MCS Selection for Throughput Improvement using Machine Learning in Downlink LTE

by

Nazifa Nawar (180021201)

Maisha Mumtaz Rahman (180021206)

E.M.Manwar (180021208)

A Thesis
Submitted to the Department of Electrical & Electronic Engineering

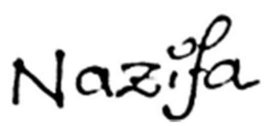
In Partial Fulfillment of the Requirements
for the Degree of Bachelor of Science in Electrical & Electronic Engineering

Islamic University of Technology (IUT)
Gazipur, Bangladesh
May, 2023

Declaration of Authorship

We, Nazifa Nawar (180021201), Maisha Mumtaz Rahman (180021206), E.M.Manwar (180021208), declare that this thesis titled, “MCS Selection for Throughput Improvement using Machine Learning in Downlink LTE”and the works presented in it are our own.We certify that this work has been completed in part to satisfy the requirements for this university's Bachelor of Science in Electrical and Electronic Engineering degree. Additionally, it has been declared that no portion of this thesis has ever been presented elsewhere for the purpose of obtaining a degree. When we have consulted someone else's published work, we have always stated the sources in full.

Authors:



Nazifa Nawar (180021201)

Maisha Mumtaz Rahman (180021206)

E.M.Manwar (180021208)

Certificate of Approval

The thesis titled "MCS Selection for Throughput Improvement using Machine Learning in Downlink LTE" submitted by Nazifa Nawar (180021201),Maisha Mumtaz Rahman (180021206),E.M.Manwar (180021208) has been found as satisfactory and accepted as partial fulfillment of the requirement for the degree of Bachelor of Science in Electrical and Electronic Engineering on 29th May,2023 .

Approved by :

Tawhid
May 29, 2023

(Signature of the Supervisor)
Dr. Mohammad Tawhid Kawser

Professor

Department of Electrical and Electronic Engineering

Islamic University of Technology (IUT)

Acknowledgements

We express our gratitude to the Almighty Allah**. Prof. Tawhid Kauser** of Faculty of Engineering, IUT, is our esteemed supervisor, and we are grateful for his direction and attention in doing this research. He has always encouraged us to expand our knowledge and study new things in order to keep our minds stimulated. We would not be interested in investigating the telecommunication if it weren't for his motivation. We would also like to thank the whole academics of the EEE Department of IUT for their unending support and assistance. Last but not the least, our heartfelt gratitude goes to our family and friends, who has fostered and assisted us in overcoming life's challenges.

Abstract

In Long-Term Evolution (LTE) networks, the desire for high data rates and an improved user experience have driven the demand for effective modulation and coding schemes (MCS) selection techniques in the downlink direction. This thesis investigates the use of machine learning (ML) algorithms to enhance MCS selection and increase throughput in LTE systems.

First, a thorough analysis of the current MCS selection procedures is provided along with a list of their shortcomings. The difficulties brought on by the wireless channel's dynamic nature and the requirement for swift decision-making are highlighted. A innovative ML-based technique is next suggested as an approach to these problems.

The suggested method makes use of system parameters as well as prior channel state data to train ML models, facilitating thoughtful MCS selection options. In terms of throughput performance and complexity, various ML algorithms, including decision trees, support vector machines, and deep neural networks, are compared and evaluated.

The efficiency of the ML-based MCS selection strategy in delivering substantial throughput gains compared to conventional methods is demonstrated by extensive simulations utilizing realistic network configurations. The outcomes open the door for additional research into using ML for next-generation wireless systems and validate the potential of ML for enhancing downlink performance in LTE networks.

Table of Contents

List of tables

List of figures

Chapter 1
INTRODUCTION

1.1 Overview

The Long Term Evolution (LTE) system has been extensively utilized due to the rising demand for faster data transfer rates and better user experiences in mobile communication networks. Higher throughput on LTE networks has become imperative as a result, particularly due to the introduction of bandwidth-intensive applications like online gaming, video streaming, and the Internet of Things (IoT). The rate of successful data transfer from the base station to the user equipment is expressed by throughput, which is essential for achieving the best network performance and exceeding user demands.

The right Modulation and Coding Scheme (MCS) selection is crucial for the downlink LTE system since it affects the transmission's data rate, dependability, and resilience. Traditional MCS selection methods, however, have issues adjusting to dynamic channel conditions and precisely foreseeing the best MCS for throughput enhancement.

By utilizing the potential of machine learning algorithms to increase throughput in the downlink LTE system, this research seeks to solve the constraints associated with conventional MCS selection techniques. Artificial neural networks, support vector machines, decision trees, and reinforcement learning are a few examples of machine learning approaches that allow for the ability to learn from past data, adapt to shifting channel circumstances, and make prudent decisions for the best MCS selection.

With a focus on adaptability and real-time decision-making, the main goal of this thesis is to create and evaluate machine learning-based algorithms for MCS selection in the downlink LTE system. Based on different channel conditions encountered, these algorithms seek to maximize MCS selection. The proposed algorithms will learn complex patterns and relationships from the data, adapt to dynamic channel fluctuations, and make prudent choices to maximize data rates while providing reliable communication. These will do this by utilizing large datasets, computational capacity, and highly sophisticated algorithms.

The results of this study will help enhance MCS selection techniques that are more effective and flexible, enhancing downlink LTE system performance and user experience overall. The results will set the path for future advancements in LTE optimization and network performance improvement by offering insightful information on the effectiveness of different methods using machine learning for MCS selection.

1.2 Evolution of LTE and the Need for Throughput Improvement:

Due to its ability to deliver quick data transmission and improved user experiences, the LTE system has undergone substantial expansion and acceptance in mobile communication networks. Demand for additional throughput in LTE networks has risen as the use of data-intensive applications like online gaming, streaming video, and the Internet of Things (IoT) grows. In order to maintain optimal network performance and satisfy user needs, throughput, which refers to the speed at which data can be effectively transmitted from the base station to the user equipment, is essential.

The demand for increased throughput continues to increase as mobile data usage continues increasing. For a variety of applications, including real-time video streaming, cloud services, and high-quality multimedia content, users increasingly demand smooth connectivity along with rapid data transfer. In order to satisfy these expectations, the downlink LTE system's performance has to be optimized through enhanced throughput.

1.3 Modulation and Coding Scheme (MCS) Selection in LTE:

Within the LTE system, the modulation and coding scheme (MCS) choice is of crucial significance because it directly affects the transmission's data rate, dependability, and robustness. Based on the current channel conditions, this selection procedure involves choosing the most appropriate modulation scheme and coding rate. Even in the presence of variable channel circumstances, the system can achieve higher data rates while guarantees reliable communication by using adequate MCS selection..

Rule-based and model-based solutions have historically been the two main ways used to handle MCS selection. MCS decisions are made utilizing heuristics and specified thresholds in rule-based techniques. Such strategies, however, frequently fail to provide the ability to adapt needed to accommodate constantly shifting channel circumstances. On the other hand, in order to determine the best MCS, model-based strategies, like CQI-based methods, rely on accurate channel estimation and forecasts. Nevertheless, these methods can be computationally taxing as well as prone to errors.

1.4 Challenges and Limitations of Conventional MCS Selection:

Traditional MCS selection methods in LTE have a number of drawbacks and limitations. Despite their simplicity, rule-based approaches have trouble responding to changes in the channel which happen in real time. They are unable to sufficiently take into account dynamic changes in channel conditions, which leads to poor MCS selection choices. In contrast, model-based methods rely significantly on accurate channel estimation and projections, which in real-world situations can be problematic due to things like noise, interference, and environmental changes. These limitations highlight the need for more versatile and perceptive methods to enhance MCS selection in LTE networks.

1.5 Significance of Adaptive MCS Selection:

The efficacy of adaptive MCS selection is crucial for the downlink LTE system's throughput optimization, particularly since channel circumstances vary dynamically. By continually altering the MCS based on in-the-moment channel quality measurements, adaptive selection algorithms are essential for increasing data rates and ensuring dependable communication. This dynamic solution allows for effective resource allocation, lowering the possibility of resource overload or underutilization, and ultimately enhancing network performance and user satisfaction.

The adaptive MCS selection methods are excellent at achieving the appropriate level of dependability while optimizing data throughput. These algorithms can identify what is the most appropriate MCS that offers the highest attainable data rate while assuring reliable communication by continuously monitoring and analyzing channel conditions. This adaptability becomes particularly critical in situations with changing interference levels, fluctuating signal intensities, and changing motion patterns.

1.6 Introduction to Machine Learning in Communication Systems:

Communication systems are one area where machine learning has established itself as a formidable beneficial technology. It takes advantage of large databases, processing power, and advanced algorithms in order to identify patterns, forecast consequences, and improve complex processes. Machine learning algorithms have the capacity to learn from past data, adapt to evolving channel circumstances, and make smart choices to improve throughput and system performance in the context of MCS selection in LTE systems.

A feasible approach to overcoming the drawbacks of traditional MCS selection arrives at is the application of machine learning techniques. Machine learning algorithms have the ability to independently discover patterns and relationships from accessible data, in contrast to traditional approaches that rely on predetermined criteria or models. This enables them to adjust to changing settings and choose MCSs in a context-aware approach. Machine learning algorithms can reveal complex linkages and capture the ever-changing behavior of the wireless channel by utilizing the abundance of data gathered through network measurements, user feedback, and real-time channel conditions. As a result, these algorithms facilitate MCS selection procedures that are more exact and effective.

1.7 Machine Learning Techniques for MCS Selection:

The selection of MCS in communication systems has been thoroughly researched using a variety of machine learning techniques. Artificial neural networks, support vector machines, decision trees, and reinforcement learning algorithms are among these techniques. Each approach has its own advantages and disadvantages, taking into account things like computational complexity, the quantity and quality of training data, and real-time versatility. The applicability and efficacy of these strategies can be determined by analyzing and contrasting them especially in the context of MCS selection within the downlink LTE system.

Artificial neural networks are efficient at extracting complicated patterns and relationships from data because they are modeled after the intricate structure of the human brain. On the other hand, support vector machines feature effective capabilities for classification and regression problems, making them suitable for MCS selection projects.Decision trees offer a clear and understandable structure for deliberative procedures. Agents can learn the most appropriate MCS selection rules through repeated interactions with the environment thanks to reinforcement learning techniques. This research attempts to determine the most efficient method for adaptive MCS selection in the downlink LTE system by studying and evaluating these various machine learning algorithm development..

1.8 Research Gap and Rationale:

Although MCS selection approaches have advanced, there is still a research gap regarding utilizing machine learning to improve MCS selection and optimum throughput in the downlink LTE system. Real-time MCS selection optimization frequently requires adaptability and intelligence that current methodologies frequently lack. The proposed thesis seeks to fill this research gap through the use of machine learning algorithms to overcome the shortcomings of traditional methods and advance better and more versatile MCS selection algorithms for downlink LTE systems.

The predicted advantages that machine learning might bring to MCS selection serve as the primary driving force behind this research. Machine learning algorithms have the capacity to learn from previous data and current measurements, which enables them to select MCSs that are intelligent and context-aware. Machine learning algorithms can significantly enhance the effectiveness and adaptability of MCS selection procedures by taking advantage of the wealth of data present in LTE networks, that includes channel quality indicators, received signal intensity, and signal-to-noise ratios. As a consequence, throughput and overall network performance noticeably increase.

1.9 Research Objectives and Contribution:

The primary objective of this work is to create and evaluate machine learning-based algorithms for MCS selection in order to increase downlink LTE system throughput. The adaptability, real-time decision-making, and channel-condition-based MCS selection optimization are given highest importance by the suggested algorithms. The aim of the study is to increase the effectiveness of MCS selection methods, which will enhance downlink LTE systems' overall performance.

This thesis will investigate the possibility of adaptive MCS selection algorithms that can dynamically adapt to fluctuating channel conditions by using machine learning techniques. Throughput is going to be optimized, and the network's overall performance will be improved, thanks to this adaptability. The study's findings will add to our understanding of the usefulness of different machine learning techniques for MCS selection as well as enhance our understanding of LTE optimization and network performance optimization.

To sum up, the goal of this research is to close the gap between conventional MCS selection methods and the unrealized potential of machine learning for increasing downlink LTE system throughput. This thesis aims to contribute to the creation of intelligent and adaptive MCS selection techniques that can improve network effectiveness, maximize throughput, as well as enhance user experience in LTE systems by addressing the shortcomings and difficulties inherent in current approaches and utilizing the power of machine learning algorithms.

Chapter 2

Literature Review

This paper [1] proposes a modulation-and-coding scheme selection scheme based on the harmonic mean of effective packet-level signal-to-interference-plus-noise ratio (SINR) for LTE systems. The study demonstrates that this scheme achieves performance close to the optimal scheme in terms of throughput improvement. By utilizing the effective packet-level SINR, the proposed scheme effectively adapts the MCS selection to varying channel conditions.

This paper [2] presents optimized modulation and coding scheme selection schemes designed to ensure the block error rate (BLER) performance in LTE systems with poor channel conditions. The research focuses on overcoming the challenges posed by unfavorable channel conditions and proposes techniques to select the most suitable MCS to maintain reliable communication under such conditions.

The paper [3] introduces a modulation and coding scheme selection criterion that leads to a significant capacity gain of approximately 15% in LTE Voice over IP (VoIP) systems compared to conventional methods. The study emphasizes the benefits of the proposed criterion in enhancing the efficiency of VoIP communication.

The paper[4] proposes a modulation and coding scheme selection criterion specifically designed for Adaptive Modulation and Coding (AMC) in LTE systems. The criterion compensates for channel variations over the slot duration to guarantee the required bit error rate (BER) performance, ensuring reliable transmission.

The paper [5] introduces an Ant Colony Optimization (ACO)-based scheme combined with the harmonic mean (HM) approach for modulation and coding scheme selection. The proposed ACO-HM algorithm demonstrates improved performance with fewer resource blocks (RBs) while providing Quality of Service (QoS) guarantees.

1. MCS Selection for Throughput Improvement in Downlink LTE Systems
2. MCS Selection for Performance Improvement in Downlink TD-LTE System
3. A Novel MCS Selection Criterion for VOIP in LTE
4. A novel MCS selection criterion for supporting AMC in LTE system
5. An intelligent optimization algorithm for joint MCS and resource block allocation in LTE femtocell downlink with QoS guarantees

Chapter 3

AIMS AND OBJECTIVES

3.1 Our Proposal

In order to increase performance by reducing the computational complexity and selection time, we present a machine learning-based methodology for MCS selection for downlink LTE. The AMCS selection algorithm might be made more effective and performant by incorporating ML approaches, which would result in quicker and more accurate decision-making.
Based on the needs of various users or applications, AMCS enables the allocation of suitable resources. It can allocate more resources to users with better channel conditions by dynamically modifying the modulation and coding, resulting in higher data rates and better QoS. Real-time channel estimate, quality evaluation, MCS table lookup, and MCS selection are all steps in the selection of the Adaptive Modulation and Coding Scheme (AMCS). Signal processing methods including filtering, demodulation, and error rate estimates are used to analyze received signal properties and compare them to established thresholds or performance metrics. The actual MCS selection process entails assessing the channel quality and choosing the best MCS from the MCS table. These are calculation intensive steps, which despite having low time complexity can be further improved and made power efficient.

3.2 Motivation behind our idea

Based on the needs of various users or applications, AMCS enables the allocation of suitable resources. It can allocate more resources to users with better channel conditions by dynamically modifying the modulation and coding, resulting in higher data rates and better QoS. Real-time channel estimate, quality evaluation, MCS table lookup, and MCS selection are all steps in the selection of the Adaptive Modulation and Coding Scheme (AMCS). Signal processing methods including filtering, demodulation, and error rate estimates are used to analyze received signal properties and compare them to established thresholds or performance metrics. The actual MCS selection process entails assessing the channel quality and choosing the best MCS from the MCS table. These are calculation intensive steps, which despite having low time complexity can be further improved and made power efficient.
To make the system more power efficient we have to reduce the calculations done during each downlink transmission in the transmission time interval. If ML is implemented, then instead of analyzing the channel conditions every time to calculate the MCS index, the system can predict the required MCS index from channel data without doing the calculations. This can reduce the selection time which will eventually aid in the minimization of latency.

3.3 Thesis Objective

The main objectives of this thesis are-

- To minimize the latency in downlink transmission
- To enhance the throughput of the system
- To make the system more energy efficient by reducing the computational complexity at every TTI
- To analyze the performance of different ML algorithms and deciding on the best algorithm

Chapter 4

MCS SELECTION

4.1 Modulation and Coding Scheme (MCS)

In wireless communication systems, modulating data signals into radio waves and encoding them for transmission are techniques referred to as modulation and coding scheme (MCS). It is an essential part of several wireless communication protocols, including those for Wi-Fi, cellular networks (such as 4G and 5G), and satellite communications.

Modulation is the process of altering an information-carrying carrier wave, typically a high-frequency sinusoidal signal. To represent the digital data being transferred, various modulation techniques change particular aspects of the carrier wave, such as its amplitude, frequency, or phase. Amplitude Shift Keying (ASK), Frequency Shift Keying (FSK), Phase Shift Keying (PSK), and Quadrature Amplitude Modulation (QAM) are examples of common modulation techniques. Regarding data throughput, spectral efficiency, and robustness against noise and interference, each modulation scheme offers specific benefits and trade-offs.

Coding is the process of incorporating error-correction codes into the data prior to transmission. Error-correction codes add redundancy to the broadcast signal, enabling the receiver to find and fix potential transmission mistakes. Numerous coding techniques are used, including Turbo codes, convolutional codes, and Reed-Solomon codes. Higher levels of redundancy often offer better resilience to channel impairments. These codes provide various levels of error detection and correction capabilities.

According to the characteristics of the communication channel, such as the signal-to-noise ratio, channel conditions, available bandwidth, and desired data rate, MCS entails choosing an optimal combination of modulation and coding schemes. Different MCS choices are often included in communication standards, enabling the system to dynamically adjust depending on the situation to maximize the trade-off between data throughput and dependability. Lower MCS levels give more dependability at the expense of lower data rates, while higher MCS levels often offer larger data rates but may be more error-prone and require better channel conditions.

4.2 Factors influencing MCS selection.

4.2.1 Channel Conditions

The quality and features of the wireless medium that signals travel through are referred to as channel conditions. These parameters include Doppler shift, fading, multipath effects, noise, signal intensity, interference levels, and frequency selective fading. Channel conditions are improved by strong signals and little interference, whereas noise and fading cause problems. Due to reflections, diffraction, and scattering, multipath effects distort signals. Channel response varies across different frequencies as a result of frequency selective fading. Relative motion causes Doppler shift, which modifies signal frequency. For choosing the proper Modulation and Coding Scheme (MCS) that can reduce impairments and maximize data rate while ensuring dependable communication in wireless systems, it is essential to be aware of certain channel conditions.

4.2.2 Signal-to-Interference-plus-Noise Ratio (SINR)

Signal-to-Interference-plus-Noise Ratio is a key metric used to assess the quality of a received signal in wireless communication systems. It calculates the difference between the targeted signal's power and the sum of its interference and background noise powers. Better channel conditions are implied by a higher SINR, which shows a stronger desired signal in comparison to interference and noise. While noise comes from random sources, interference is caused by signals or sources that use the same frequency band. By taking the SINR into account, MCS selection algorithms can select the proper modulation and coding schemes that can effectively use the channel capacity available, ensuring dependable and high-quality data transmission even in the presence of noise and interference.

4.2.3 Available Bandwidth

Available bandwidth refers to the portion of the frequency spectrum that is allocated for wireless communication in a particular system or environment. It represents the maximum amount of bandwidth that can be utilized for transmitting data. The available bandwidth directly impacts the data rate and capacity of the wireless channel. A wider available bandwidth allows for higher data rates and supports the deployment of advanced modulation schemes with more capacity-efficient coding techniques. Conversely, a limited available bandwidth constrains the achievable data rate and may require the use of more conservative modulation and coding schemes. Optimal utilization of available bandwidth is crucial in MCS selection to maximize the data throughput while adhering to regulatory constraints and avoiding interference with other communication systems.

4.2.4 Data Rate

Data rate describes how quickly digital data is sent or received through a communication system. The amount of data that can be successfully transmitted through the channel in a certain amount of time is measured. The unit of data rate is commonly bits per second (bps) or a multiple of it (e.g., Mbps or Gbps). The modulation and coding system (MCS) used, the available bandwidth, the signal's quality, and the channel circumstances are some of the variables that affect data rate. Faster information transmission made possible by higher data rates allows for the use of applications like real-time communication, huge file transfers, and streaming of high-definition video. Wider bandwidth, dependable channel conditions, and effective modulation methods are required for larger data rates in order to reduce errors and increase throughput.

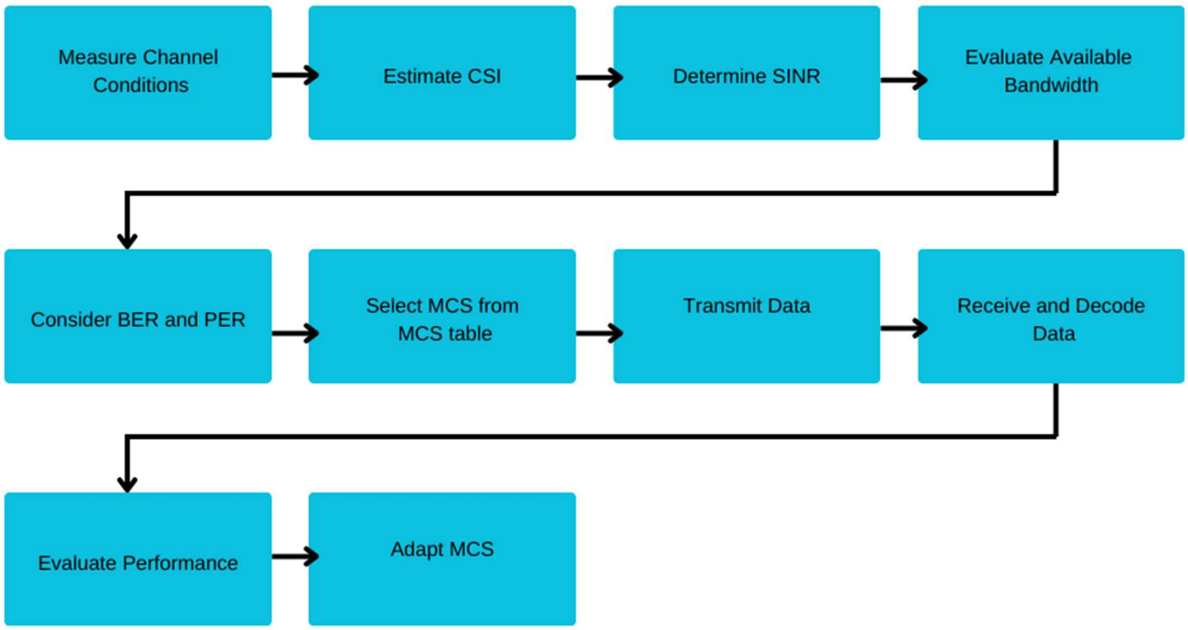
4.2.5 Error Rate Requirements

The allowed degree of mistakes in data transmission in a wireless communication system is referred to as error rate requirements. For reliable data transmission, it specifies the maximum allowable bit error rate (BER) or packet error rate (PER). Based on their unique needs, several applications and services have variable error rate requirements. For instance, low error rates may be necessary for real-time speech or video transmission to preserve quality, yet greater error rates may be acceptable for other data applications. These error rate requirements are taken into consideration during the MCS selection process to make sure that the chosen modulation and coding schemes offer adequate error detection and correction capabilities to satisfy the set error rate targets and retain the acceptable degree of data integrity and reliability.

4.2.6 MCS Table

A wireless communication system's available Modulation and Coding Scheme (MCS) configurations are described in an MCS table, which is a predetermined list of possibilities. It acts as a reference manual for choosing an MCS based on the particular communication standard. Various combinations of modulation schemes, coding rates, and other factors are frequently included in the MCS table. Data rate and error resilience are specifically traded off for each MCS option in the table. The table offers a variety of options, enabling the system to dynamically adapt to the current channel conditions by selecting the most appropriate MCS from the available options to optimize data rate, spectral efficiency, and error performance depending on the needs of the communication link.

4.3 Process of Selection of MCS



Chapter 5

Methodology

5.1 Model Creation

Model Creation for Wireless Network in MATLAB:

To calculate the Signal-to-Interference-plus-Noise Ratio (SINR) for a wireless network consisting of multiple base stations and user equipment (UE), we utilized MATLAB coding. MATLAB provides a powerful and flexible environment for modeling and simulating wireless communication systems.

Here's an overview of the steps involved in creating the model:

5.1.1 Network Topology:

Defined the network topology by specifying the locations and characteristics of the base stations and UEs. This includes the coordinates, transmission power, antenna characteristics, and other relevant parameters.

5.1.2 Path Loss Modeling:

In wireless communication, the signal strength attenuates as it propagates through the environment due to factors such as distance, obstacles, and interference. Applied path loss models to estimate the received signal power at each UE based on the distance between the base station and UE, and other environmental factors. Commonly used path loss models include the Free Space Path Loss (FSPL), Log-Distance Path Loss (LDPL), or the more sophisticated Okumura-Hata or COST models. We implemented the FSPL path-loss model.

5.1.3 Shadow Fading:

Shadow fading refers to the additional signal attenuation caused by obstacles and varying environmental conditions. It introduces random variations in the received signal power. Incorporated a shadow fading model, such as a log-normal distribution, to account for these random variations. Assigned random shadow fading values to each link in the network to simulate realistic signal fluctuations.

5.1.4 Interference Calculation:

Determined the interference experienced by each UE from neighboring base stations. This involves evaluating the received power from other base stations within the network and considering the interference caused by those signals. The interference was estimated by calculating the received power at the UE from other base stations and subtracting the desired signal power.

5.1.5 Noise Calculation:

Took into account the background noise present in the wireless channel. The noise was modeled as additive white Gaussian noise (AWGN) and characterized by its power spectral density. Included the noise power in the SINR calculation to assess the system's performance in the presence of noise.

5.1.6 Resource Allocation:
Allocated Resource Blocks (RBs) to each UE based on the calculated SINR values. Resource allocation algorithms aim to maximize system capacity and fairness among UEs by assigning RBs with favorable SINR levels. Different allocation schemes was implemented, such as Round-Robin, Proportional Fair, or Maximum SINR.

5.1.7 SINR Calculation:
Finally, calculated the SINR for each UE in the network. The SINR is defined as the ratio of the received signal power to the sum of interference and noise power. It quantifies the quality of the received signal and provides insights into the link performance. The SINR values obtained was used for further analysis.
By utilizing MATLAB coding and incorporating path loss modeling, shadow fading, interference calculation, and resource allocation algorithms, we created a comprehensive model for a wireless network. This model enables the calculation of SINR values, which play a crucial role in assessing and optimizing the performance of the network in terms of signal quality and resource utilization.

5.2 Acquiring Dataset
Acquiring Dataset for the Model:

To create a dataset for the wireless network model, we generated 25,000 sample data points by running the model multiple times. The dataset serves as a representative sample of the network's performance and allows for training and evaluation of machine learning algorithms or statistical analysis.

Here's a detailed description of the dataset acquisition process:

5.2.1 Model Execution:
The wireless network model, developed using MATLAB coding, was executed repeatedly to simulate the operation of the network under various scenarios. Each model execution corresponds to a specific set of parameters, such as base station locations, UE positions, path loss models, shadow fading values, and interference levels.

5.2.2 Sample Generation:
During each model execution, the relevant network parameters, including the SINR values and resource allocations were collected. These parameters were captured at the UE level, capturing information about individual user experiences within the network.

5.2.3 Data Collection:
The collected parameters and metrics were recorded and stored as data points for each model execution. This process was repeated several times to ensure a sufficient number of samples for the dataset. The number of repetitions was determined to create a diverse and representative dataset that covers a wide range of network conditions and configurations.

5.2.4 Dataset Composition:

The acquired data from each model execution were aggregated to form a single dataset. The dataset comprises 25,000 sample data points, each representing a specific network configuration and corresponding performance metrics. The dataset provides a comprehensive representation of the network's behavior and allows for robust analysis and training of subsequent algorithms.

5.2.5 Data Validation:

To ensure the quality and reliability of the dataset, validation techniques were employed. This involved verifying the consistency and accuracy of the collected data, checking for potential outliers or errors, and performing data cleansing, if necessary. Additionally, appropriate data splitting techniques, such as randomization or stratification, may be applied to ensure the dataset's representativeness and avoid biases.

5.2.6 Dataset Availability:

Once the dataset acquisition and validation processes were completed, the dataset was made available for further analysis, modeling, or sharing. It can be stored in a suitable format, such as a spreadsheet, CSV file, or a database, to facilitate easy access and manipulation by researchers, data scientists, or machine learning practitioners.

By generating a dataset with 25,000 sample data points through repeated executions of the wireless network model, we have established a valuable resource for training, evaluating, and analyzing the performance of the network. This dataset provides a foundation for conducting statistical analyses, developing machine learning algorithms, or testing various optimization strategies to enhance the efficiency and effectiveness of the wireless network.

5.3 Data Preparation & Processing

Data preparation and processing are essential steps in machine learning (ML) workflows. They involve transforming raw data into a suitable format that can be effectively utilized for model training, validation, and testing. This step ensures that the data is clean, relevant, and appropriately structured to improve the performance and accuracy of ML algorithms.

Here's a breakdown of the data preparation and processing steps in ML:

5.3.1 Data Cleaning:

Data cleaning involves identifying and handling missing values, outliers, and inconsistencies within the dataset. Missing values can be imputed or removed based on the specific circumstances. Outliers, which are data points significantly deviating from the rest, can be addressed through techniques such as removing them, transforming them, or treating them as separate categories. Inconsistencies in the data may arise due to errors or inconsistencies during data collection and require correction or resolution.

5.3.2 Data Integration:

Data integration involves combining data from multiple sources or datasets to create a comprehensive and unified dataset. This step may include merging datasets, resolving inconsistencies in variable naming or encoding, and aligning the data structure to ensure compatibility and consistency across all variables.

5.3.3 Feature Selection:

Feature selection is the process of identifying the most relevant features (variables) in the dataset that contribute significantly to the desired outcome or target variable. This step helps reduce dimensionality and focuses on the most informative attributes, improving the efficiency and effectiveness of the ML model. Techniques such as correlation analysis, feature importance, or domain knowledge can guide the selection process.

5.3.4 Handling Imbalanced Data:

In cases where the dataset exhibits class imbalance, where one class is significantly more prevalent than others, additional steps may be required to address this issue. Techniques such as oversampling the minority class, undersampling the majority class, or using synthetic data generation methods can help balance the classes and prevent biased model performance.

5.3.5 Data Normalization:

Normalization ensures that the data falls within a specific range or distribution to facilitate effective model training. Common normalization techniques include min-max scaling (scaling to a specific range) or z-score normalization (scaling to have zero mean and unit variance). This step helps prevent certain features from dominating the learning process due to their larger scale.

5.4 Feature Correlation

In Python's pandas library, the function corr() is commonly used to calculate the pairwise correlation coefficient between all pairs of features in a dataset. This function allows us to assess the linear relationship between variables and provides insights into the strength and direction of their correlation.

In the specific case of analyzing the correlation between the CQI (Channel Quality Indicator) and SINR (Signal-to-Interference-plus-Noise Ratio) features, the obtained correlation coefficient is approximately 0.45. This value indicates a moderately positive linear relationship between the two variables.

Here's a detailed explanation of the interpretation of the correlation coefficient and its implications:

5.4.1 Correlation Coefficient Range:

The correlation coefficient ranges from -1 to 1. A value of 1 signifies a perfect positive linear correlation, implying that as one variable increases, the other variable also increases proportionally. Conversely, a value of -1 denotes a perfect negative linear correlation, meaning that as one variable increases, the other variable decreases proportionally. A correlation coefficient of 0 indicates no linear correlation between the variables.

5.4.2 Moderately Positive Linear Relation:

With a correlation coefficient of approximately 0.45, the CQI and SINR features exhibit a moderately positive linear relationship. This suggests that as the CQI increases, the SINR tends to increase as well, though not necessarily at a constant rate. The correlation coefficient value of 0.45 indicates a moderate degree of linear dependence between the two variables.

5.4.3 Strength of the Relationship:

The correlation coefficient value of 0.45 implies a moderate strength of the linear relationship between CQI and SINR. While it is not a perfect correlation, it still suggests that there is some consistent pattern or tendency for the two variables to vary together. However, it is important to note that correlation coefficients alone do not indicate the causality or the underlying factors driving the relationship.

5.4.4 Interpreting the Correlation Coefficient:

It is crucial to consider the context and domain knowledge when interpreting correlation coefficients. Factors such as the specific dataset, data collection process, and the nature of the variables being analyzed should be taken into account. Additionally, correlation coefficients only capture linear relationships, and non-linear relationships may not be adequately represented by this measure.

By utilizing the corr() function in the pandas library and obtaining a correlation coefficient of around 0.45 between CQI and SINR, we can conclude that these features have a moderately positive linear relationship.

Figure 5.1: Feature of Correlation Coefficient

5.5 Splitting dataset

When working with machine learning models, it is crucial to split the dataset into separate training and testing sets. This allows us to train the model on a portion of the data and evaluate its performance on unseen data. In Python, the scikit-learn library provides the train_test_split() function, which simplifies the process of splitting the dataset into training and testing sets.

Here is a detailed explanation of the steps involved in splitting the dataset using the train_test_split() function:

5.5.1 Loading the Dataset:

Loaded the dataset into your script or notebook. This was done using various methods depending on the format of the dataset, such as reading a CSV file

5.5.2 Splitting the Dataset:

Used the train_test_split() function to split the dataset into training and testing sets. This function takes the dataset and specifies the desired split ratio. By convention, the dataset is typically represented by two variables: X (containing the input features) and y (containing the corresponding target variable). The function returns four subsets: X_train, X_test, y_train, and y_test.

The split ratio is specified through the test_size parameter, which indicates the proportion of the dataset to allocate for testing. For example, setting test_size=0.2 will allocate 20% of the data for testing, while the remaining 80% will be used for training. Additionally, you can specify other parameters such as random_state to ensure reproducibility of the split.

5.5.3 Training and Testing the Model:

Now that we have the training and testing sets, we proceeded with training and evaluating the machine learning model. Use the X_train and y_train subsets to train the model on the training data. Once trained, used the trained model to make predictions on the testing data (X_test). Compared the predicted values with the actual values (y_test) to evaluate the model's performance.

By employing the train_test_split() function from the scikit-learn library and allocating 80% of the dataset for training and 20% for testing, we trained a machine learning model on the training data and evaluate its performance on the testing data. This approach ensures that the model is assessed on unseen data, providing a reliable estimation of its real-world performance and aiding in the identification of potential issues like overfitting or underfitting.

5.6 Cross validation

In our thesis, we employed K-Folds cross-validation, specifically using the KFold function from the scikit-learn library. We utilized 10 folds, where the training data consisted of 9 folds, and the remaining fold was used for testing. This process was repeated 10 times to ensure a robust evaluation of our machine learning model.

5.6.1 K-Fold Cross-Validation Setup:

Using the KFold function, we set up K-Fold cross-validation by specifying the number of folds as 10. This means our dataset was divided into 10 equally sized folds for training and testing.

5.6.2 Training and Testing:

We iterated through each fold using a loop. In each iteration, we obtained the indices for the training and testing data for that specific fold. The model was then trained on the training data from the current fold and evaluated on the corresponding testing data.

5.6.3 Repeated Process:

We repeated the process for all 10 folds, ensuring that every fold served as the testing set once. This allowed us to obtain a comprehensive assessment of the model's performance across different subsets of the data.

5.6.4 Aggregation of Results:

Finally, we aggregated the evaluation results obtained from each fold to derive an overall performance measure. Aggregation techniques such as calculating the mean or standard deviation of the evaluation metrics across the folds were applied.

By employing K-Folds cross-validation with 10 folds, we ensured a robust evaluation of our machine learning model. The use of 9 folds for training and 1 fold for testing, repeated 10 times, allowed us to thoroughly assess the model's ability to generalize to unseen data. This approach mitigated potential issues of overfitting or underfitting, providing reliable performance estimates for our model.

Cross-validation, especially K-Folds, is a valuable technique in machine learning for obtaining a more comprehensive understanding of a model's performance. By incorporating it into our project, we gained confidence in the reliability and generalizability of our machine learning model's results.

5.7 Model Selection

In our thesis, we explored various machine learning models to find the most suitable one for our task. We considered a range of models, including Gradient Boosting Regressor, SVC, Ridge, PLS Regression, Elastic Net, Linear Regression, MLP Classifier, and KNN. This report summarizes the key details of each model and provides insights into their potential applications.

5.8 Evaluation

In our thesis, we evaluated the performance of our machine learning models using various performance metrics. These metrics provided insights into the models' accuracy, precision, recall, F1 score, and ROC score. This report summarizes these performance metrics and their significance in assessing model performance.

5.9 Hyperparameter Tuning

By employing the GridSearchCV technique for hyperparameter tuning, we were able to enhance the performance of our machine learning models. The process involved systematically searching through different hyperparameter combinations and selecting the best-performing model based on evaluation metrics. As a result, we observed increased accuracy for all models, with SVC showing particularly promising improvements. These optimized models have the potential to provide more reliable and accurate predictions for our specific task.

5.10 Prediction

In our project, we successfully made predictions for the CQI Index, Code Rate, Rate (bit/symbol), and Modulation Order based on various modulation schemes. These predictions are valuable for optimizing the performance and resource allocation in wireless communication systems. By understanding and predicting these parameters, we can design and configure the system to achieve efficient and reliable data transmission in different channel conditions and modulation schemes.

Predicting the CQI Index, Code Rate, Rate (bit/symbol), and Modulation Order is crucial in wireless communication systems. These predictions help in optimizing the transmission parameters and resource allocation, ensuring efficient and reliable communication. By accurately predicting these parameters based on the selected modulation schemes, we can make informed decisions and tailor the communication system to the specific channel conditions and performance requirements.

5.11 Implementation

In our thesis, we implemented the MCS (Modulation and Coding Scheme) selection procedure as described in reference [1]. We created a model in MATLAB to follow the procedure, and we also implemented the same procedure using our machine learning model in Python. Additionally, we calculated and compared the MCS selection duration for both models. This report provides an overview of the implementation process and the comparison results.

5.11.1 Implementation Details:

5.11.1.1 MCS Selection Procedure:

We followed the MCS selection procedure outlined in reference [1] to determine the appropriate Modulation and Coding Scheme for wireless communication.
The procedure involves analyzing various parameters such as signal-to-noise ratio (SNR), channel conditions, coding schemes, modulation schemes, and transmission requirements to select the most suitable MCS.

5.11.1.2 MATLAB Implementation:

We created a model in MATLAB to implement the MCS selection procedure.
The MATLAB model utilized the relevant algorithms and calculations specified in reference [1] to determine the optimal MCS for a given set of input parameters.

5.11.1.3 Python Implementation with Machine Learning:

In addition to the MATLAB implementation, we also implemented the MCS selection procedure using our machine learning model in Python.
We trained our model using appropriate datasets, considering the input parameters and the corresponding optimal MCS obtained from reference [1].

5.11.1.4 MCS Selection Duration Comparison:

To compare the two models, we calculated the MCS selection duration for both MATLAB and Python implementations.
The MCS selection duration refers to the time required for the models to determine the optimal MCS given a set of input parameters.

By comparing the MCS selection durations for both models, we observed the potential advantages of leveraging machine learning techniques for improved efficiency and accuracy in MCS selection. This demonstrates the applicability and potential benefits of incorporating machine learning models in wireless communication systems to enhance the selection of suitable Modulation and Coding Schemes.

Chapter 6

RESULT AND DISCUSSION

6.1 Algorithms Used

6.1.1 Support Vector Classifier (SVC)

A supervised machine learning approach for classification tasks is the SVC algorithm. This Support Vector Machine (SVM) method looks for the ideal hyperplane in a high-dimensional space to split different classes of data points. By transforming the input data into a higher-dimensional feature space, the SVC method determines the hyperplane that maximizes the margin between many classes. It can handle data that can be separated linearly and non-linearly by applying a variety of kernel functions to the input.

6.1.2 Gradient Boosting Regressor

Gradient Boosting Regressor is a supervised machine learning algorithm that is popular due to its high accuracy and robustness, capacity to handle complex, non-linear relationships in the data, and effectiveness in predicting continuous numerical values. The method begins by fitting a straightforward model to the data, then iteratively enhances it by adding new models that concentrate on residuals or errors introduced by earlier models. The updated models are honed to reduce the mistakes produced by the ensemble of models up to this point. This iterative procedure keeps on until either a stopping requirement is satisfied, or a predetermined number of models are produced.

6.1.3 Ridge

The Ridge algorithm was created to reduce the issue of multicollinearity, which occurs when predictor variables have a high degree of correlation with one another. The objective of ordinary least squares (OLS) regression is to minimize the squared residual sum between the predicted and observed values. The predicted coefficients, however, can become unstable and extremely sensitive to slight changes in the data when there is multicollinearity. By including a penalty component to the standard least squares objective function, ridge regression addresses this problem. This fine is inversely proportional to the square of the coefficients' magnitudes. Ridge regression enables the model to disperse the impact of linked predictors more evenly by punishing big coefficient values, leading to more stable and accurate coefficient estimates.

6.1.4 PLS Regression

PLS (Partial Least Squares) regression is a statistical method used for regression modeling, especially when dealing with datasets with high dimensionality and potential multicollinearity among predictors. In PLS regression, the goal is to establish a predictive relationship between a set of predictor variables (X) and a response variable (Y). It achieves this by creating new latent variables, known as components, through a linear combination of the original predictors. These components are designed to capture the maximum covariance between X and Y.

6.1.5 Elastic Net

Elastic Net is a regularization technique used in linear regression and related models and is designed to address the limitations of individual regularization methods, such as Lasso (L1) and Ridge (L2) regression. The objective function consists of two components: the L1 penalty, which encourages sparsity and feature selection, and the L2 penalty, which encourages shrinkage and reduces the impact of multicollinearity. The Elastic Net regularization term is a linear combination of the L1 and L2 penalties, controlled by two hyperparameters: alpha (α) and lambda (λ). The alpha parameter determines the mix between the L1 and L2 penalties. The lambda parameter controls the overall strength of the regularization. A higher lambda value increases the penalty, leading to more shrinkage and feature selection. Selecting an appropriate value for lambda is crucial and often involves techniques such as cross-validation.

6.1.6 Linear Regression

Linear regression is a fundamental statistical modeling technique used to establish a relationship between a dependent variable and one or more independent variables. It assumes a linear relationship between the predictors and the response variable, aiming to find the best-fit line that minimizes the difference between the predicted and actual values. In multiple linear regression, there are multiple independent variables, and the relationship between these variables and the dependent variable is represented by a hyperplane in a higher-dimensional space. The objective is to estimate the coefficients for each independent variable that minimize the sum of squared residuals. Linear regression models can be used for both prediction and inference. They allow us to make predictions on new, unseen data by plugging in values for the independent variables. Additionally, they provide insights into the significance of each predictor variable and the overall goodness-of-fit of the model.

6.1.7 MLP Classifier

MLP (Multilayer Perceptron) Classifier is a type of artificial neural network used for classification tasks in machine learning. It is designed to learn complex patterns and relationships in the data by using multiple hidden layers between the input and output layers. Each neuron in the network receives input from the neurons in the previous layer, applies an activation function to the weighted sum of inputs, and passes the result to the next layer. The weights and biases associated with each neuron are iteratively adjusted during training to minimize the error or loss function. The number of hidden layers and the number of neurons in each layer are hyperparameters that need to be defined before training the MLP Classifier. These hyperparameters impact the model's capacity to learn complex patterns, and selecting the appropriate architecture often requires experimentation and tuning.

6.2 Performance parameters

6.2.1 Accuracy

Accuracy measures the overall correctness of a model by calculating the ratio of correctly predicted instances to the total number of instances. It offers a broad evaluation of the model's effectiveness across all classes. If the dataset is unbalanced, or when some classes have a significantly higher number of occurrences than others, accuracy can be deceptive.

6.2.2 Precision

Precision measures the percentage of correctly predicted positive cases among all instances that were projected to be positive. In order to reduce false positives, it focuses on the quality of positive predictions. When the cost of false positives is large or when we want to guarantee high precision in a certain class, precision is particularly important.

6.2.3 Recall

The proportion of accurately anticipated positive cases out of all actual positive instances is measured by recall, also known as sensitivity or true positive rate. It prioritizes collecting all instances of positivity while reducing false negatives. When the cost of false negatives is large or when we want to guarantee a high recall in a certain class, recall is essential.

6.2.4 F1 Score

A harmonic mean of memory and precision makes up the F1 score. It offers a balanced measurement that takes precision and recall into account concurrently. When we need to take into account both the accuracy of positive predictions and the completeness of collecting positive cases, the F1 score is helpful. Models with unbalanced accuracy and recall levels are penalized.

6.2.5 ROC Score

Performance metrics for binary classification models include the ROC (Receiver Operating Characteristic) score and AUC (Area Under the ROC Curve). With regard to various classification thresholds, it assesses the trade-off between true positive rate (TPR) and false positive rate (FPR). Regardless of the classification threshold selected, the ROC score provides a measure of the model's ability to distinguish between positive and negative instances.

6.3 Results

6.3.1 For 2 allocated resource blocks

When the number of resource blocks allocated for the system is 2, SVC algorithm gives the best results for all the parameters. Accuracy is 99.7 %, Precision is 99.8 %, Recall is 99.7 %, F1 Score is 99.5 % and ROC Score is 99.9 %.

Table 6.1: Performance parameters for 2 resource blocks

Allocated Resource Blocks (RB) = 2					
Algorithm	Accuracy %	Precision %	Recall %	F1 Score %	ROC Score %
SVC	99.70	99.80	99.70	99.50	99.90
Gradient Boosting Regressor	99.50	99.40	99.60	99.50	99.80
Ridge	99.20	99.20	99.40	99.30	99.50
PLS Regression	99.00	99.00	99.10	99.10	99.20
Elastic Net	98.90	99.00	99.00	99.00	99.10
Linear Regression	98.80	98.80	99.00	98.90	99.00
MLP Classifier	98.60	98.50	98.70	98.60	98.80

6.3.2 For 3 allocated resource blocks

When the number of resource blocks is 3, SVC algorithm gives an accuracy of 99.4 %, Precision of 99.4%, Recall of 99.30% F1 score 99.40% and ROC Score of 99.5 %. The other algorithms also give very high accuracy and precision scores.

Table 6.2: Performance parameters for 3 resource blocks

Allocated Resource Blocks (RB) = 3					
Algorithm	Accuracy %	Precision %	Recall %	F1 Score %	ROC Score %
SVC	99.40	99.40	99.30	99.40	99.50
Gradient Boosting Regressor	99.30	99.30	99.20	99.30	99.50
Ridge	98.90	98.60	98.90	98.70	99.20
PLS Regression	98.70	98.80	98.90	98.80	99.10
Elastic Net	98.60	98.70	98.50	98.60	99.30
Linear Regression	98.50	98.80	98.90	98.80	99.10
MLP Classifier	97.90	98.40	98.60	98.50	98.90

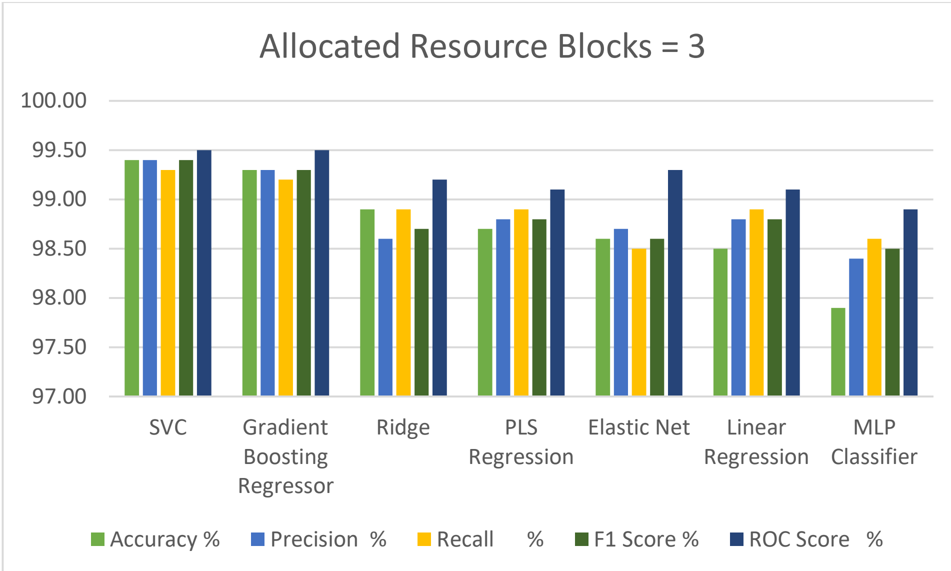


Figure 6.1: For 3 allocated resource blocks

6.3.3 For 4 allocated resource blocks

When no of resource blocks allocated is 4, the performance parameters are all in the range of 98-99 % for all the algorithms. Gradient Boosting Regressor gives the highest values with Accuracy 99 %, Precision 99.1 %, Recall 99 %, F1 Score 99 %, ROC Score 99%.

Table 6.3: Performance parameters for 4 resource blocks

Allocated Resource Blocks (RB) = 4					
Algorithm	Accuracy %	Precision %	Recall %	F1 Score %	ROC Score %
SVC	98.90	98.80	99.00	98.90	99.00
Gradient Boosting Regressor	99.00	99.10	99.00	99.00	99.10
Ridge	98.90	98.90	98.90	98.90	99.00
PLS Regression	98.80	98.80	98.80	98.80	98.80
Elastic Net	98.73	98.75	98.71	98.72	98.73
Linear Regression	98.50	98.42	98.44	98.43	98.45
MLP Classifier	98.45	98.44	98.47	98.45	98.46

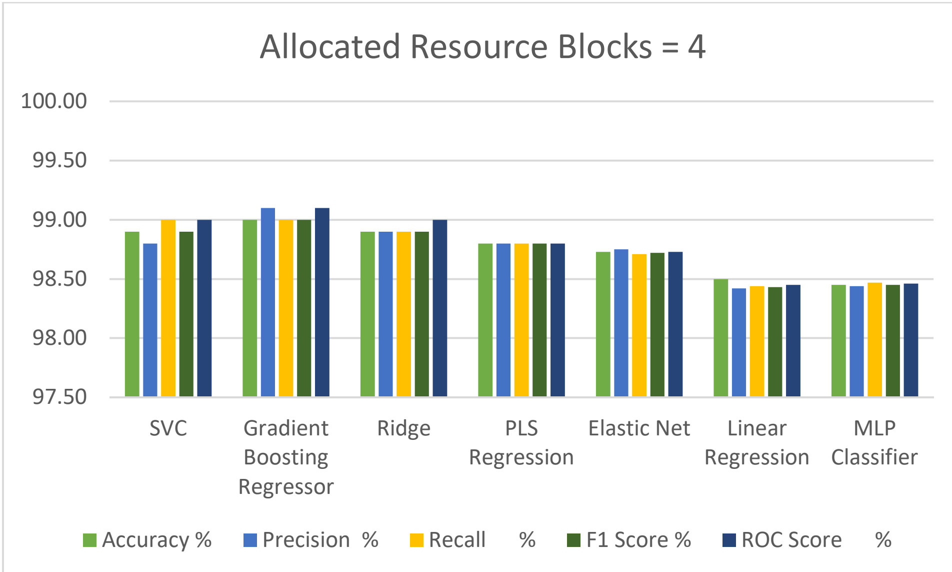


Figure 6.2: For 4 allocated resource blocks

6.3.4 For 5 allocated resource blocks

When the number of resource blocks allocated is 5, the values of performance parameters range from 92-99 %. This shows that with the increase of RBs, there is a slight decrease in the performance of the system. SVC gives highest accuracy of 98.10 % and precision of 98.50%. Gradient Regressor Boosting algorithm gives 97.80 % for Recall which is the highest, and 99.80 % for ROC Score.

Table 6.4: Performance parameters for 5 resource blocks

Allocated Resource Blocks (RB) = 5					
Algorithm	Accuracy %	Precision %	Recall %	F1 Score %	ROC Score %
SVC	98.10	98.50	97.50	98.00	99.20
Gradient Boosting Regressor	97.80	97.90	97.80	97.80	99.80
Ridge	97.50	97.50	97.50	97.50	99.50
PLS Regression	96.80	96.70	96.90	96.80	99.50
Elastic Net	95.60	95.80	95.70	95.70	97.90
Linear Regression	94.70	94.90	94.60	94.70	98.50
MLP Classifier	93.70	94.50	92.30	93.40	97.70

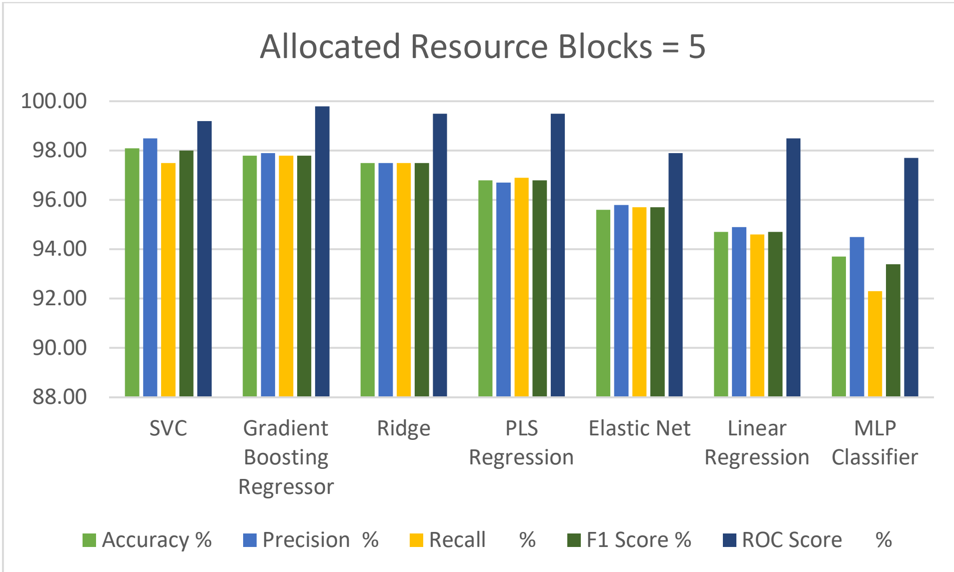


Figure 6.3: For 5 allocated resource blocks

6.3.5 Decrease in Latency

We simulated the model of [1] in MATLAB and determined the time required for the computation. It was 0.15 seconds. But for the ML- based model, the simulation time was 0.09 pico-seconds. This means that our proposed model makes the process of MCS selection much faster than the model of [1].

6.4 Discussion

MCS, CQI, code rate, and rate (bit/symbol) are predicted by our model. When fewer resource blocks are allocated (RB = 2), our model's accuracy is very high. The accuracy decreases marginally as the number of RBs rises. SVC method with two RBs was found to have the maximum accuracy. High precision and recall show that our model efficiently recognizes affirmative cases.

The MCS selection time is 0.09 ps, which is a very good result for the ML-based model in terms of reducing latency. In conditions of high mobility and interference, it functions better. As a result, the system's throughput is increased.

Chapter 7

Conclusion

7.1 Challenges

Throughout the course of this thesis, a number of obstacles had to be addressed in order to improve throughput in downlink Long-Term Evolution (LTE) networks through the use of machine learning-based modulation and coding scheme (MCS) selection.

The first major difficulty was obtaining a large-scale, diversified dataset for training the machine learning algorithms. It took a lot of planning and effort to gather real-world data that covers different network circumstances, user scenarios, and interference levels.

Second, modeling and precisely forecasting channel conditions proved more challenging due to the dynamic nature of the wireless channel. The creation of resilient and adaptive machine learning (ML) algorithms that could cope with shifting surroundings in real time was required to cope with channel fading, interference, and multipath effects.

Thirdly, it was difficult to choose the most effective ML algorithm and optimize its parameters. Finding the ideal balance was essential for increasing throughput while retaining computing efficiency because each method displayed various trade-offs in terms of complexity, accuracy, and generalization capabilities.

It was challenging to ensure that the ML-based MCS selection could be used in real-world LTE systems. It took extensive thought and validation to integrate the suggested technique into the current infrastructure while taking into account resource limitations, latency requirements, and compliance with legacy standards.

Last but not least, it was difficult to fairly and thoroughly evaluate the performance of the suggested technique to that of the currently used MCS selection strategies. It required careful preparation and execution for developing rigorous simulation tests that replicate real-world network conditions and efficiently account for different performance measures.

Despite these difficulties, the thesis effectively surmounted each one, offering insightful information about the use of machine learning for throughput increase in LTE networks and opening the door for further development of wireless communication systems.

7.2 Our contribution

Our thesis provides several important advances to the field that boost throughput in downlink Long-Term Evolution (LTE) networks by machine learning-based modulation and coding scheme (MCS) selection. What follows are the main contributions:

1. Novel ML-Based Approach: For intelligent MCS selection, we provide a novel approach that makes use of machine learning algorithms. Our method provides real-time decision-making for optimized throughput by utilizing historical channel status data and system characteristics.

2.Comparative Evaluation: We thoroughly assess and compare several machine learning algorithms in terms of complexity and throughput performance, including decision trees, support vector machines, and deep neural networks. This evaluation offers insightful information regarding the applicability of various ML approaches for MCS selection in LTE networks.

3.Realistic Simulation Framework: We develop and put into practice a thorough simulation framework that faithfully reproduces real-world network scenarios. This system takes into account user portability, interference effects, dynamic channel conditions, and network topology, enabling complete performance analysis of the suggested ML-based MCS selection technique.

4. Performance Validation: We demonstrate that our ML-based technique outperforms conventional MCS selection strategies through extensive simulations. Our findings provide significant throughput gains, demonstrating the importance of machine learning for enhancing downlink performance in LTE networks.

5. Implementing the ML-based MCS selection strategy in LTE systems: We examine the practical viability of doing so. We offer insights into the potential integration of our technique into actual installations while taking into account resource limitations, latency requirements, and compatibility with legacy standards.

Overall, by offering a fresh perspective, completing an exhaustive analysis, and showcasing the practical viability of machine learning-based MCS selection for increased throughput in downlink LTE networks, this thesis advances to the body of current knowledge.

7.3 Future Scope

The successful implementation and assessment of machine learning-based modulation and coding scheme (MCS) selection in downlink Long-Term Evolution (LTE) networks available the way for an assortment of potential future research directions. The following is the future scope of this work:

1. Next-Generation Wireless Systems: The insights gleaned from this thesis can be applied to advanced wireless communication standards, including 5G. In light of their particular requirements and characteristics, additional investigation can examine the usefulness of machine learning techniques in optimizing MCS selection in these sophisticated systems.

2. Reinforcement Learning: Additional studies may explore the application of reinforcement learning strategies for MCS selection, even though this thesis focuses on supervised learning procedures. Adaptive decision-making based on feedback from the environment is made possible by reinforcement learning, which could end up in MCS selection strategies that are more efficient still.

3.Hybrid Approaches: By combining machine training with conventional analytical techniques, better MCS selection solutions can be provided. It is feasible to investigate hybrid techniques that combine the best aspects of each, improving performance and lowering computing complexity.

4. Dynamic Channel Prediction: By looking into cutting-edge methods for channel prediction, MCS selection may be made more accurately and efficiently. To consistently forecast channel conditions in real-time, deep learning-based models or hybrid approaches including physical layer insights might be researched.

5. Beneficial Deployments: Upcoming studies may concentrate on the useful use of MCS selection based on machine learning in live LTE networks. The sustainability, scalability, and performance of the suggested solution in realistic network contexts can be better understood through field trials and actual deployments.

In conclusion, the future scope of this work encompasses improvements to next-generation wireless systems, studies on hybrid and reinforcement learning methodologies, enhancements to channel prediction methods, and real-world deployments to validate the suggested strategy.

Reference

1.Fan et al. - 2011 Proceedings of 20th International Conference on Computer Communications and Networks (ICCCN) - 2011

2.Ji et al. - 2012 Second International Conference on Business Computing and Global Informatization - 2012

3.Ding et al. - 2011 7th International Conference on Wireless Communications, Networking and Mobile Computing - 2011

4.Fa-tang Chen & Gen-lin Tao - 2010 International Conference on Computer Application and System Modeling (ICCASM 2010) - 2010

5.Li et al. - The 2014 5th International Conference on Game Theory for Networks - 2014