14] E. Larsson and L. Van der Perre, "Massive MIMO for 5G - IEEE Future Networks", Futurenetworks.ieee.org, 2017. [Online]. Available: https://futurenetworks.ieee.org/tech-focus/march-2017/massive-mimo-for-5g.

[15] S. Gunnarsson, J. Flordelis, L. Van der Perre and F. Tufvesson, "Channel Hardening in Massive MIMO — A Measurement Based Analysis", Lund University & KU Leuven, 2018.

[16] "Massive MIMO increasing capacity and spectral efficiency", Ericsson.com. [Online]. Available: https://www.ericsson.com/en/networks/trending/hot-topics/5g-radio-access/massive-mimo. [Accessed: 10- May- 2019].

[17] "Training, validation, and test sets", En.wikipedia.org, 2019. [Online]. Available: https://en.wikipedia.org/wiki/Training_validation_and_test_sets[Accessed: 3- May- 2019].

[18] C. Hsu, C. Lin. "A comparison of methods for multiclass support vector machines", IEEE Trans Neural Netw. 2002;13(2):415-425. doi:10.1109/72.991427

[19] C. Cortes, V. Vapnik, "Support-vector network", Machine Learning, vol. 20, pp. 273-297, 1995.

[20] J. Niu, D. Lee, T. Su, G. Li, and X. Ren, "User classification and scheduling in lte downlink systems with heterogeneous user mobilities," IEEE Transactions on Wireless Communications, vol. 12, no. 12, pp. 6205–6213, December 2013.

[21] S. M. Musa and N. F. Mir, "An analytical approach for mobility load balancing in wireless networks." CIT, vol. 19, no. 3, pp. 169–176, 2011.

[22] A. Dua, F. Lu, and V. Sethuraman, "Speed-adaptive channel quality indicator (cqi) estimation," Jun. 28 2012, uS Patent App. 13/253,039. [Online]. Available: http://www.google.com/patents/US20120163207

[23] A. Prasad, P. Lunden, O. Tirkkonen, and C. Wijting, "Energy-efficient flexible inter-frequency scanning mechanism for enhanced small cell discovery," in Vehicular Technology Conference (VTC Spring), June 2013, pp. 1–5.

[24] C. M. Bishop. "Pattern Recognition and Machine Learning". Springer, 2006, p227-228.

[25] S, Westlinder, “Video Traffic Classification”, Master Thesis, Karlstad University, Sweden.

[26] “Kernel method”. En.wikipedia.org, 2019. [Online]. Available: https://en.wikipedia.org/wiki/Kernel_method. [Accessed: 16- May- 2019]

[27] “Logistic function”.En.wikipedia.org, 2019. [Online]. Available: https://en.wikipedia.org/wiki/Logistic_function. [Accessed: 17- May- 2019]

[28] A. Molisch, Wireless communications, 2nd ed. Chichester: John Wiley & Sons, 2014.

[29] L. Bernado, T. Zemen, F. Tufvesson, A. Molisch and C. Mecklenbrauker, ”Delay and Doppler Spreads of Nonstationary Vehicular Channels for Safety-Relevant Scenarios”, IEEE Transactions on Vehicular Technology, vol. 63, no. 1, pp. 82-93, 2014.

[30] M. Claesen, F. De Smet, J. A. K. Suykens, et al. ”Fast Prediction with SVM Models Containing RBF Kernels[J]”. Computer Science, 2014.

[31] F. de Figueiredo, F. Cardoso, I. Moerman and G. Fraidenraich, ”Channel Estimation for massive MIMO TDD systems assuming pilot contamination and flat fading”, EURASIP Journal on Wireless Communications and Networking, 2018.

[32] M. Viswanathan, ”Characterizing a MIMO channel – Channel State Information (CSI) and Condition number”, GaussianWaves, 2014. [Online]. Available: https://www.gaussianwaves.com/2014/08/characterizing-a-mimo-channel/. [Accessed: 13- May- 2019].

[33] ”Massive MIMO increasing capacity and spectral efficiency”, Ericsson.com. [Online]. Available: https://www.ericsson.com/en/networks/trending/hot-topics/5g-radio-access/massive-mimo. [Accessed: 10- May- 2019].

[34] E. Ali, M. Ismail, R. Nordin and N. Abdulah, ”Beamforming techniques for massive MIMO systems in 5G: overview, classification, and trends for future research”, Frontiers of Information Technology & Electronic Engineering, vol. 18, no. 6, pp. 753-772, 2017. Available: https://link.springer.com/article/10.1631/FITEE.1601817

[35] K. Fredenslund. Computational Complexity Of Neural Networks. March 25, 2018 [Online]. Available:https://kasperfred.com/posts/computational-complexity-of-neural-networks [Accessed: 30- May- 2019].

[36] Aalto University School of Electrical Engineering. Lecture 8: Gradient descent and logistic regression. 17.11.2016 [Online]. Available: https://mycourses.aalto.fi/pluginfile.php/393629/mod_resource/content/1/Lecture8.pdf

[37] MathWorks. What is a Neural Network? [Online]. Available: https://se.mathworks.com/discovery/neural-network.html [Accessed: 3- June-2019]

[38] B. Sahu, ”Design and Implementation of 4x4 Butler Matrix”, Indian Institute of Science, Bangalore.

[39] Bianchini M , Scarselli F . On the Complexity of Neural Network Classifiers: A Comparison Between Shallow and Deep Architectures[J]. IEEE Transactions on Neural Networks and Learning Systems, 2014, 25(8):1553-1565.

[40] S. Lepkowski, ”An ultra-compact and low loss passive beamforming network integrated on chip with off chip linear array”, Georgia Institute of Technology, 2015.

[41] ”Scale Fading - an overview — ScienceDirect Topics”, Sciencedirect.com, 2019. [Online]. Available: https://www.sciencedirect.com/topics/engineering/scale-fading. [Accessed: 27- May- 2019].

[42] M. Skolnik, ”Doppler Frequency Shift”, Sciencedirect.com, 2002. [Online]. Available: https://www.sciencedirect.com/topics/engineering/doppler-frequency-shift. [Accessed: 27- May- 2019].

[43] ”About Fading”, Rfmw.em.keysight.com, 2019. [Online]. Available: http://rfmw.em.keysight.com/wireless/helpfiles/n5106a/about_fading.htm. [Accessed: 28- May- 2019].

[44] ”Path loss”, En.wikipedia.org, 2019. [Online]. Available: https://en.wikipedia.org/wiki/Path_loss. [Accessed: 01- Jun- 2019].

[45] H. Ngo, E. Larsson and T. Marzetta, ”Aspects of Favorable Propagation in Massive MIMO”, 2014. Available: https://arxiv.org/abs/1403.3461.

[46] A. Garcia-Rodriguez, V. Venkateswaran, P. Rulikowski and C. Masouros, ”Hybrid Analog–Digital Precoding Revisited Under Realistic RF Modeling”, IEEE Wireless Communications Letters, vol. 5, no. 5, pp. 528-531, 2016. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=7539303.

[47] “Butler_matrix” [Online]. 22 April 2019. Available: https://en.wikipedia.org/wiki/Butler_matrix. [Accessed: 07- Jun- 2019].

[48] R. Portengen. “Phased Array Antenna Processing on Recongurable Hardware” https://essay.utwente.nl/735/1/scriptie_Portengen.pdf

[49] R. Apelfröjd, Channel Estimation and Prediction for 5G Applications. Uppsala: Acta Universitatis Upsaliensis, 2018.

[50] ”Beamforming technology – Ericsson”, Ericsson.com, 2019. [Online]. Available: https://www.ericsson.com/en/networks/trending/hot-topics/5g-radio-access/beamforming. [Accessed: 04- June- 2019].

[51] ”Channel state information”. 28 January 2019. [Online]. Available: https://en.wikipedia.org/wiki/Channel_state_information. [Accessed: 09- June- 2019].

[52] ”Machine learning”. 8 June 2019. [Online]. Available: https://en.wikipedia.org/wiki/Machine_learning. [Accessed: 10- June- 2019]

Printed by Tryckeriet i E-huset, Lund 2019

LUND UNIVERSITY

Series of Master's theses
Department of Electrical and Information Technology
LU/LTH-EIT 2019-712
http://www.eit.lth.se