PREDICTING UPLIFT CAPACITY OF SQUARE ANCHOR PLATE USING MACHINE LEARNING TECHNIQUES

THESIS SUBMITTED IN

PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE AWARD OF THE DEGREE OF

Master of Engineering

in

Soil Mechanics and Foundation Engineering

(Civil Engineering Department)

Jadavpur University

by

SK AJFAR HOSSAIN

Class Roll No: 002110402018

Examination Roll No: M4CIV23004

Registration No: 131294 of 2015-2016

Under the guidance of

Dr. Sumit Kumar Biswas

Associate Professor

Department of Civil Engineering, Jadavpur University

Kolkata - 700032

West Bengal, India

DECLARATION OF ORIGINALITY AND COMPLIANCE OF ACADEMIC ETHICS

I hereby declare that this thesis contains Previous Work and original work by the undersigned candidate, as part of my Master of Engineering in Soil Mechanics and Foundation Engineering in the Department of Civil Engineering. All information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that I have thoroughly cited and referenced all material and findings which are not original to this research, as provided by these rules and conduct.

Name: Sk. Ajfar Hossain.

Exam Roll No: M4CIV23004

Thesis Title: Predicting Uplift Capacity of Square Anchor Plate Using Machine Learning Techniques.

Signature of Candidate

FACULTY OF ENGINEERING & TECHNOLOGY

JADAVPUR UNIVERSITY

CERTIFICATE

This is to certify that the thesis entitled — **"PREDICTING UPLIFT CAPACITY OF SQUARE ANCHOR PLATE USING MACHINE LEARNING TECHNIQUES"** has been carried out by **Sk Ajfar Hossain** bearing Class Roll No: **002110402018**, Examination Roll No.: **M4CIV23004** and Registration No: **131294 of 2015-2016**, under my guidance and supervision and be accepted in partial fulfillment of the requirement for the degree of Master of Engineering in Soil Mechanics and Foundation Engineering in the Department of Civil Engineering.

Dr. Sumit Kumar Biswas

Supervisor

Department of Civil Engineering

Jadavpur University

Prof. Partha Bhattacharya

Head of the Department

of Civil Engineering

Jadavpur University

Prof. Ardhendu Ghoshal

Dean

Faculty Council of Engineering

Jadavpur University

FACULTY OF ENGINEERING & TECHNOLOGY

JADAVPUR UNIVERSITY

CERTIFICATE OF APPROVAL

The forgoing thesis titled **"PREDICTING UPLIFT CAPACITY OF SQUARE ANCHOR PLATE USING MACHINE LEARNING TECHNIQUES"** is hereby approved as a creditworthy study of an engineering subject conducted and presented satisfactorily to warrant its acceptance as a precondition to the degree for which it was submitted. It is understood that the undersigned does not automatically support or accept any argument made, opinion expressed, or inference drawn in it by this approval, but only approves the thesis for the reason it was submitted.

Committee on Final Examination

for Evaluation of the Thesis

_____

Signature of External Examiner

_____

Signature of Supervisor

ACKNOWLEDGEMENTS

This thesis entitled **"PREDICTING UPLIFT CAPACITY OF SQUARE ANCHOR PLATE USING MACHINE LEARNING TECHNIQUES"** is the result of the work whereby I have been accompanied and supported by many people, including my guide, my friends, and lab seniors. It is a pleasant aspect that now I can express my gratitude to all of them.

First and foremost, I would like to express my sincere gratitude to my thesis supervisor **Dr. Sumit Kumar Biswas**, Associate Professor of the Department of Civil Engineering, at Jadavpur University for his valuable guidance, insightful suggestions, and support while conducting this thesis work as well as in the writing of this thesis. I have been very fortunate to have a guide like him. His positivity, confidence, and ideas helped me to complete my thesis work successfully and he guided me as a guardian. I am also very grateful to our HOD and all the faculties of Civil Engineering department for their continuous help and support and for letting me use all the available resources for my thesis work.

I would like to acknowledge the help of Retired **Prof. Sibapriya Mukherjee**, for helping me with the idea, implementation, and analysis. I am also thankful to my fellow project mates, friends, and technical and non-technical staff of Jadavpur University who have helped me directly or indirectly during the tenure of my thesis work.

I want to express my gratitude to my parents and family also, as, without their sacrifices, I can't do anything. And their invaluable love, encouragement, and support make me, whatever I am today.

Sk. Ajfar Hossain.

Soil Mechanics and Foundation Engineering

Department of Civil Engineering

Jadavpur University

Kolkata-32, West Bengal, India

ABSTRACT

Regarding analysis and design, the way anchors react to uplift forces is quite significant. Engineers require correct knowledge of the uplift capacity of anchors, a crucial characteristic for assessing the uplift force, in order to appropriately assess the stability of anchors. However, because field and laboratory studies are intricate and time-consuming, quantifying uplift force is typically difficult. This paper uses numerical test data from a research study conducted at the Civil Engineering Department at Jadavpur University to predict the uplift capacity using machine learning (ML) techniques. To forecast the uplift capacity, the study used four machine learning (ML) algorithms: Simple Linear Regression (SLR), Random Forests (RFs), Stochastic Gradient (SGD), and eXtreme Gradient Boosting (XGBoost). These ML models were tested for effectiveness and generalizability using the remaining 20% of the datasets after they had been trained on 80% of them. Through three methods, including Bayesian optimisation, random search CV, and grid search CV with k-fold cross-validation, the hyperparameters for each ML model were adjusted. Five alternative metrics, including R2 score, mean absolute error (MAE), mean squared error (MSE), maximum error (ME), and mean absolute percentage error (MAPE), were used to assess each ML model's performance. The results demonstrated that the XGBoost model consistently performed well across all metrics. It achieved high accuracy and the lowest level of errors, indicating superior accuracy and precision in predicting uplift capacity. The RF model exhibited average performance, with slightly higher error metrics compared with the XGBoost model. However, the Linear Regressor and SGD model performed poorly, with very higher error rates and uncertainty in predicting uplift capacity. Based on these results, we can conclude that the XGBoost model is highly effective at accurately predicting uplift capacity of anchors using the data with minimal input features.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER-1

INTRODUCTION

1.1. GENERAL:

Many structures experiences overturning moments due to lateral loads which results in a combination of tension and compression responses at foundation level. The design of some structures needs to foundation systems to resist uplift forces. In these conditions, an effective and safety design method can achieve through the use of tension elements that these elements are referred to ground anchors. This element is typically fixed to the structure and embedded in the ground to effective depth so that they can resist uplifting loads. Soil anchors typically used to resist such uplift loads, although they also provided as a measure to increase the soil stabilization. This system used for retaining wall, transmission towers, foundations, sea walls, pipelines.

The design of many structures needs to foundation systems to resist vertical or horizontal uplift loads. As part of a larger effort to improve the performance of foundation systems the development of guidelines for anchor system design and installation (as seen in Fig.1.1). The different structures like transmission towers, tunnels, sea walls (FIg.1.2), buried pipelines (Fig. 1.3) and retaining wall are subjected to considerable uplift forces. In such cases, an absorbing and economic design solution may be obtained through the use of tension members. These elements, which are related to as anchors, are generally fixed to the structure and embedded in the ground to effective depth so that they can resist uplifting forces, will safety. The anchors are a thin foundation system designed and constructed specifically to resist any uplift force or overturning moment placed on a structure. Generally, anchors are used to transmit different forces from a structure to the soil. Their strength is obtained through the shear strength and dead weight of the surrounding soil. The different types of anchors used in geotechnical engineering and anchors are including: 1. Grout system, 2. Helical system (Fig. 1.4), 3. Plate system, 4. Soil hook system.

A research study on uplift capacity of square anchor plate in Jadavpur University has been carried out. In this research, both experimental and numerical tests have been done. And those tests provide continuous and reliable anchor and soil data, making it an efficient and cost-effective method in geotechnical engineering practice. This data has given an opportunity to further improve the prediction accuracy of uplift load employing machine learning (ML) algorithms. ML algorithms have shown great promise in accurately predicting uplift load from numerical results data. The ML algorithms can learn complex relationships between input variables (e.g., plate size, inclination angle, embedment ratio etc) and output variables (e.g., uplift load of anchor) from large datasets without the need for explicit mathematical models. Many ML algorithms, such as gradient boosting, random forest, support vector machine (SVM) artificial neural network (ANN), and decision trees (DT), have been used in various geotechnical applications, including soil classification, liquefaction analysis, stability analysis, and settlement prediction. The application of ML algorithms in geotechnical engineering has shown promising results in terms of efficiency and accuracy

1.2. The significance of accurately predicting anchor plate uplift capacity for ensuring safety and stability:

Accurately predicting anchor plate uplift capacity is of paramount importance for ensuring the structural safety and stability of various construction projects. Anchor plates serve as critical connections that secure structural elements to the foundation or supporting structures, and their performance directly impacts the overall integrity and reliability of the entire structure with the soil.

When distributing and transferring loads from the superstructure to the foundation or supporting elements, anchor plates are crucial. Accurate uplift capacity forecasts help maintain the structural integrity of the entire building or infrastructure by ensuring that the anchor system can successfully bear the applied loads without failing.

The anchor connections may not hold if the uplift strength of the anchor plates is underrated, which could cause localised deformations, structural instability, or even a catastrophic collapse. Accurate forecasts help avoid such occurrences, protecting finances, property, and lives.

Engineers can optimise the design of anchor systems by making accurate uplift capacity estimations. With the selection of the proper anchor types, sizes, and combinations ensured by this optimisation, cost-effective and efficient structural solutions are produced without sacrificing safety.

Structures are exposed to various external forces, such as wind, seismic events, and dynamic loads. The structure's resilience and capacity to handle unforeseen events are increased by accurate uplift capacity forecasts that guarantee anchor plates can bear these forces.

Regulatory standards and building codes frequently outline the absolute minimum levels of safety for anchor plates. Accurate projections guarantee that the design complies with these requirements, adding another level of safety assurance.

Accurate forecasts help in determining the condition of existing anchor-plate constructions. Engineers can anticipate problems early on by tracking the actual uplift capacity and comparing it to estimates. They can then develop effective maintenance or repair plans.

Overestimating uplift capacity could result in the wasteful use of resources, higher building costs, and negative environmental effects. Proper material utilisation while preserving structural safety is ensured by accurate projections.

To sum up, it is critical to precisely anticipate anchor plate uplift capability in order to guarantee the structural security, steadiness, and dependability of structures and infrastructure. It enables engineers to make wise choices throughout the design, building, and maintenance phases, lowering the likelihood of failures and guaranteeing the long-term performance of structures in a variety of difficult situations. To design safe and resilient built environments that can last the test of time, these forecasts must be accurate and reliable.

1.3. The limitations of traditional methods and emphasize the potential of machine learning in improving predictions:

The following summary emphasises the shortcomings of conventional approaches and the potential of machine learning for enhancing predictions:

Following are some limitations of traditional methods:

1. Simplified Assumptions: The complexity of anchor plate behaviour under various loading circumstances and material qualities may not be well captured by traditional approaches, which frequently rely on simple assumptions and empirical formulas.

2. Limited Scope: Conventional techniques are frequently only suitable to particular anchor types and loading conditions, which limits their adaptability to a variety of engineering applications.

3. Linear Relationships: Many traditional methods assume linear relationships between input parameters and uplift capacity, overlooking potential non-linear behavior that could exist in certain anchor configurations.

4. Empirical Data Reliance: Traditional methods heavily depend on historical empirical data, which may not cover all possible design variations, especially for innovative anchor types or unique structural applications.

5. Lack of Sensitivity Analysis: Conventional methods often lack the ability to perform sensitivity analysis on input parameters, neglecting the relative importance of different factors in the prediction process.

6. Inability to Adapt: As construction practices and materials evolve, traditional methods might struggle to adapt and incorporate new anchor technologies, potentially leading to less accurate predictions.

7. Time and Cost-Intensive Testing: Physical testing to validate traditional methods can be time-consuming, expensive, and may not cover the entire range of design scenarios.

Emphasis on the Potential of Machine Learning:

Following are some points of the potential of machine learning:

1. Data-Driven Approach: Machine learning adopt a data-driven approach, leveraging large datasets to identify patterns and relationships that might not be evident in traditional methods.

2. Capturing Non-Linear Behavior: These techniques can capture non-linear relationships between input parameters and uplift capacity, allowing for more accurate modeling of anchor plate behavior under different conditions.

3. Generalization: Machine learning can generalize from training data to make predictions for unseen anchor configurations and loading scenarios, enhancing their applicability and versatility.

4. Sensitivity Analysis: These approaches enable sensitivity analysis, providing insights into the impact of individual input parameters on uplift capacity, aiding in optimizing anchor designs.

5. Adaptability: Machine learning can easily adapt to incorporate new anchor types or changes in construction practices, ensuring up-to-date and accurate predictions.

6. Efficient Prediction: Once trained, the predictive models are computationally efficient and can provide quick estimates of anchor plate uplift capacity, reducing time and cost compared to physical testing.

7. Continuous Improvement: The models can be continuously updated and improved as new data becomes available, enhancing their predictive performance and adaptability to changing conditions.

8. Integration with Design Tools: Machine learning can be integrated into existing design software, allowing engineers to seamlessly incorporate uplift capacity predictions into the design process.

9. Enhanced Accuracy: By capturing complex relationships and patterns in the data, these techniques offer the potential for improved accuracy in predicting anchor plate behavior, leading to safer and more reliable structural designs.

In conclusion, by offering data-driven, precise, and flexible forecasts of anchor plate uplift capability, machine learning and artificial neural networks have the potential to overcome the drawbacks of conventional approaches. These cutting-edge methods provide a road forward for designing anchor plates that are more effective, affordable, and resilient, ultimately improving structural safety and stability in a variety of engineering applications.

1.4. OBJECTIVES:

The goal of the thesis is to advance the design of anchor plates by utilising machine learning to precisely anticipate uplift capacity using the data obtained from the research work carried out in Jadavpur University. The goals are to increase structural safety, promote creative design methods, and develop knowledge of anchor plate behaviour in various engineering situations by a thorough analysis.

Several objectives were drawn to attain the goal:

- Converting data into an appropriate form using various preprocessing techniques for the implementation of Machine Learning algorithms.
- Finding critical features that will most influence uplift load.
- To determine the appropriate Machine Learning algorithm for predicting uplift capacity of anchor plates.
- Hyperparameter tuning of those Machine Learning algorithm to improve the performance of the model.
- Selecting various metrics to compare the performance of the applied Machine Learning algorithms.

1.5. SCOPE OF THE WORK:

The purpose of the work is to employ machine learning to create precise predictive models for the uplift capacity of anchor plates. To reach to the objectives, progress has been made using the following steps:

1) The dataset which has been used in this work has total six input features and the target feature is the uplift load.

2) This dataset has been split into two parts for the training and testing purpose (80:20 ratio).

3) Then various machine learning algorithms have been used to trained model using the training dataset and after that with the test dataset these models predict the output result. Total four algorithms which are Simple Linear Regressor, Random Forest Regressor, SGD Regressor and XGBoost Regressor have been used to predict the results using different accuracy metrics like R2 score, mean absolute error (MAE), mean squared error (MSE), maximum error (ME) and mean absolute percentage error (MAPE).

4) At last, all the four results have been compared and the best model has been chosen.

The results of the study should help make anchor plate designs safer and more dependable and encourage geotechnical engineers to employ more sophisticated modelling methods.

1.6. ORGANISATION OF THE THESIS:

This thesis is organized in to total of 5 chapters, as follows:
Chapter 1: Introduction

- General.
- The significance of accurately predicting anchor plate uplift capacity for ensuring safety and stability.
- The limitations of traditional methods and emphasize the potential of machine learning in improving predictions.
- Objectives of the thesis.
- Scope of the work.

Chapter 2: Literature Review:

- Comprehensive literature review of anchor plate behaviour, factors influencing uplift capacity, and conventional estimation techniques.
- Review relevant studies on the application of machine learning algorithms and artificial neural networks in geotechnical engineering purposes.
- Motivation behind the work.

Chapter 3: Methodology:

- Overview of the dataset.
- Experimental environment in which the work has been done. This gives a brief concept of machine learning, python (language which has been used) and its different libraries.
- Machine learning algorithms which have been used in this work.
- Selection of performance metrices.
- Explanation of feature selection.
- Brief discussion on data correlation method.

- Importance of features.
- Description of preprocessing steps applied to the datasets.
- Hyperparameter tunning and its different types and the application of hyperparameter tunning on machine learning algorithms used in this thesis.
- Explanation of K-fold validation.
- Specification of performance metrics used for evaluation.

Chapter 4: Results:

- Presentation of results obtained from linear regression (with the help of box plot) using performance metrics.
- Presentation of results obtained from random forest regression (with the help of box plot) using performance metrics.
- Presentation of results obtained from SGD regression (with the help of box plot) using performance metrics.
- Presentation of results obtained from XGBoost regression (with the help of box plot) using performance metrics.

Chapter 5: Analysis and Discussion:

- Explanation of the performance of the four algorithms used in this thesis.
- Comparative analysis of Performance Metrics with the help of bar chart.
- Comparison of outcomes among different models.
- Conclusions.

CHAPTER-2

LITERATURE REVIEW

In this chapter, a review of available literature relevant to this research are to be furnished, overview of previous research works done in the concerned areas of anchor plate behavior, factors influencing uplift capacity, conventional estimation techniques and the application of machine learning algorithms and artificial neural networks in geotechnical engineering purposes are presented. The review has been presented for different methodologies and in chronological order.

2.1. Comprehensive literature review of anchor plate behavior, factors influencing uplift capacity, and conventional estimation techniques:

Ilamparuthi et al. (2002) proposed that the uplift capacity of circular anchors is governed by their diameter, embedment ratio, and sand density. Two modes of failure develop within the soil mass depending on the anchor embedment ratio. The surface anchoring behaviour is characterized by a raised trunk of a soil cone extending from the anchor to the sand surface, with sloping sides approximately φ / 2 to the vertical, regardless of the density of the sand. The behaviour of the deep anchorage is characterized by a rupture zone in the form of a balloon in the mass of the ground on the anchor. The flat part of this rupture surface emerges from the upper edge of the anchor and is inclined at 0.8φ with respect to the vertical, it is also independent of the density of the sand. A three-phase behaviour that characterizes the superficial case and a behaviour of two phases the deep case

Merifield et al. (2003) estimated the ultimate uplift capacity of different shapes of anchor in clay using a new three-dimensional numerical procedure based on finite element formulation of the lower bound analysis theorem. From the analysis, an estimate of the lower limit of anchor failure factor (N_c) was obtained for square, circular and rectangular anchors, as shown in Fig. 2.1(a,b&c). The estimated capacities were encouraging compared to the results of published and available laboratory tests. It was found that the anchoring capacity of strip anchor increased when overburden pressure reached a limiting value reflecting the change from shallow to deep anchoring behaviour. In addition, according to them, at a given depth of anchorage, an anchor may behave as shallow or deep, depending on the dimensionless overload ratio H/C_u. From their analysis, simple parametric equations for avoidance factors, as indicated below, were suggested to determine the capacity of square and circular anchors ina homogeneous soil profile for different anchoring depths.

$$N_{co} = S\,[2.56 \ln (2\,H/B)]$$

Where, N_{co}= break out factor
S= shape factor for square or circular anchor
H/B= embedment ratio

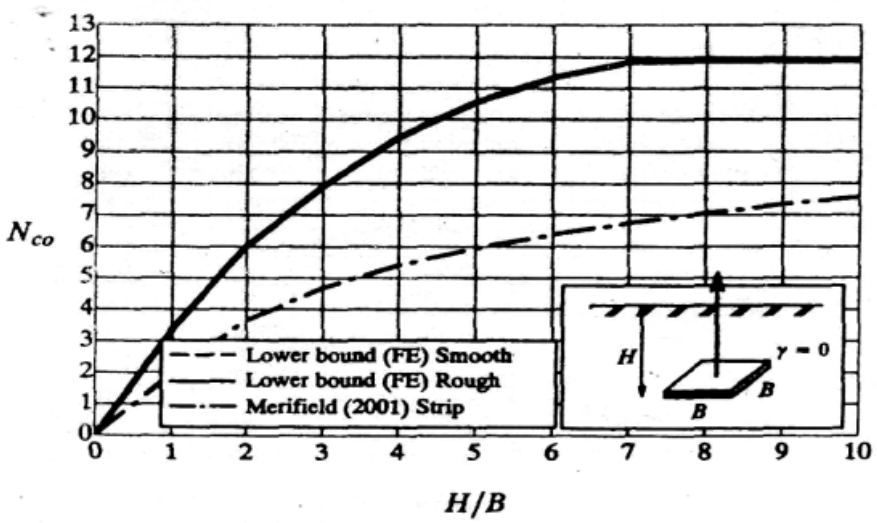

Fig. 2.1(a): Breakout factor for square anchor in clay after (Merifield et al. 2003)

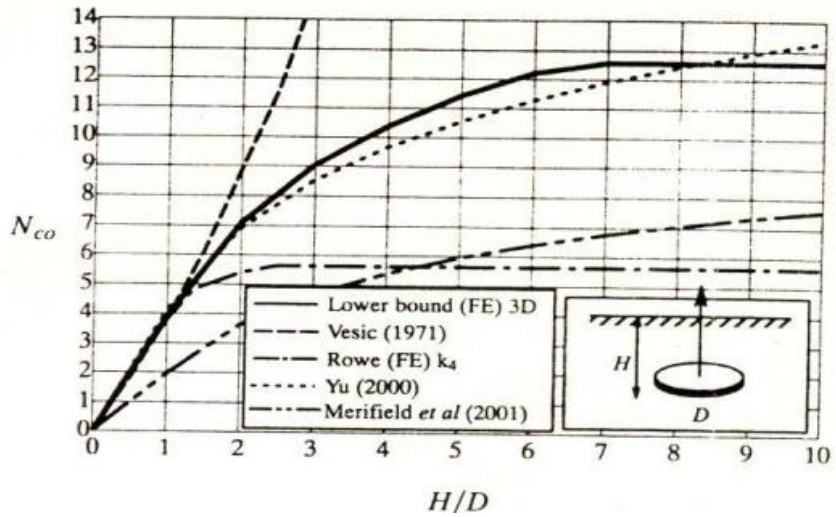

Fig. 2.1(b): Breakout factor for square anchor in clay (after Merifield et al. 2003)

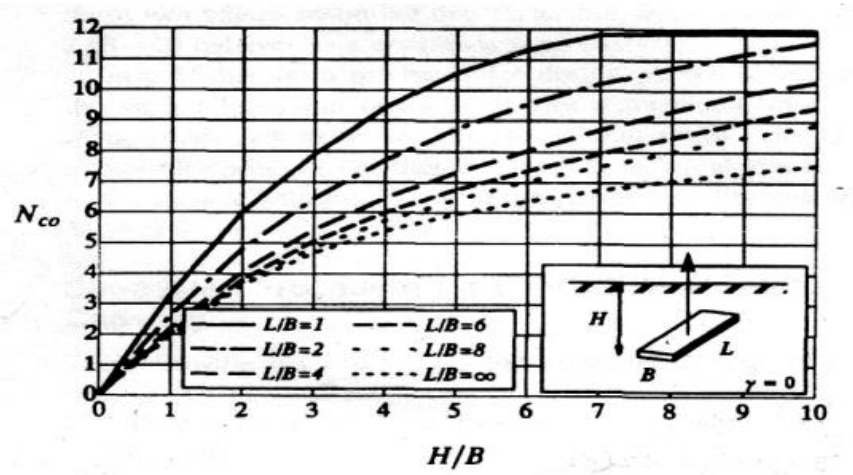

Fig. 2.1(c): Breakout factor for square anchor in clay (after Merifield et al. 2003)

Thorne et al. (2004) studied the uplift behaviour of horizontal strip anchors in clay under fast loading. The possible failure mechanisms were reviewed, including failure due to shear and traction in the soil and the development of suction in the porous fluid. The analysis was made using the finite element program AFENA (Carter and Ballaam, 1995) of the problem of large deformation and an assumption of progressive displacement. Based on their findings the following conclusions were drawn:

- The behaviour of the strip anchors in the uplift capacity are functions of the nondimensional parameters H/B, ϒH/C, U_C/C, where H is the embedment depth, C is the cut resistance without drainage, U_C is the magnitude of the maximum tensile stress of the pore water in the soil and ϒ and B are the unit weight and width of plate.
- Shallow anchors in relatively strong soil tend to fail due to the development of a tensile failure in the soil that is above the anchor and the ultimate capacity is a function of the undrained shear strength of the soil, its own weight and the tensile capacity of the porous fluid

The failure mechanism of the deep anchors where the initial vertical total stress in the plate exceeded seven times the resistance without draining involved only one cut fault located around the anchor. The ultimate capacity in such a case becomes a function only of the resistance without draining the soil.

Goel et al. (2005) worked out the breakout resistance of inclined plate anchors in sand using limit equilibrium approach. The breakout resistance was calculated for different soil friction angles with varying relative depth ratio and anchor inclination. It was found that the breakout factor increased continuously with the inclination of the anchor. A comparison of the predicted values of breakout resistance from the proposed analysis with the experimental values of the other researchers showed reasonably good agreement. The proposed breakout factor was:

$$N_q = \frac{4D}{\pi B} K tan\Phi I_i Sec^2 i$$

where K = coefficient of earth pressure, I_i = coefficient of inclination, i = inclination angle of plate anchor.

Merifield et al. (2005) applied numerical limit analysis and displacement finite-element analysis to evaluate the stability of inclined strip anchors in undrained clay. Consideration was given to the effects of embedment depth and anchor inclination. The ultimate uplift capacity is expressed as:

$$Q_u = AC_u N_c$$

Where $N_c = N_{co\beta} + \frac{\Upsilon H}{Cu}$

$N_{co\beta}$ = breakout factor which takes care of inclination of plate anchor. Breakout factors based on various anchor geometries were presented in the form of charts to facilitate their use in solving practical design problems.

Bhattacharjee et al. (2008) studied behaviour of square plate anchors under uplift load in reinforced clay using a three-dimensional finite element displacement model with ANSYS software. Soil anchor System was discretized with eight nodewise-parametric brick elements for the soil and four nodewise-parametric shell elements for the plate.

Geotextile used as reinforcing material was modelled with two nodded spar element activating tension only. Nonlinear soil behaviours were considered with Drucker-Prager model and the geometrical nonlinearity of geotextile was also addressed in the analysis. They found that, compared to unreinforced clay, the ultimate uplift capacity was more in reinforced clay

with less ultimate displacement. The uplift capacity was found to be dependent on embedment ratio and the position of the geotextile with respect to the embedment depth. The uplift capacity was found to increase with increase in embedment ratio but it decreased with the increase in height of placement of geotextile above the plate.

Khatri & Kumar (2009) employed an axisymmetric static limit analysis formulation in combination with finite elements to obtain the vertical uplift resistance of circular plate anchors, embedded horizontally in a clayey stratum whose cohesion increases linearly with depth. The variation of the uplift factor with changes in the embedment ratio was computed for several rates of increases of soil cohesion with depth. It was noted that in all cases, the magnitude of the uplift factor increases continuously with depth up to a certain value of critical embedment ratio, beyond which it becomes essentially constant.

Mistri et al. (2011) presented the analysis of finite elements for the anchoring of plates in homogeneous and non-homogeneous soils using the PLAXIS 3D. In the initial stages, the final uplift capacity in homogeneous clay shows a rapid increase and can become almost constant at great depth. They proposed that such a change in the rate of increase occurs at the depth of transition where the behaviour of the surface anchor changes to the deep anchor. However, such transitional behaviour is not observed markedly in the clay when increasing the shear strength. As the consistency of the soil increases, the variation in the final uplift capacity also increases.

Shahoo and Kumar (2012) investigated the effects of a spacing factor between two vertical anchors. It was revealed that in a maximum spacing between anchors (Scr), the magnitude of group failure load becomes maximum. Also, they revealed that an increase in (γH=Co) contributes to an increase in the magnitude of group failure load. Also, it was revealed that the critical spacing between two vertical anchors is approximately 0.7-1.2 times the height of the anchor plate. In Fig. 2.2, each of the two anchor plates have the height of B and the bottom edge of the lower vertical anchor is located at a depth H from ground surface. Also, Shahoo and Kumar presented the following equation for estimation of the undrained shear strength due to the fact that it varies linearly with the depth (h):

$$c=c_0(1+ m\frac{h}{B})$$

where, C=value of cohesion at a depth of h

C_0=value of cohesion at ground surface

m=nondimensional factor

They assumed that the group of anchors are subjected to horizontal uplift load and the failure occurs in the both of the vertical plates at the same time.

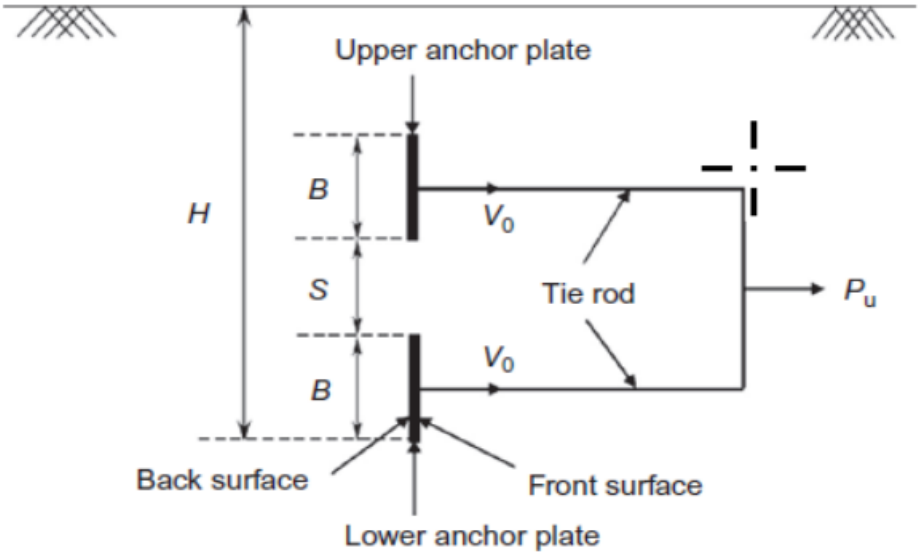

Fig. 2.2 Geometric parameters of two vertical anchors embedded in clay

Niroumand et al. (2013) investigated the failure shape of horizontal anchor plates (circular, rectangular, square, and strip plates) in reinforced sand by geogrid and grid-fixed reinforced (GFR) as an innovative soil reinforcement system that tied the geogrid to soil. Using the soil reinforcement changed the failure mechanism of horizontal anchor plates in the reinforced zone of soil because it makes two failure zones that need to be considered: the first zone is the relation between the plate anchor and soil reinforcement materials; the second zone between soil reinforcement layer and ground surface.

Beirne et al. (2017) investigated field data from reduced scale anchor tests at two sites to validate a new release-to-rest model for dynamically installed anchors. This model considered the motion of the anchor from the point of release in the water column, modelling the drag resistance acting on the anchor and its mooring line. They stated that although dynamically installed anchors were an attractive and often a cost-effective anchoring solution, their global acceptance had been somewhat hampered by uncertainties on their striking the seabed within an acceptable spatial variation, and on achieving the targeted embedment depth in the seabed. The latter was addressed in their research paper through a new release-to-rest model for anchor installation. The model simulates the motion history of the anchor during free-fall in water and dynamic embedment in soil, providing as output the final anchor embedment depth as output to calculate anchor capacity. The importance of considering the motion response in water was demonstrated through model simulations that highlighted the role of drag resistance acting on the trailing mooring line. The observations from these simulations were also reflected in measurements made in field tests. Those field test results were used to validate the model.

Biradar. J et al. (2019) carried out numerical simulations on three different sizes of square anchor plates. A single layer geosynthetic was used as reinforcement in the analysis and placed at three different positions from the plate. The effects of various parameters like embedment ratio, position of reinforcement, width of reinforcement, frequency and loading amplitude on the uplift capacity were presented in the study. The load-displacement behaviour of anchors for various embedment ratios with and without reinforcement was also noted. The

uplift load, corresponding to a displacement equal to each of the considered maximum amplitudes of a given frequency, was presented in terms of a dimensionless breakout factor. The uplift load for all anchors was noticed to increase by more than 100% with embedment ratio varying from 1 to 6. Finally, a semi empirical formulation for breakout factor for square anchors in reinforced soil was proposed by carrying out regression analysis on the data obtained from numerical simulations. In the above section the studies related with analysing the response of horizontal and inclined anchors embedded in different types of soil, were covered. It was noted form the above sections that researchers were studied the response of horizontal and inclined anchors embedded in different types of soil from both experimental as well as numerical approach.

Majumder et al. (2019) investigated the uplift behaviours of plate anchors in geotextile reinforced soft clay by using experimental model set up and numerical model analysis in ABAQUS software. They concluded that the uplift capacity increases with the increase in plate size, embedded depth and inclusion of geotextile as reinforced material.

2.2. Review relevant studies on the application of machine learning algorithms and artificial neural networks in geotechnical engineering purposes:

H. I. Park and C. W. Cho (2009) have applied artificial neural network (ANN) to predict the resistance of driven piles in dynamic load tests. The data was taken from various construction sites in Korea. For piles with appropriate measurements for the tip, shaft, and total pile resistance available, predictions on these values were developed. Using ANN modelling and parametric analysis, it was determined how the key parameters affected the pile resistance values. The findings of this study show that the ANN model may be used as an accurate and straightforward forecasting tool to properly take into account a number of crucial elements for estimating the resistance of driven piles.

P. Samui & T. G. Sitharam (2011) created two machine learning models to forecast the susceptibility of soil to liquefaction using the standard penetration test (SPT) results from the 1999 Chi-Chi, Taiwan earthquake. The first method that use a Levenberg-Marquardt backpropagation algorithm-trained multi-layer perception (MLP)-based artificial neural network (ANN). The Support Vector Machine (SVM), is used in the second machine learning technique to estimate liquefaction susceptibility using corrected SPT [$(N_1)_{60}$] and cyclic stress ratio (CSR). Additionally, an effort has been made to simplify the models so that the prediction of liquefaction susceptibility only requires the two parameters [$(N_1)_{60}$ and peck ground acceleration (a_{max}/g)].

M. H. Baziar et al (2014) has made an ANN model using 1300 recorded settlement data from 101 pile loading tests with the cone penetration test (CPT). The model gives the prediction of pile settlement. In this study the relative importance of input parameters has been evaluated using senility analysis and the accuracy predictions of the proposed model, along with other methods, were compared with the recorded values from the loading tests with the aid of different statistical parameters. This comparison indicated the superiority of the proposed model over previous methods.

Jian Zhou et al. (2015) has compared different classification techniques for pillar stability in hard rock mines. The data has been used in this study was obtained from total 251 pillar cases between 1972 and 2011 and has six features, namely pillar width, pillar height, the ratio of pillar width to its height, uniaxial compressive strength of the rock, pillar strength and pillar stress. The capacity to learn for pillar stability based on various input parameter combinations was assessed for six supervised learning algorithms, including linear discriminant analysis, multinomial logistic regression, multilayer perceptron neural networks, support vector machine (SVM), random forest (RF), and gradient boosting machine. And he found that SVM and RF achieve comparable median classification accuracy rate and Cohen's kappa values.

Shakti Suman, Sarat Kumar Das & Ranajeet Mohanty (2016) have developed models for predicting friction capacity in clay based on experimental test results using two artificial intelligence techniques; multivariate adaptive regression splines (MARS) and functional networks (FN). In terms of statistical parameters like correlation coefficient (R), Nash-Sutcliffe coefficient of efficiency (E), absolute average error, maximum average error, root mean square error, and normalised mean bias error, the effectiveness of developed MARS and FN models has been compared with some previously developed models. Based on statistical results, it is discovered that MARS and FN models are more accurate predictors.

Xiao L. et al. (2018) has tried to use data-driven algorithms to predict landslide susceptibility along the China- Nepal highway based on temporal and spatial sensor data. Ten landslide instability factors were prepared, including elevation, slope angle, slope aspect, plan curvature, vegetation index, built-up index, stream power, lithology, precipitation intensity, and cumulative precipitation index. They have used four machine learning algorithms, namely decision tree (DT), support vector machines (SVM), Back Propagation neural network (BPNN), and Long Short-Term Memory (LSTM) and after final prediction they found that LSTM outperformed the other three models due to its capability to learn time series with long temporal dependencies. It indicates that the dynamic change course of geological and geographic parameters is an important indicator in reflecting landslide susceptibility.

Chen Renpeng et al. (2019) has investigated the efficiency and feasibility of six machine learning (ML) algorithms, namely, back-propagation neural network, wavelet neural network, general regression neural network (GRNN), extreme learning machine, support vector machine and random forest (RF), to predict tunnelling-induced settlement. Models are constructed using field data sets, including geological conditions, shield operational parameters, and tunnel geometry, acquired from four parts of the 3.93 km long tunnel. Each computational model's effectiveness has been shown using the three indicators mean absolute error, root mean absolute error, and coefficient of determination (R2). Comparing the results to the conventional multivariate linear regression method, the results showed that ML algorithms had far greater potential to predict tunnelling-induced settlement. Among six ML algorithms that successfully identify the development of tunnelling-induced settlement, the GRNN and RF algorithms perform the best. The Pearson correlation coefficient was also used to look into the relationship between the input factors and settlement.

Demir, S. & Sahin E.K (2022) has used three relatively recent and reliable tree-based ensemble techniques which are Adaptive Boosting, Gradient Boosting Machine, and eXtreme Gradient Boosting (XGBoost) to predict soil liquefaction from SPT dataset. Recursive Feature

Elimination, Boruta, and Stepwise Regression were used to create models with a high degree of accuracy and minimal complexity for feature selection. To choose the optimal model, the effectiveness of ML algorithms with feature selection methods was compared in terms of four performance metrics: overall accuracy, precision, recall, and F-measure. Fore best prediction model Wilcoxon's sign rank test, a statistical significance test has been used. The study's findings imply that all developed ensemble models based on trees may accurately predict soil liquefaction. From thi study it has been found that the XGBoost with the Boruta model outperformed the other models in all scenarios taken into consideration, according to both validation and statistical data.

Zhang W. et al. (2022) has performed extreme gradient boosting (XGBoost) and random forest regression (RFR) to predict the factor of safety (FOS) against basal heave for deep braced excavation with the numerical results for 1778 hypothetical cases which has been obtained from a finite-element analysis considering the anisotropy for the undrained shear strength was performed to examine the effects of the total stress-based anisotropic model NGI-ADP (develop by Norwegian Geotechnical Institute based on the Active-Direct simple shear-Passive concept) parameters on the base stability of the deep braced excavations in clays. The results indicated that the anisotropic characteristics of soil parameters need to be considered when determining the FS against basal heave for braced excavation. An accurate forecast of the FS can be obtained using XGBoost and RFR.

Yaren Aydin et al. (2023) has used machine learning (ML) for soil classification. The dataset which they have used, consists of 805 soil samples taken during the construction of the new Gayrettepe-Istanbul Airport metro line. To cope with the missing data during the data preprocessing stage, data imputation techniques were first used. After testing two distinct imputation methods, the data were finally imputed using the KNN imputer. Later, using the synthetic minority oversampling approach (SMOTE), a balance was attained. Following preprocessing, 10-fold cross-validation the have used a number of machine learning methods. And they found that new gradient-boosting techniques like XGBoost, LightGBM, and CatBoost gives high classification accuracy rates of up to +90%; and a significant improvement in prediction accuracy (when compared with earlier research) was attained.

2.3. Motivation of Work:

In civil engineering practice, various types of anchors are utilized for offshore and onshore structures to counteract uplift forces. Offshore structures typically use anchors such as drag embedment anchors (DEAs), suction anchors, or gravity anchors, which are designed to provide stability and resist the forces exerted by waves, currents, and wind. On the other hand, onshore structures often utilize anchors such as concrete or steel anchors, helical anchors, or ground anchors, which are designed to resist uplift forces caused by factors such as soil erosion, wind loads, or seismic activity. Thus, it is understood that the response of anchors against uplift forces has great importance in respect of analysing and designing.

Machine learning has a very important role in making models in the prediction of various geotechnical problems now. Machine learning algorithms have the ability to analyse large amounts of data and identify patterns, which can be used to develop predictive models for geotechnical problems. These models can help in predicting outcomes such as soil behaviour, slope stability, settlement, and other geotechnical parameters. By training on historical data and incorporating various input variables, machine learning models can provide valuable insights

and predictions for geotechnical engineers and researchers. This can aid in decision-making, risk assessment, and optimizing design solutions in the field of geotechnical engineering.
As reviewed here it was noted that different machine learning algorithms have been applied mostly for analysis of soil liquefactions, land slide susceptibility, assessing tentative frictional capacity or capacity of piles, evaluating the probable settlement of piles, etc.
Based on the review of the literature relating to the studies subjected to the machine learning algorithms in relation with the assessment of uplift capacity of anchors, it was felt that this sector needs to be addressed.
These algorithms have the potential to enhance the understanding of anchor performance and provide valuable insights. However, it is crucial to address certain aspects in these studies. Firstly, the selection of appropriate datasets and features is essential to ensure accurate and reliable predictions. Additionally, the validation and testing of the machine learning models should be conducted rigorously to assess their performance and generalizability. Furthermore, it is important to consider the limitations and assumptions of the algorithms used, as well as the potential biases that may arise.
By addressing these factors, this can be anticipated that the studies on anchor's uplift capacity assessment using machine learning algorithms will be robust and will contribute effectively to the field.
So, an attempt has been taken in the present study to apply machine learning algorithms for predicting the uplift capacity of anchors and the concerned data relating to this attempt were obtained from the recent studies on the uplift capacity of anchors which were performed in the Civil engineering Department of the Jadavpur University.

CHAPTER 3

METHODOLOGY

In this chapter, the overview of the dataset, some key concepts of machine learning and the details procedure will be discussed to understand the details of the algorithm that have been implemented for the predication of the uplift capacity of the anchor plates. Fig 3.1 showing the flow diagram illustrating of the whole process.

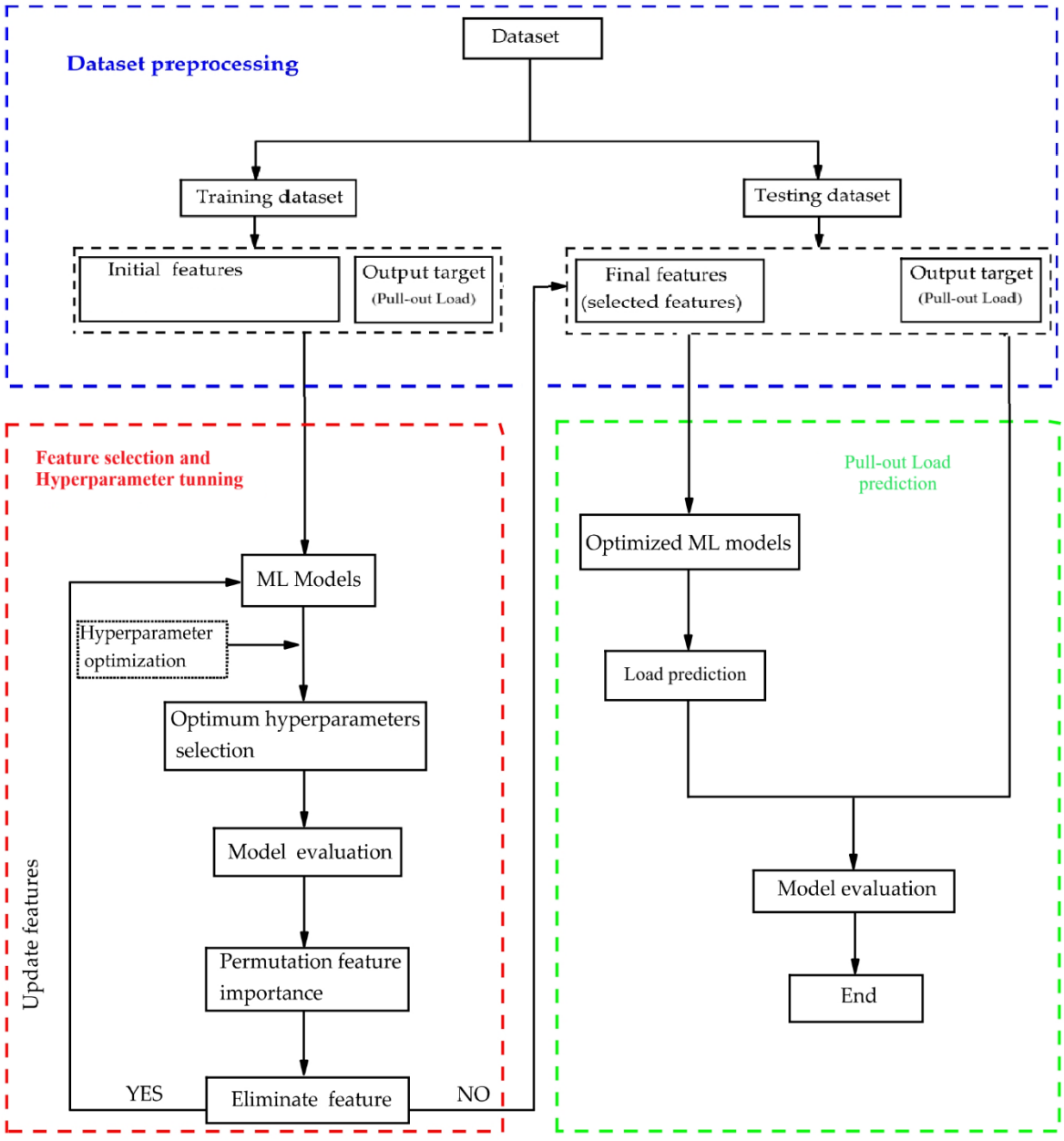

Fig. 3.1: Flow chart of the machine learning process

3.1. DATA SET:

A research study on uplift capacity of square anchor plate in Jadavpur University is ongoing currently and the result dataset obtained from numerical analysis of the determination of uplift capacity of square anchor plates under dynamic loading, is used in this study. The dataset contains 216 results and there are 7 attributes.

```
#Read the excel dataset
dataset=pd.read_excel("Dynamic_Dataset.xlsx")

## print shape of dataset with rows and columns
print(dataset.shape)
```

```
(216, 7)
```

The head of the dataset looks like

```
## print the top5 records
dataset.head()
```

	plate_size	inclination_angle	embedment_ratio	frequency	aplitude	soil_type	load
0	25	30	1	0.2	2	unreinforced	30.10
1	25	30	2	0.2	2	unreinforced	35.91
2	25	30	3	0.2	2	unreinforced	39.13
3	25	45	1	0.2	2	unreinforced	32.20
4	25	45	2	0.2	2	unreinforced	35.97

3.2. Experimentation Environment:

3.2.1. Python:

Python is a commonly used high-level programming language, it was designed by Guido van Rossum which can be easy to interpret and read. Python has specific functionality and is convenient to be used for both quantitative and analytical computational purposes. Data Science Python is popularly used and, as well as being a dynamic and open-source language, is a top choice. Its massive libraries are also used to manipulate the data however for a beginner data analyst they are really simple to learn. The python libraries used in this thesis are briefly described as follows:

NumPy

NumPy is a library that consists of multidimensional array objects and a set of array processing routines. NumPy is used along with SciPy and Matplotlib packages. This combination is used for technical computing. Mathematical and logical operations are performed with the help of NumPy.

Pandas

Pandas is a software library that is designed for manipulating the data and analysis in a python programming language. It is open-source which is released under the BSD license of three clauses. It is based on the NumPy package, and the Data Frame is its main data structure.

Matplotlib

Matplotlib is a module of Python used to plot the attractive graphs. Visual representation in data science is a significant step. One can quickly understand how data is split by using visual representation. There are many libraries to represent the data, but the matplotlib is very widely known and easier to visualize.

SKlearn

Scikit-learn is a free python library. It features multiple clustering classification and regression algorithms including random forests, DBSCAN, k-means, gradient boosting, support vector machines, and gradient boosting which is programmed to interface with the NumPy and SciPy libraries.

Seaborn

Seaborn is an open-source python library that is used for statistical graphics. It offers a data set-oriented API to analyze relationships among different variables, as well as resources to select colour palettes that truly in the data.

3.3. Machine Learning:

The field of study known as machine learning enables machines to learn without being explicitly programmed. When a computer program's performance at tasks in a class of tasks T, as assessed by a performance measure P, improves with experience E, that programme is said to have learned from experience E related to that class of tasks T. Machine learning is a broad term for a programme that uses data analysis and data exploration to manage a variety of tasks.

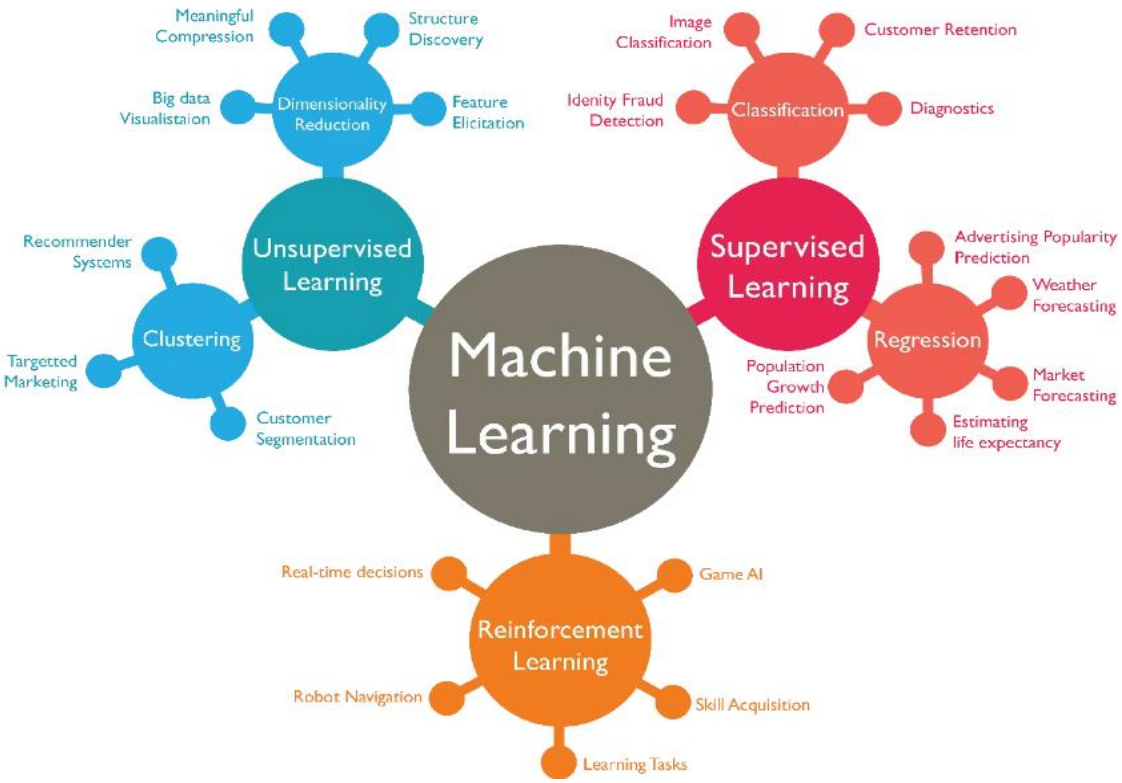

Fig. 3.2: Types of Machine Learning

Applications of machine learning that are widely used include the identification of email spam, the theft of credit cards, stock predictions, personal assistants, product suggestions, self-driving cars, sentiment analysis, etc.

3.3.1. Supervised Learning:

Supervised learning is the model that is used the most often for machine learning operations. When the mapping between input-output data is reliable, it is frequently employed for such types of data. Learning from labelled test data is known as supervised learning, which is a branch of machine learning that focuses on developing models for regression or classification.

3.3.2. Unsupervised Learning:

In the situation of unsupervised learning, where all the data is unlabeled, the classes are not explicitly labelled on the data. The model can learn from the data by spotting implicit patterns. Based on the data, unsupervised learning classifies the densities, structures, connected segments, and other comparable attributes.

3.3.3. Reinforcement Learning:

A branch of machine learning is reinforcement learning. It involves acting appropriately to maximise reward in a certain scenario. To identify the optimal course of action or direction it will take in a certain event, many algorithms and computers are used. The difference between supervised learning and reinforcement learning is that in supervised learning, the answer key is included in the training data, allowing the model to be trained with the correct response from the start, whereas in reinforcement learning, there won't be a response and the reinforcement agent will decide how to carry out the task. In the absence of training data, it is necessary for it to learn from its experience.

3.4. Machine Learning Algorithms:

Forecasting refers to making future predictions, usually using historical data. For making forecasts for a very long time, statistical models were frequently used. The function of generalisation in machine learning has been taken into account. In this thesis we have used supervised learning algorithms such as, Simple Linear Regression, Random Forest Regression, Stochastic Gradient (SGD) Regression and XGBoost Regression. These can make it easier to find better outcomes compared to traditional analytical techniques.

3.4.1. Simple Linear Regression:

Simple linear regression is useful for defining a relationship between two continuous variables. One is an indicator or independent variable and another is an answer or dependent variable. It looks for a statistical relationship, but not a deterministic one. The relationship between the two variables is said to be deterministic if one variable can be precisely represented by the other. For example, it is possible to correctly forecast Fahrenheit by using temperature in degree Celsius. The mathematical equation is not sufficient to assess the association between the two variables. For example, the relationship between weight and height. The Equation for the Simple linear regression is:

$$Y = a + bX$$

where Y is the expected value of the dependent variable y for every specified value of the independent variable X, a is the intercept, b is the regression coefficient and X is independent variable.

3.4.2. Random Forest Regression:

Random Forest is one of the most powerful Machine Learning frameworks for predictive analytics. A random forest method is a type of discrete structure that allows predictions by integrating decisions from a series of simple models. More formally, this subset of models can be written as:

$$g(x) = f0(x) + f1(x) + f2(x) + f3(x) +.....$$

Where the initial configuration g is the number of the initial specific model's fi. Here, any base classifier is a simple decision tree. This wide-ranging technique of using multiple models to improve predictive performance is called model assembling. In random woods, all baseline models are built independently using a separate subset of results.

3.4.3. Stochastic Gradient (SGD) Regression:

The Stochastic Gradient Descent (SGD) Regressor is a fundamental optimization algorithm used for training regression models, especially when dealing with large datasets. It's a variant of the traditional gradient descent algorithm that offers computational efficiency by updating model parameters based on subsets of data rather than the entire dataset.

SGD Regressor updates the model parameters using small random batches of data in each iteration. This results in faster convergence and makes it suitable for handling large datasets that might not fit entirely in memory. The "stochastic" in SGD refers to the randomness introduced by the selection of data batches. This randomness can lead to noisy updates but often helps escape local minima and converge to a reasonably good solution. The algorithm iteratively refines the model parameters by adjusting them in the direction that reduces the loss function. This continues for a fixed number of iterations (epochs) or until a convergence criterion is met. Key hyperparameters include the learning rate, which determines the step size of parameter updates, and the batch size, which controls the number of data points used in each update. The choice of loss function depends on the regression problem. Common choices include mean squared error (MSE) for least squares regression and mean absolute error (MAE) for robust regression. After training, the final model parameters are used to make predictions on new data.

3.4.4. XGBoost Regression:

XGBoost (Extreme Gradient Boosting) Regression is a highly popular and powerful machine learning algorithm used for regression tasks. It's an ensemble learning method that combines the predictions of multiple individual models (typically decision trees) to create a robust and accurate final prediction. XGBoost employs a gradient boosting framework, which iteratively builds decision trees to correct the errors made by the previous trees. Each new tree focuses on the residual errors of the ensemble's current prediction. This technique includes built-in regularization techniques to prevent overfitting. This helps control the complexity of the individual trees and the overall ensemble, enhancing the model's generalization ability. XGBoost provides a mechanism to measure the importance of features in making predictions.

This information is valuable for feature selection, understanding model behavior, and identifying key factors that drive predictions.

This algorithm is designed to optimize a specific loss function, such as mean squared error (MSE) for regression problems. It uses both the gradient and the second derivative of the loss function to guide the tree-building process.

XGBoost offers a wide range of hyperparameters that can be tuned to improve model performance. These include parameters related to the number of trees, their depth, learning rate, regularization terms, and more.

3.5. Selection of Machine Learning Algorithms:

For every problem, choosing an algorithm is not a trivial decision. There is no proper algorithm that works for any problem, but few algorithms are widely recognized for performing the algorithms better than others in some cases. One cannot assume the more accuracy from the algorithms for all types of data, accuracy will differ from data to data. In this thesis Machine Learning Algorithms such as Simple Linear Regression, Random Forest Regression, Stochastic Gradient (SGD) Regression and XGBoost Regression were considered in which they expected to perform well on the issues.

3.6. Selection of Performance Metrics:

To identify the appropriate algorithm, one needs to evaluate the results and then we can predict it. In this case, the R2 score would play a crucial role while measuring the performance of the algorithm. For calculating the average magnitude of errors mean absolute error metric will be used in this study. In real-time data there might be a chance of worst-case error between the actual value and predicted value in this particular scenario max error is used. Mean absolute percentage error is used to keep the positive and negative errors from cancelling one another out and uses relative errors to enable you to compare forecast accuracy between time series models. And also mean squared error is used to measure the average of the square of the errors. That is, the average squared difference between the estimated values and the actual value.

3.7. Feature Selection:

There are various types of factors that can make the model of Machine Learning more effective on any given task. One of the methods of feature selection is data correlation

	plate_size	inclination_angle	embedment_ratio	frequency	aplitude	load
count	216.00000	216.000000	216.000000	216.000000	216.000000	216.000000
mean	50.00000	45.000000	2.000000	0.350000	3.500000	82.398667
std	20.45983	12.275898	0.818393	0.150348	1.503484	42.113175
min	25.00000	30.000000	1.000000	0.200000	2.000000	26.100000
25%	25.00000	30.000000	1.000000	0.200000	2.000000	45.310250
50%	50.00000	45.000000	2.000000	0.350000	3.500000	74.914000
75%	75.00000	60.000000	3.000000	0.500000	5.000000	111.905250
max	75.00000	60.000000	3.000000	0.500000	5.000000	202.645000

Fig. 3.3: Dataset description

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 216 entries, 0 to 215
Data columns (total 7 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   plate_size        216 non-null    int64
 1   inclination_angle 216 non-null    int64
 2   embedment_ratio   216 non-null    int64
 3   frequency         216 non-null    float64
 4   aplitude          216 non-null    int64
 5   soil_type         216 non-null    object
 6   load              216 non-null    float64
dtypes: float64(2), int64(4), object(1)
memory usage: 11.9+ KB
```

Fig. 3.4: Dataset summary

	plate_size	inclination_angle	embedment_ratio	frequency	aplitude	load
plate_size	1.000000e+00	1.031603e-16	7.204114e-17	-3.021644e-17	2.363597e-17	0.396988
inclination_angle	1.031603e-16	1.000000e+00	1.480297e-17	2.327785e-16	1.190752e-16	0.136016
embedment_ratio	7.204114e-17	1.480297e-17	1.000000e+00	3.504268e-16	1.091149e-16	0.163244
frequency	-3.021644e-17	2.327785e-16	3.504268e-16	1.000000e+00	2.935466e-16	-0.124455
aplitude	2.363597e-17	1.190752e-16	1.091149e-16	2.935466e-16	1.000000e+00	0.837622
load	3.969878e-01	1.360157e-01	1.632442e-01	-1.244554e-01	8.376223e-01	1.000000

Fig. 3.5: Correlation Values

which will have a major impact on the model's performance. This will reduce a lot of strain on the Machine Learning model during preprocessing and cleansing the data. The data attributes chosen for training the Machine Learning model would have a major impact on the efficiency of the model. Because of the irrelevant features that are presented, the model output will be reduced. The feature selection method provides an efficient way to remove data redundancy and irrelevant data that helps to reduce computation time, improve accuracy, and also enhance understanding of the model. The selection of features plays a crucial role in classification and involves selecting a subset of features that reflect the complete attributes that currently exist. Feature selection techniques are intended to improve classification efficiency by selecting the essential features from the data sets according to particular algorithms.

3.8. Data Correlation Method:

Data correlation is a method that helps to predict one attribute from another attribute and is used as a basic quantity in many modeling techniques. If one feature increases, the correlation will be positive, so the other feature increases as well and negative if one feature increases there will be a reduction in another. If there is no relation between any two attributes then it is said to be no correlation. If there is a linear relationship between the constant variables then the Pearson correlation coefficient is used. If there is a non-linear relation between the constant variables then the Spearman correlation coefficient is used.

Since the considered data set is linear so the Pearson correlation coefficient is used for the selection of features in this study. This correlation for all the attributes is shown in figure 3.4. To improve the efficiency of the Machine Learning model, the attributes that have negative correlations were removed. It is a statistic measuring the linear correlation of two variables X and Y. It has a value between +1 and 1, where 1 is a linear positive correlation, 0 is not a linear correlation and 1 is a linear negative correlation.

The motivation for considering the correlation is when people know a score on one measure, they can make a prediction of another measure that is highly related to it more accurate. The more accurate the prediction, the stronger the relationship between the variables.

The heat map for correlation between non-numerical attributes is plotted as follows:

```
plt.figure()
sns.heatmap(dataset.corr(),annot=True,fmt=".0%")
```

```
<AxesSubplot: >
```

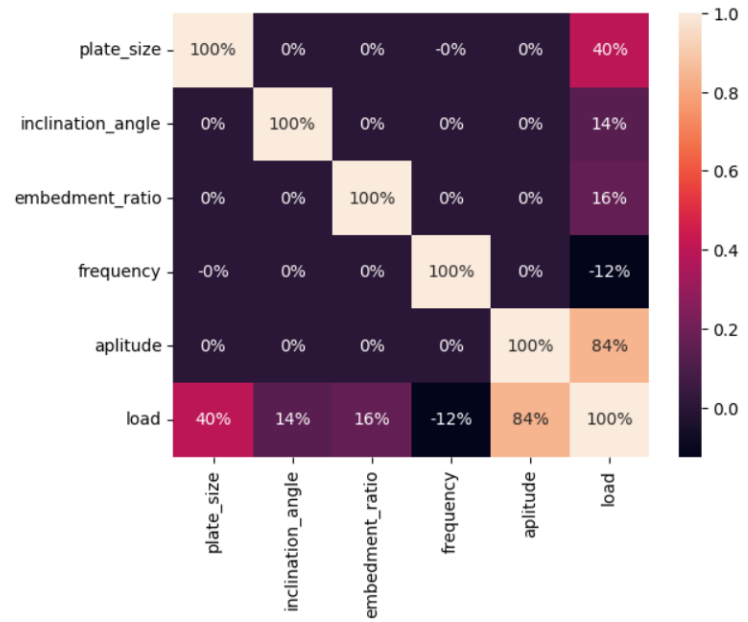

Fig. 3.6: Heat Map

3.9. Feature Importance:

Feature Importance refers to a class of approaches for assigning values to input features to a predictive model which determines the relative significance of each factor while forecasting.

Feature importance scores provide overview into the model. Most significant scores are determined using a prediction approach that was fitted to the dataset. Inspecting the score of importance gives insight into that particular model and what features are the most essential and least important to the model while making a prediction. This is a type of interpretation of the model that can be carried out for those models that encourage it.

Feature Importance can be used to enhance a predictive model. This can be accomplished by selecting those features to remove (lowest scores) or those features to retain, using the importance scores. This is a type of selection of features, and can simplify the modeling problem, accelerate the modeling process, and in certain cases improve model performance.

3.10. Data Preprocessing:

Before applying Machine Learning algorithms some of the missing values have been found which can impact the model's output so this should be handled. To make the dataset more efficient, these missing values will be replaced by the most promising values. In this case there is no missing value. So, this step need not to be performed in this dataset. If there's more

correlation between two of the different attributes with similar work, then removing one of the attributes will make the work better. In this case this step is also need not to be performed. In this dataset there is an attribute "soil_type" which is a categorical variable. Since most machine learning models only accept numerical variables, we had to transform categorical variables into numerical representation such that the model is understand and extract valuable information. This step known as encoding categorical variables.

3.10.1. Encoding Categorical Values:

Categorical data contains label values that are considered nominal values. Each value has categories of different types. Besides, a few of the groups have a normal relationship with each other is known as natural ordering. The categorical data can be converted into numerical data to improve the efficiency of the Machine Learning model.

In Python, there are several commonly used techniques for categorical encoding. Here are four types of categorical encoding methods:

- **One-Hot Encoding**: With this method, each category is represented by a binary column, where a value of 1 denotes the presence of the category and a value of 0 denotes its absence. For one-hot encoding, the pandas package offers the *get_dummies()* function.
- **Label Encoding**: Each category is given a different numerical label through label encoding. Ordinal categorical variables can use it. For label encoding, use the *LabelEncoder* class from the *sklearn.preprocessing* module.
- **Ordinal Encoding**: The ordinal link between categories is maintained by ordinal encoding, which gives each category a numerical value. When categories are arranged in a useful manner, it is suitable. Ordinal encoding often makes use of the *OrdinalEncoder* class from the *sklearn.preprocessing* module.
- **Target Encoding**: Target encoding is a Baysian encoding technique. Target encoding involves calculating the mean of the target variable for each category and substituting the mean value for the category variable. The posterior probability of the goal replaces each category in the case of categorical target variables.

In this study one-hot encoding technique has been used to encode the categorical variables.

One Hot Encoding

One hot encoding is the method where the data is represented in binary format and included as a feature. It is one of the most common methods, comparing each level of the numerical variable with a fixed starting point. In this thesis for the data set that had taken, one hot encoding is used to represent categorical variables as binary vectors.

This approach results in a dummy variable trap because it is easy to predict the outcome of one variable with the support of the existing variables. This trap leads to a multicollinearity problem. It occurs when the independent features become dependent upon each other. To overcome the multicollinearity problem one of the dummy variables should be dropped. The following figure represents the before and after one hot coding.

```
X.head()
```

	plate_size	inclination_angle	embedment_ratio	frequency	aplitude	soil_type
0	25	30	1	0.2	2	unreinforced
1	25	30	2	0.2	2	unreinforced
2	25	30	3	0.2	2	unreinforced
3	25	45	1	0.2	2	unreinforced
4	25	45	2	0.2	2	unreinforced

```
X.tail()
```

	plate_size	inclination_angle	embedment_ratio	frequency	aplitude	soil_type
211	75	45	2	0.5	5	reinforced
212	75	45	3	0.5	5	reinforced
213	75	60	1	0.5	5	reinforced
214	75	60	2	0.5	5	reinforced
215	75	60	3	0.5	5	reinforced

Fig.3.7: Dataset Before Encoding

```
X.head()
```

	plate_size	inclination_angle	embedment_ratio	frequency	aplitude	unreinforced
0	25	30	1	0.2	2	1
1	25	30	2	0.2	2	1
2	25	30	3	0.2	2	1
3	25	45	1	0.2	2	1
4	25	45	2	0.2	2	1

```
X.tail()
```

	plate_size	inclination_angle	embedment_ratio	frequency	aplitude	unreinforced
211	75	45	2	0.5	5	0
212	75	45	3	0.5	5	0
213	75	60	1	0.5	5	0
214	75	60	2	0.5	5	0
215	75	60	3	0.5	5	0

Fig. 3.8: Dataset After Encoding

3.11. Hyperparameter Tuning:

Hyperparameter tuning is a crucial step in the process of developing machine learning models. Hyperparameters are settings that are not learned directly from the data during the training process, but rather are set by the user before training begins. They have a significant impact on the performance and generalization ability of the model.

The purpose of hyperparameter optimization is to find the global optimal value x^* of the objective function f(x) can be evaluated for any arbitrary $\boldsymbol{x}\in\boldsymbol{X}$, x^*=arg min $\boldsymbol{x}\in\boldsymbol{X}$ $f(xx)$, and X is a hyperparameter space that can contain categorical, discrete, and continuous variables. In order to construct the design of different machine learning models, the application of effective hyperparameter optimization techniques can simplify the process of identifying the best hyperparameters for the models. HPO contains four major components: First, an estimator that could be a regressor or any classifier with one or more objective functions, second: a search space, Third: an optimization method to find the best combinations, and Fourth: a function to

make a comparison between the effectiveness of various hyperparameter configurations. Some of the common hyperparameter techniques which has been used in this study are discussed below:

- **Grid Search**: Grid search is a process that exhaustively searches a manually specified subset of the hyperparameter space of the target algorithm. A traditional approach to finding the optimum is to do a grid search, for example, to run experiments or processes on a number of conditions, for example, if there are three factors, a 15 × 15× 15 would mean performing 3375 experiments under different conditions. Grid search is more practical when: (1) the total number of parameters in the model is small, say M <10. The grid is M- dimensional, so the number of test solutions is proportional to L^M, where L is the number of test solutions along each dimension of the grid. (2) The solution is known to be within a specific range of values, which can be used to define the limits of the grid. (3) The direct problem d=g(m) can be computed quickly enough that the time required to compute L^M from them is not prohibitive. (4) The error function E(m) is uniform on the scale of the grid spacing, Δm , so that the minimum is not lost because the grid spacing is too coarse.

  There are many problems with the grid search method. The first is that the number of experiments can be prohibitive if there are several factors. The second is that there can be significant experimental error, which means that if the experiments are repeated under identical conditions, different responses can be obtained; therefore, choosing the best point on the grid can be misleading, especially if the optimum is fairly flat. The third is that the initial grid may be too small for the number of experiments to be feasible, and it could lose characteristics close to the optimum or find a false (local) optimum.
- **Random Search:** Random search is a basic improvement on grid search. It indicates a randomized search over hyper-parameters from certain distributions over possible parameter values. The searching process continues till the predetermined budget is exhausted, or until the desired accuracy is reached. These methods are the simplest stochastic optimization and are very useful for certain problems, such as small search space and fast-running simulation. RS finds a value for each hyperparameter, prior to the probability distribution function. Both the GS and RS estimate the cost measure based on the produced hyperparameter sets. Although RS is simple, it has proven to be more effective than Grid search in many of the cases.

  Random search has been shown to provide better results due to several benefits: first, the budget can be set independently according to the distribution of the search space, therefore, random search can work better especially when multiple hyper-parameters are not uniformly distributed. Second: Because each evaluation is independent, it is easy to parallelize and allocate resources. Unlike GS, RS samples a number of parameter combinations from a defined distribution, which maximizes system efficiency by reducing the likelihood of wasting a lot of time in a small, underperforming area. In addition, this method can detect global optimum values or close to global if given a sufficient budget. Third, although getting optimal results using random search is not promising, more time consumption will lead to a greater likelihood of finding the best hyperparameter set, whereas longer search times cannot guarantee better results in Grid searches.

The use of random search is recommended in the early stages of HPO to narrow the search space quickly, before using guided algorithms to get better results. The main drawback of RS and GS is that each evaluation in its iteration does not depend on previous evaluations; thus, they waste time evaluating underperforming areas of the search space.

3.12. Applying Hyperparameter Tuning in ML Models:

In order to put the theory into practice, several experiments have been performed on an industrial based synthetic polymer model. This section describes experiments with two different HPO techniques on four general and representative ML algorithms. In the first part of the section, we discussed the experimental setup and the main HPO process. In the second part, we compare and analyze the results of the application of different HPO methods.

An overview of common ML models we used in this work, their hyper-parameters are listed below:

ML Model	Hyper-parameter
Simple Linear Regression	copy_X, fit_intercept, n_jobs, normalize, positive
Random Forest Regression	n_estimators, max_depth, max_features, min_samples_split, min_samples_leaf, bootstrap
Stochastic Gradient (SGD) Regression	loss, penalty, alpha, l1_ratio, max_iter
XGBoost Regression	learning_rate, n_estimators, max_depth, subsample, colsample_bytree, gamma

3.13. K-fold Cross-Validation:

Cross-validation (CV) is a procedure of statistical analysis used to assess the effectiveness of a Machine Learning technique, as well as a re-sampling method used to validate an algorithm if there is insufficient data. Stratification is the process of rearranging the data to ensure each fold is a good representative of the whole. Data splitting into folds may be controlled by criteria such as ensuring where each fold has the same ratio of outcomes with a given categorical value, such as the class outcome value. This process is called stratified k-fold cross-validation. Common techniques of cross-validation include K-fold cross validation, Stratified K-fold cross-validation, and cross-validation leave-one-out. The motivation behind the 5-fold stratified cross-validation is that the estimator has a lower variance than a single hold-out set estimation method which could be very essential if there is a limited amount of data. There will be plenty of variance in the results estimate for various data samples, or for specific data partitions to create training and test sets. The 5-fold stratified cross-validation removes this variance by comparing more than 5 separate partitions, thereby making the performance estimate less sensitive to data partitioning.

3.14. Performance Metrics:

Several metrics can be used while evaluating how well a model is performing. It is necessary to understand how each metric measures to select the evaluation metric to better assess the

model. This thesis main objective was to compare the performance of Machine Learning techniques by evaluating all of these performance metrics such as R2 score, Mean Absolute Error, Mean squared error, Max error and Mean absolute percentage error.

3.14.1. R2 Score:

The R-squared (R2) score, also known as the coefficient of determination, is a statistical measure used to evaluate the goodness of fit of a regression model. It provides information about the proportion of the variance in the dependent variable that is explained by the independent variables in the model. The R2 score ranges from 0 to 1, where 0 indicates that the model does no explain any variability in the dependent variable and 1 indicates that the model perfectly explains the variability in the dependent variable. However, it's important to note that a high R2 score does not necessarily mean that the model is a good fit for the data, as it might still suffer from overfitting or other issues.

Mathematically, the R2 score is calculated as:

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

Where,

SSres is the sum of squared residuals (the sum of the squared differences between the actual and predicted values).

SStot is the total sum of squares (the sum of the squared differences between the actual values and the mean of the dependent variables).

3.14.2 Mean Absolute Error:

Mean Absolute Error is a process performance measure that is used for regression models. A model's mean absolute error concerning a test data set is the average of the actual values on all instances in the test set of the specific prediction errors. For instance, every predictive error is the difference between the predicted value and the actual value. Mean Absolute Error is one of several metrics for summing up and measuring the Machine Learning model's performance.

$$MAE(y, x) = \left(\frac{1}{n_{samples}}\right) \sum_{i=0}^{n_{samples}-1} |y_i - x_i|$$

Where y_i describes the actual values, x_i describes the expected values.

3.14.3. Mean Squared Error:

Mean squared error (MSE) is another commonly used metric for evaluating the performance of regression models. It measures the average of the squared differences between the actual and predicted values of the target variable. In other words, it quantifies the average squared "error" between the predicted values and the actual values, giving more weight to larger errors.

The formula for the calculating the MSE is as follows:

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

Where,

n is the number of data points.

y_i is the actual target value for the ith data point.

$\hat{y}_i$ is the predicted value for the ith data point.

3.14.4 Max Error:

The function max error measures the maximum standard errors, a metric representing a worse-case error between the expected value and the actual value. Max error Chapter 3. Method 18 would be 0 on the test set in a properly fitted single-output regression analysis, and while this would be extremely impossible in the modern world, this measurement indicates the amount of error the model has when it was placed in.

$$MaxError(y, x) = max(|y_i - x_i|)$$

Where y_i describes the actual values, x_i describes the expected values.

3.14.5. Mean Absolute Percentage Error:

The mean absolute percentage error (MAPE) is a metric used to evaluate the accuracy of a forecasting or regression model in terms of the percentage difference between the predicted and actual values. It is particularly useful when you want to understand the relative magnitude the predicted and actual values of a variable.

The formula for calculating MAPE is as follows:

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i - \hat{y}_i|}{y_i} \cdot 100\%$$

where,

y_i is the actual target value for the ith data point.

$\hat{y}_i$ is the predicted value for the ith data point.

CHAPTER-4

RESULTS

After hyperparameter tunning, Simple Linear Regression, Random Forest Regressor, SGD Regressor and XGBoost Regressor are trained with the set of data using K-Fold cross-validation approach that dynamically selected the training and testing with fixed proportion each time and efficiency was calculated using R2 score, mean absolute error, mean squared error, max error, mean absolute percentage error metrics.

4.1. Simple Linear Regressor:

The box plot in Fig. 4.1 shows the R2 score obtained by the Simple Linear Regressor during a 10-fold cross-validation test. The upper line of box-plot represents the maximum score of 0.962, the middle line represents the median of R2 score of 0.928, and the lower line of the box-plot represents the minimum score of 0.906.

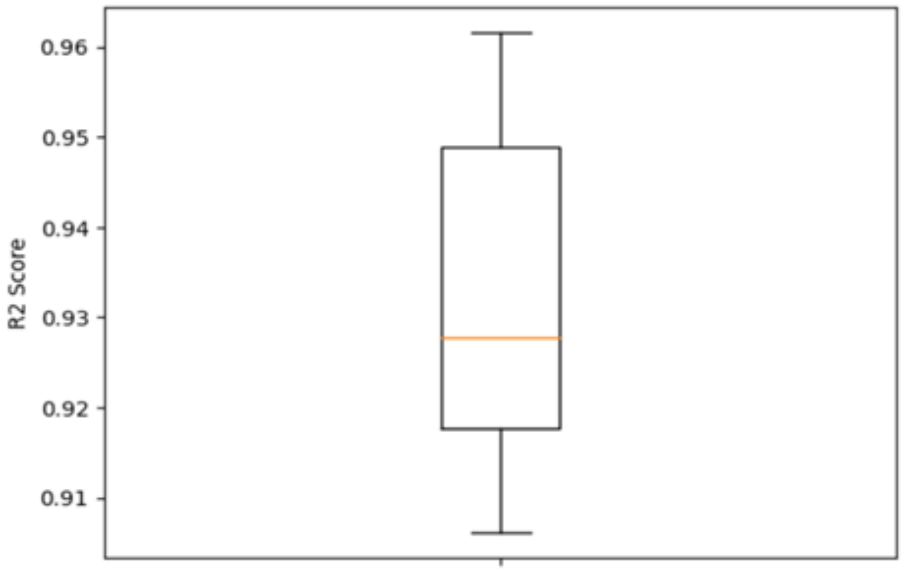

Fig. 4.1: R2 Score for Simple Linear Regressor

The box plot in Fig. 4.2 shows the mean absolute error (MAE) obtained by the Simple Linear Regressor during a 10-fold cross-validation test. The upper line of box-plot represents the maximum MAE (excluding the outlier of 13.212) of 11.348, the middle line represents the median of MAE of 7.979, and the lower line of the box-plot represents the minimum MAE of 6.658.

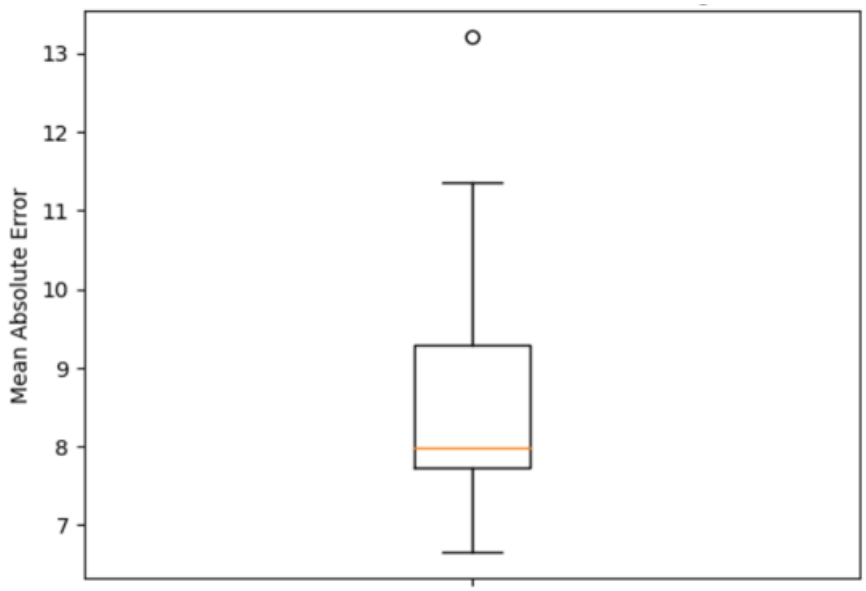

Fig. 4.2: MAE for Simple Linear Regressor

The box plot in Fig. 4.3 shows the mean squared error (MSE) obtained by the Simple Linear Regressor during a 10-fold cross-validation test. The upper line of box-plot represents the maximum MSE (excluding the outlier of 250.223) of 177.967, the middle line represents the median of MAE of 99.985, and the lower line of the box-plot represents the minimum MAE of 60.561.

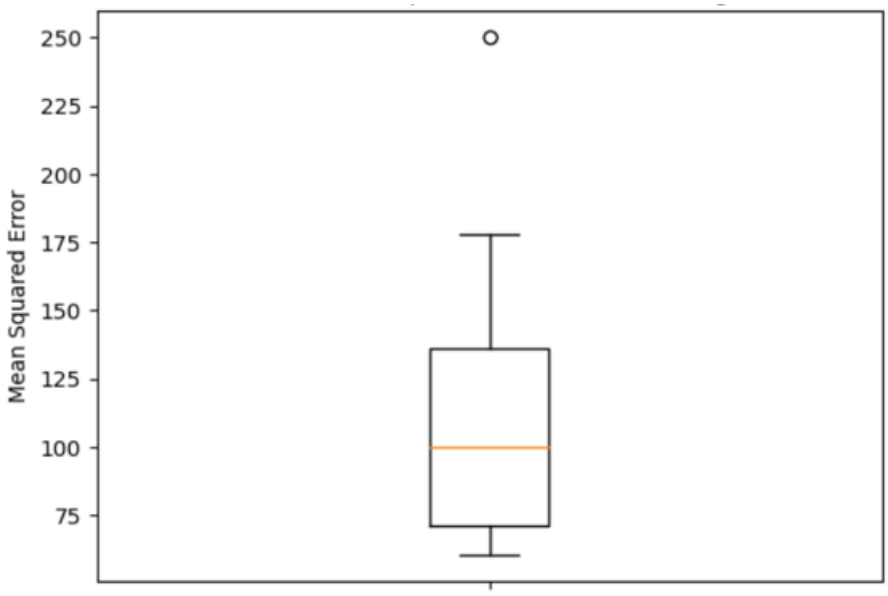

Fig. 4.3: MSE for Simple Linear Regressor

The box plot in Fig. 4.4 shows the maximum error (ME) obtained by the Simple Linear Regressor during a 10-fold cross-validation test. The upper line of box-plot represents the maximum ME of 38.911, the middle line represents the median of ME of 22.563, and the lower line of the box-plot represents the minimum ME of 14.183.

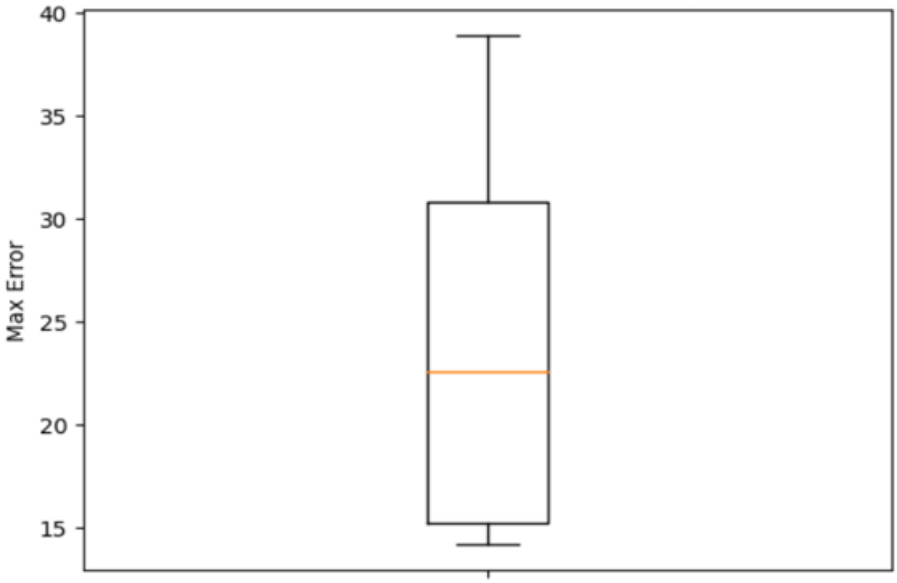

Fig. 4.4: ME for Simple Linear Regressor

The box plot in Fig. 4.5 shows the mean absolute percentage error (MAPE) obtained by the Simple Linear Regressor during a 10-fold cross-validation test. The upper line of box-plot represents the maximum MAPE (excluding the outlier of 0.205) of 0.162, the middle line represents the median of MAPE of 0.129, and the lower line of the box-plot represents the minimum MAPE of 0.093.

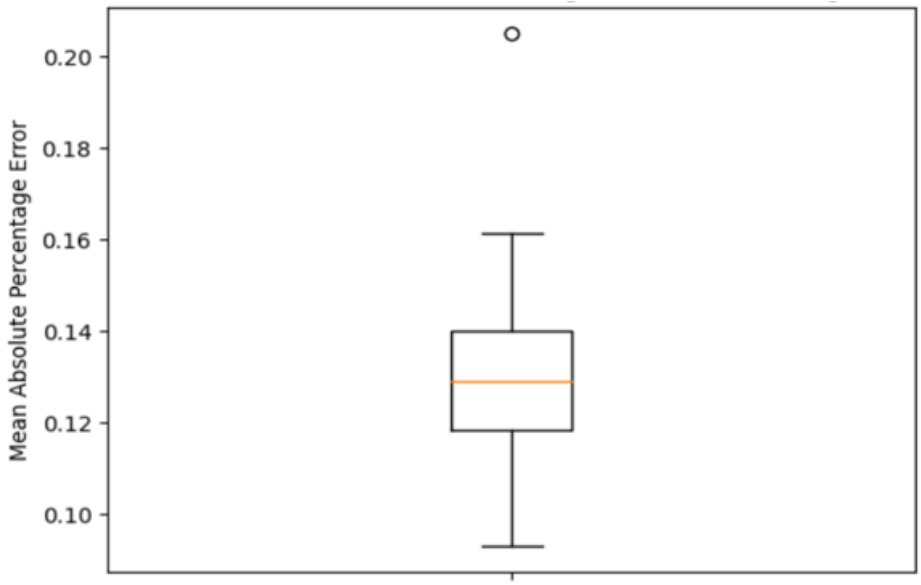

Fig. 4.5: MAPE for Simple Linear Regressor

4.2. Random Forest Regressor:

The box plot in Fig. 4.6 shows the R2 score obtained by the Random Forest Regression (which has been hyperparameter tuned by Bayesian Optimization Technique) during a 10-fold cross-validation test. The upper line of box-plot represents the maximum score of 0.993, the middle line represents the median of R2 score of 0.982, and the lower line of the box-plot represents the minimum score of 0.972.

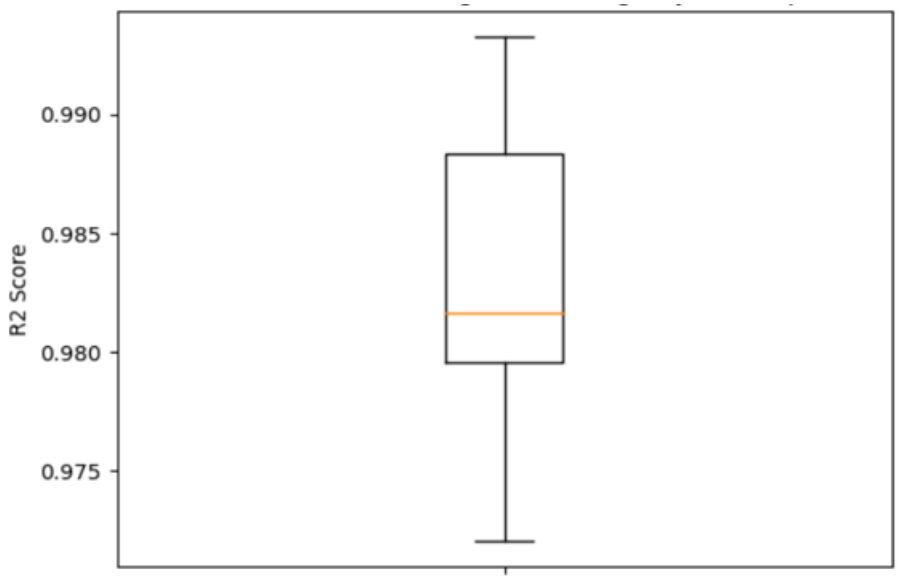

Fig. 4.6: R2 Score for Random Forest Regressor

The box plot in Fig. 4.7 shows the mean absolute error (MAE) obtained by the Random Forest Regressor during a 10-fold cross-validation test. The upper line of box-plot represents the maximum MAE of 5.739, the middle line represents the median of MAE of 3.755, and the lower line of the box-plot represents the minimum MAE of 2.411.

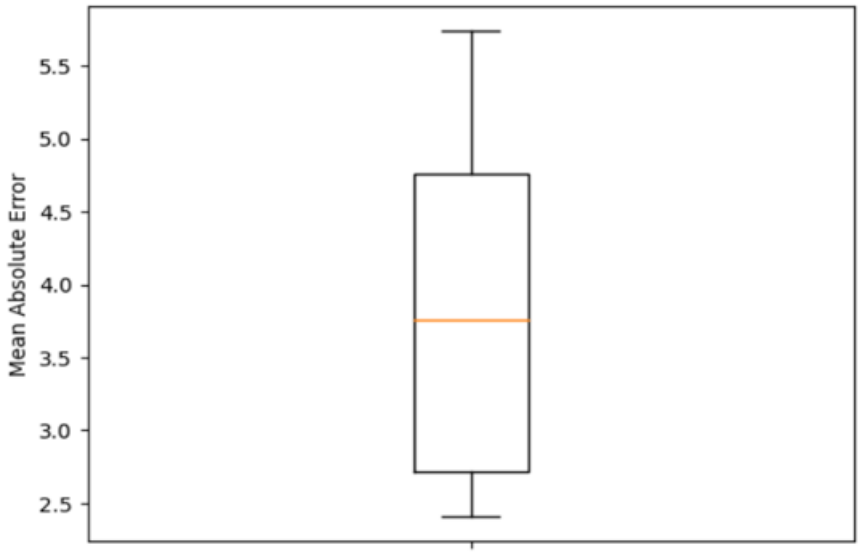

Fig. 4.7: MAE for Random Forest Regressor

The box plot in Fig. 4.8 shows the mean squared error (MSE) obtained by the Random Forest Regressor during a 10-fold cross-validation test. The upper line of box-plot represents the maximum MSE (excluding the outlier of 79.316) of 49.640, the middle line represents the median of MSE of 27.095, and the lower line of the box-plot represents the minimum MSE of 9.467.

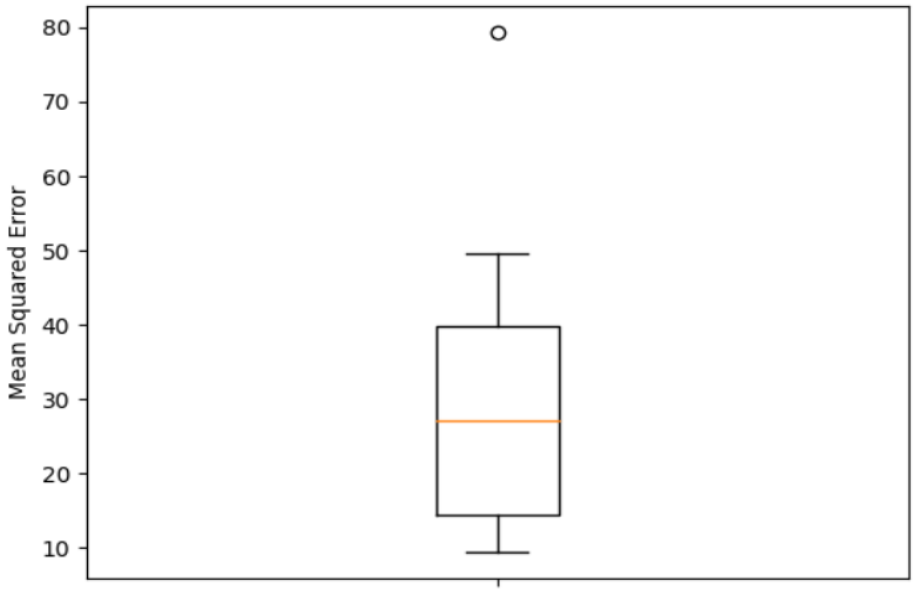

Fig. 4.8: MSE for Random Forest Regressor

The box plot in Fig. 4.9 shows the maximum error (ME) obtained by the Random Forest Regressor during a 10-fold cross-validation test. The upper line of box-plot represents the maximum ME of 27.991, the middle line represents the median of ME of 13.775, and the lower line of the box-plot represents the minimum ME of 6.937.

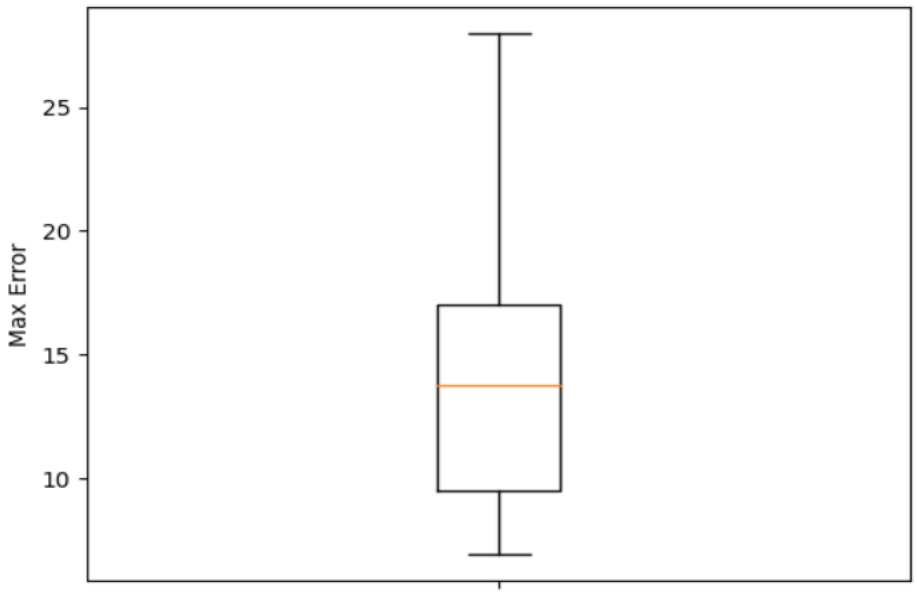

Fig. 4.9: ME for Random Forest Regressor

The box plot in Fig. 4.10 shows the mean absolute percentage error (MAPE) obtained by the Random Forest Regressor during a 10-fold cross-validation test. The upper line of box-plot represents the maximum MAPE of 0.060, the middle line represents the median of MAPE of 0.044, and the lower line of the box-plot represents the minimum MAPE of 0.033.

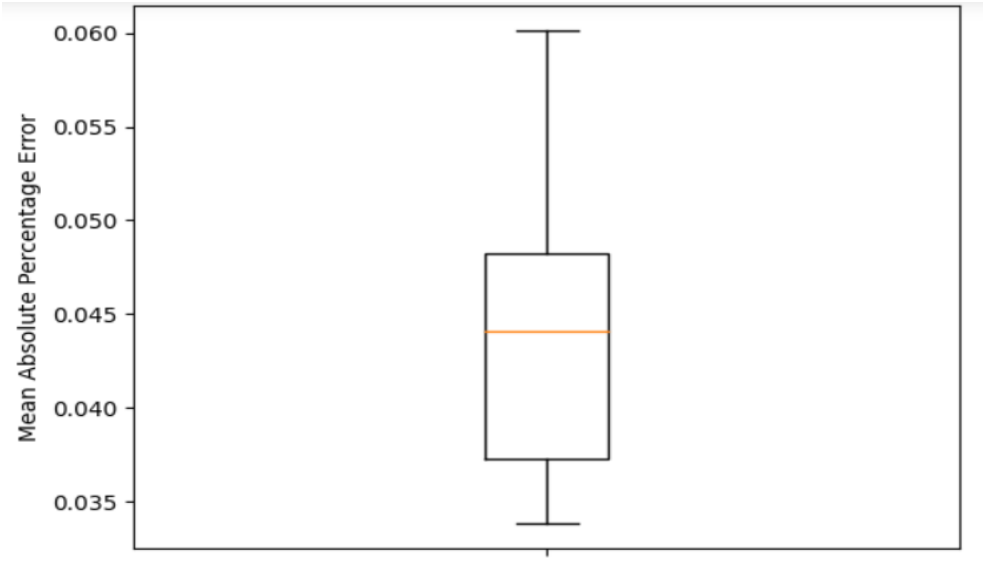

Fig. 4.10: MAPE for Random Forest Regressor

4.3. SGD Regressor:

The box plot in Fig. 4.11 shows the R2 score obtained by the SGD Regressor (which has been hyperparameter tuned by Random Search Technique) during a 10-fold cross-validation test. The upper line of box-plot represents the maximum score of 0.858, the middle line represents the median of R2 score of 0.723, and the lower line of the box-plot represents the minimum score of 0.405.

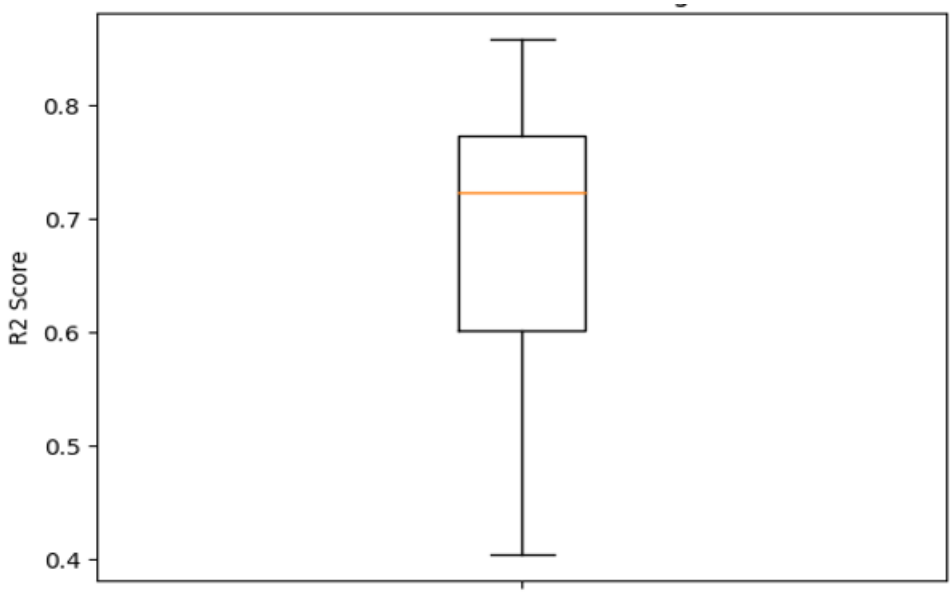

Fig. 4.11: R2 Score for SGD Regressor

The box plot in Fig. 4.12 shows the mean absolute error (MAE) obtained by the SGD Regressor during a 10-fold cross-validation test. The upper line of box-plot represents the maximum MAE (excluding the outliers of 24.342 & 28.810) of 18.518, the middle line represents the median of MAE of 16.074, and the lower line of the box-plot represents the minimum MAE of 10.722.

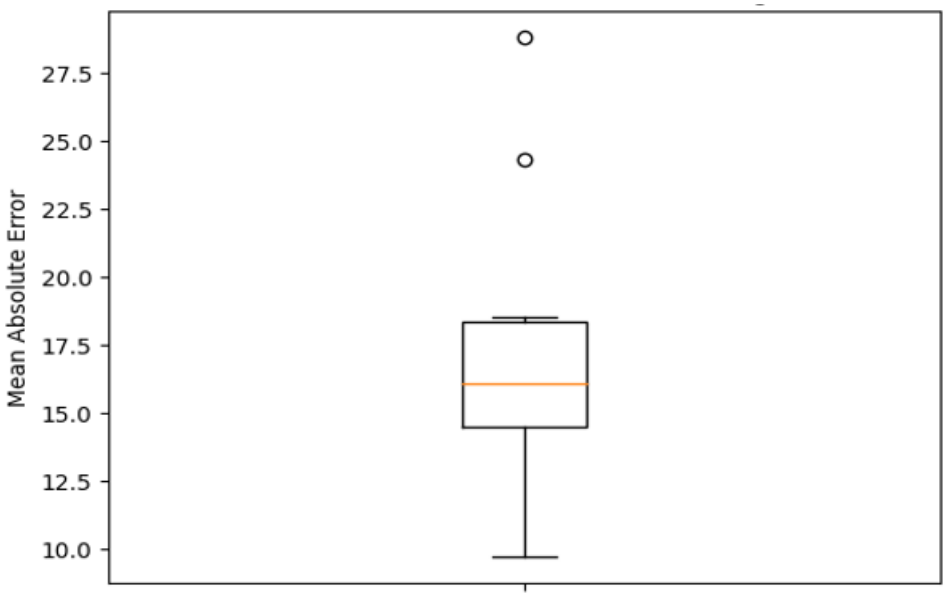

Fig. 4.12: MAE for SGD Regressor

The box plot in Fig. 4.13 shows the mean squared error (MSE) obtained by the SGD Regressor during a 10-fold cross-validation test. The upper line of box-plot represents the maximum MSE (excluding the outliers of 1200.933 & 1515.802) of 771.992, the middle line represents the median of MSE of 442.093, and the lower line of the box-plot represents the minimum MSE of 130.687.

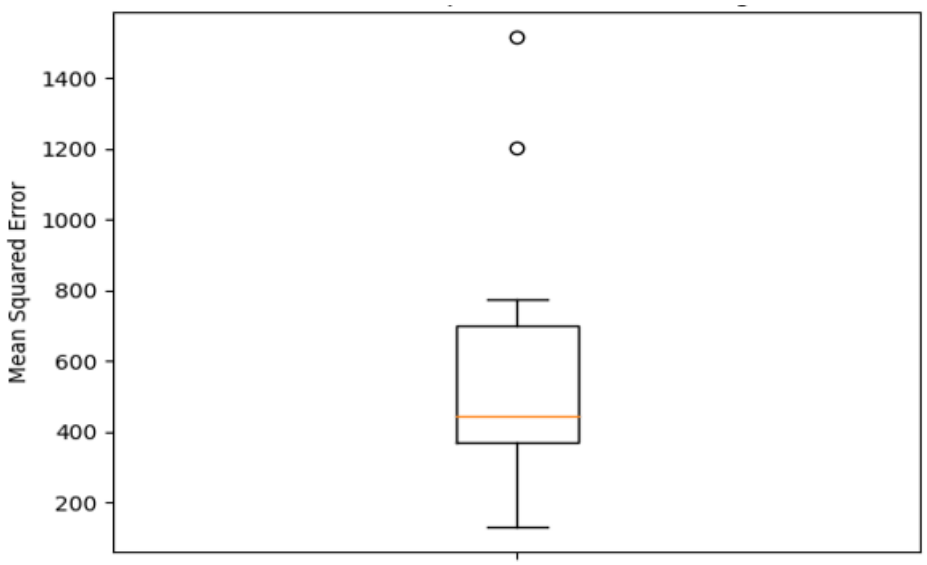

Fig.4.13: MSE for SGD Regressor

The box plot in Fig. 4.14 shows the maximum error (ME) obtained by the SGD Regressor during a 10-fold cross-validation test. The upper line of box-plot represents the maximum ME of 94.207, the middle line represents the median of ME of 59.472, and the lower line of the box-plot represents the minimum ME of 18.941.

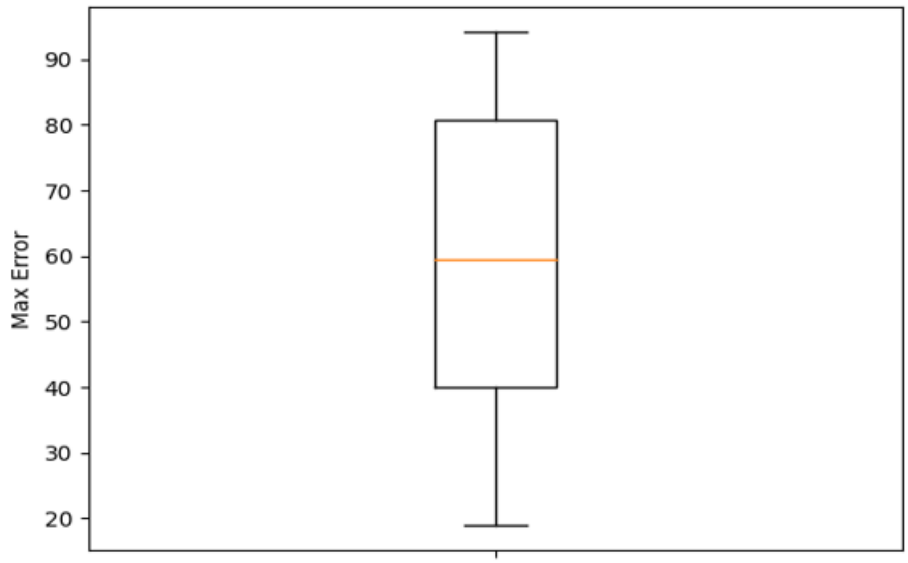

Fig. 4.14: ME for SGD Regressor

The box plot in Fig. 4.15 shows the mean absolute percentage error (MAPE) obtained by the SGD Regressor during a 10-fold cross-validation test. The upper line of box-plot represents the maximum (excluding the outliers of 0.279 & 0.312) MAPE of 0.215, the middle line represents the median of MAPE of 0.204, and the lower line of the box-plot represents the minimum MAPE of 0.145.

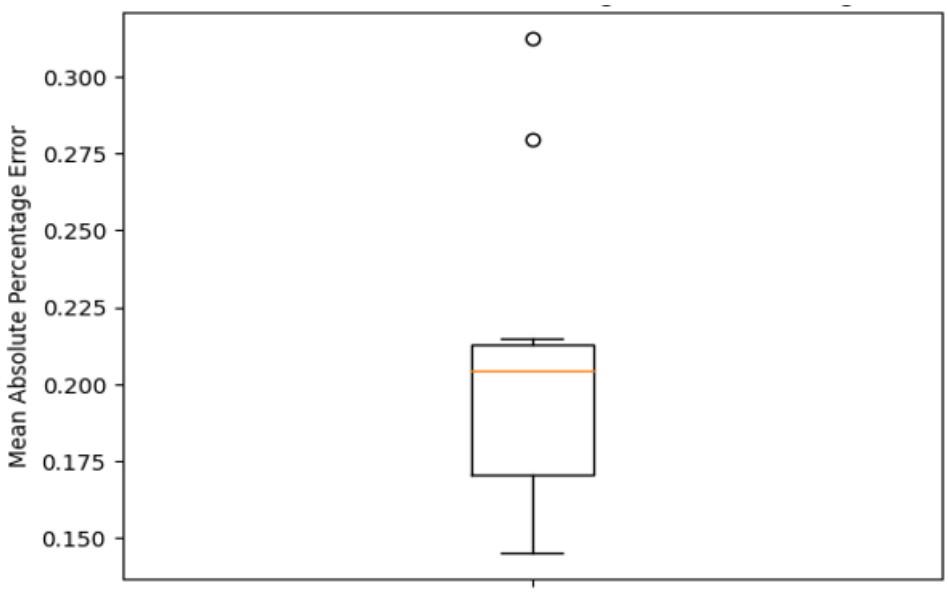

Fig. 4.15: MAPE for SGD Regressor

4.4. XGBoost Regressor:

The box plot in Fig. 4.16 shows the R2 score obtained by the SGD Regressor (which has been hyperparameter tuned by Random Search Technique) during a 10-fold cross-validation test. The upper line of box-plot represents the maximum score of 0.998, the middle line represents the median of R2 score of 0.997, and the lower line of the box-plot represents the minimum score (excluding the outlier of 0.994) of 0.996.

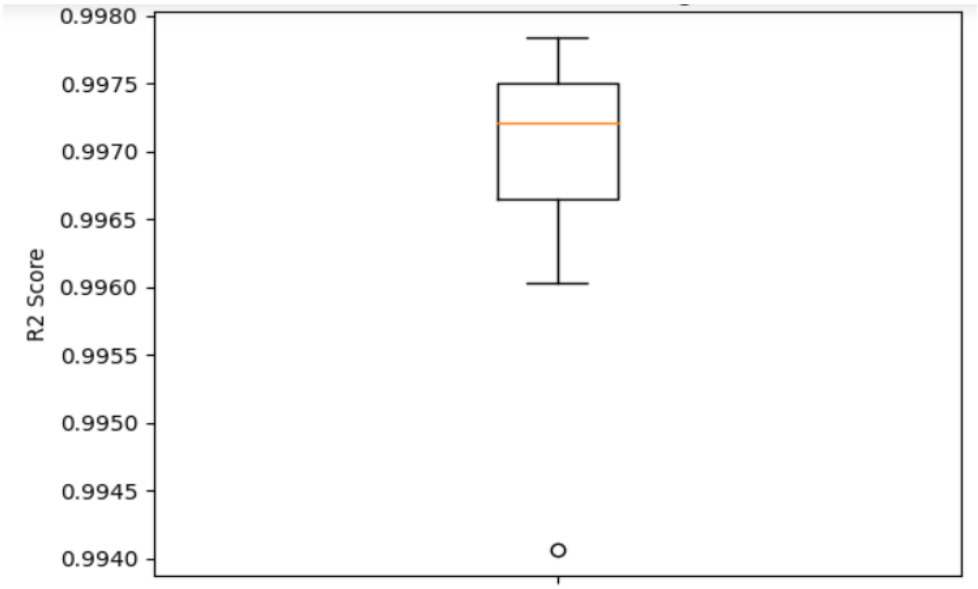

Fig. 4.16: R2 Score for XGBoost Regressor

The box plot in Fig. 4.17 shows the mean absolute error (MAE) obtained by the XGBoost Regressor during a 10-fold cross-validation test. The upper line of box-plot represents the maximum MAE (excluding the outlier of 2.851) of 2.029, the middle line represents the median of MAE of 1.704, and the lower line of the box-plot represents the minimum MAE of 1.276.

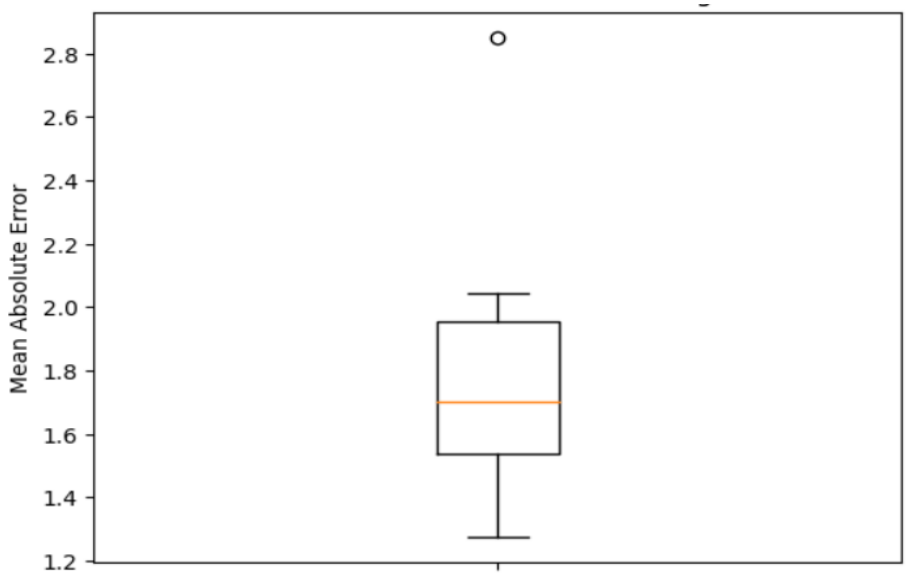

Fig. 4.17: MAE for XGBoost Regressor

The box plot in Fig. 4.18 shows the mean squared error (MSE) obtained by the XGBoost Regressor during a 10-fold cross-validation test. The upper line of box-plot represents the maximum MSE (excluding the outlier of 12.052) of 5.813, the middle line represents the median of MSE of 4.928, and the lower line of the box-plot represents the minimum MSE of 2.656.

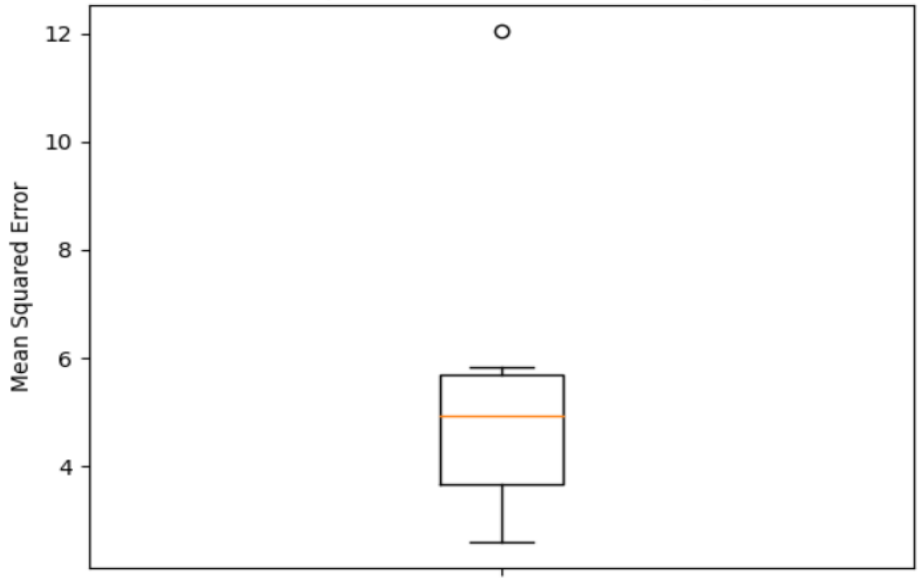

Fig. 4.18: MSE for XGBoost Regressor

The box plot in Fig. 4.19 shows the maximum error (ME) obtained by the XGBoost Regressor during a 10-fold cross-validation test. The upper line of box-plot represents the maximum ME of 8.712, the middle line represents the median of ME of 5.157, and the lower line of the box-plot represents the minimum ME of 3.664.

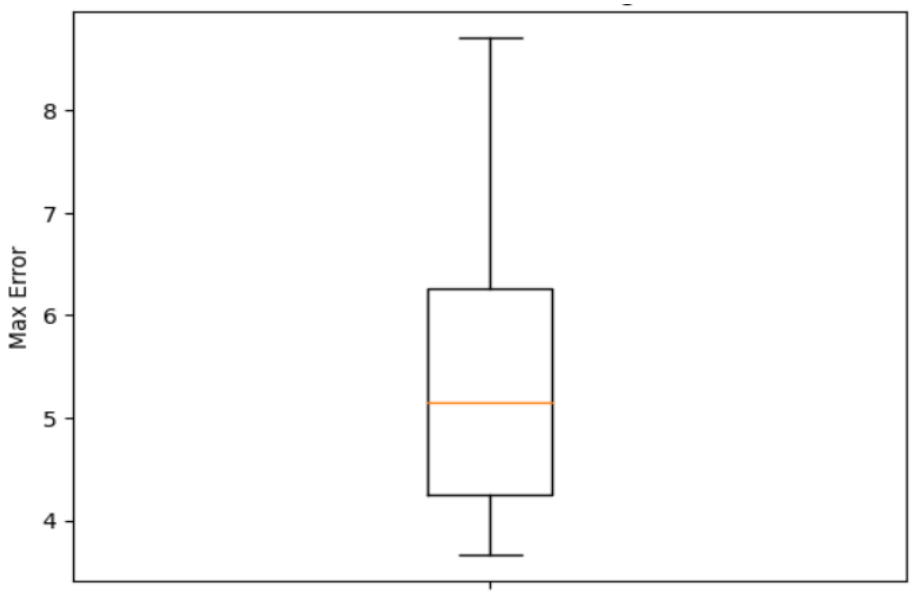

Fig. 4.19: ME for XGBoost Regressor

The box plot in Fig. 4.20 shows the mean absolute percentage error (MAPE) obtained by the XGBoost Regressor during a 10-fold cross-validation test. The upper line of box-plot represents the maximum MAPE of 0.036, the middle line represents the median of MAPE of 0.024, and the lower line of the box-plot represents the minimum MAPE of 0.019.

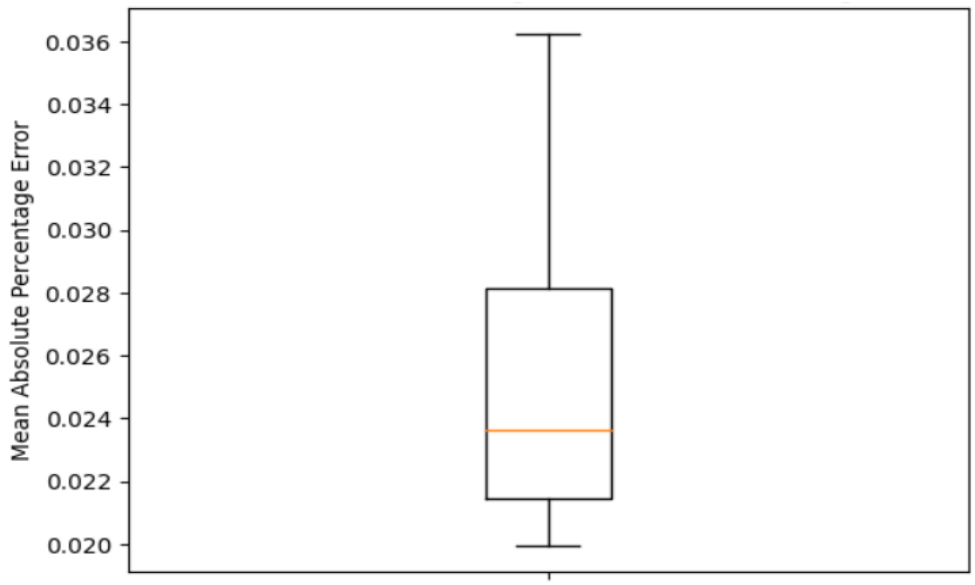

Fig. 4.20: MAPE for XGBoost Regressor

Table 4.1. Summary of evaluation results for each ML models with different hyperparameter tunning process

SCORE	LINEAR REGRESSION	RANDOM FOREST REGRESSION				SGD REGRESSION				XGBOOST REGRESSION			
		Default HP	Bayesian optimization Technique	Random Search Technique	Grid Search CV Technique	Default HP	Bayesian optimization Technique	Random Search Technique	Grid Search CV Technique	Default HP	Bayesian optimization Technique	Random Search Technique	Grid Search CV Technique
R2 Score	0.923	0.976	0.977	0.977	0.971	0.923	0.923	0.923	0.923	0.989	0.997	0.997	0.996
Mean Absolute Error (MAE)	8.099	4.371	3.429	4.295	4.947	8.098	8.097	8.098	8.099	2.743	1.639	1.638	1.816
Mean Squared Error (MSE)	127.720	40.253	31.914	37.969	48.575	127.653	127.726	127.817	127.704	18.089	4.651	4.602	6.478
Maximum Error (ME)	42.773	18.592	21.450	19.303	19.479	42.803	42.822	42.872	42.780	17.583	7.638	5.371	7.698
Mean Absolute Percentage Error (MAPE)	0.117	0.048	0.0379	0.046	0.054	0.117	0.117	0.117	0.118	0.029	0.019	0.021	0.023

Table 4.1 shows the comparison of evaluation results where XGBoost Regression (with hyperparameter tuned by Random Search technique) performed well with all the metrics R2 score, Mean Absolute Error, Max Error and Mean Absolute Percentage Error. XGBoost Regression had the minimum error in predicting the uplift load when compared to the Simple Linear Regression, Random Forest regression and Stochastic Gradient Descent Regression. SGD Regression demonstrated the worst performance with the highest error in all the metrics. A simplified tabular form based on the results is created above.

CHAPTER-5

ANALYSIS AND DISCUSSION

5.1. Performance of ML Models:

From fig. 5.1 to fig. 5.4 illustrate the correlation actual uplift load and predicted uplift load using the optimized ML models, along with ±10% error lines (red lines). The green lines show a match between actual and predicted values. The results demonstrate that among all ML models, the XGBoost model, achieved excellent predictive accuracy on both training and testing datasets with high R2 Score and low MAE, MSE, ME and MPAE. This shows that the XGBoost model can explain all the variance in the uplift load using the given features.

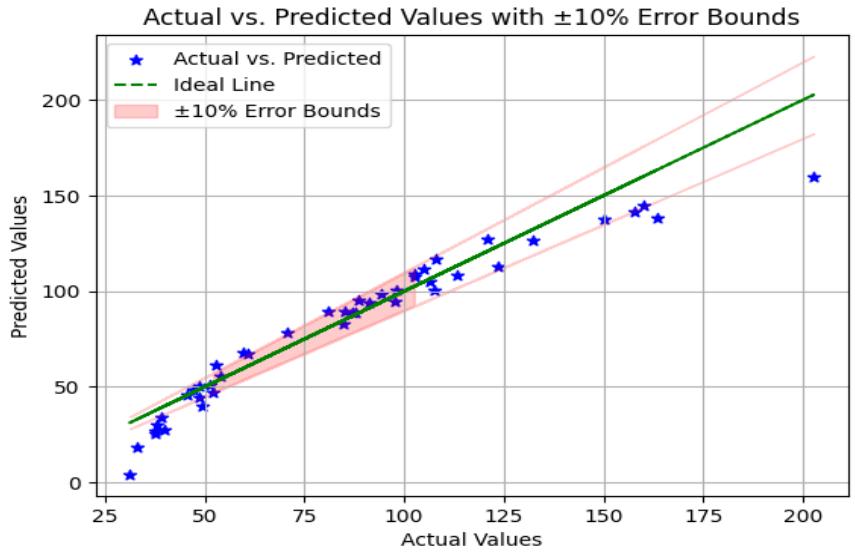

Fig. 5.1. Scatter plot illustrating the correlation between actual vs predicted values (Linear Regression)

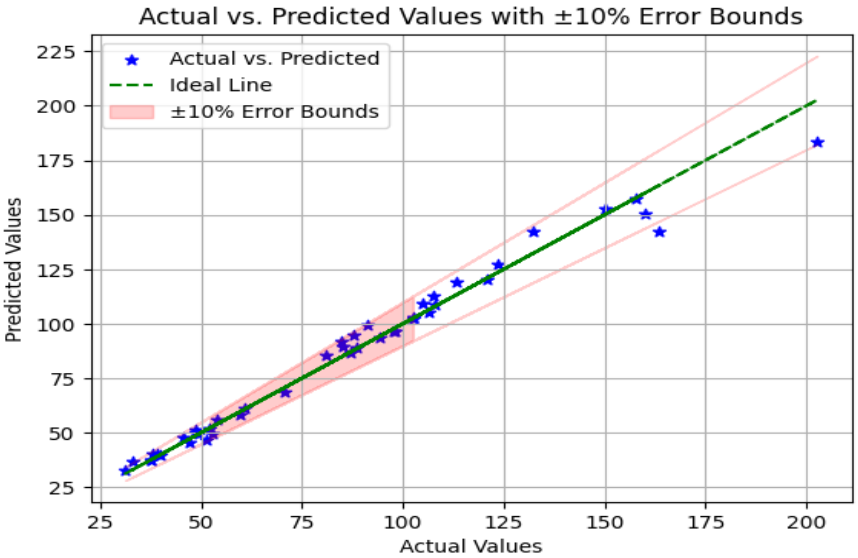

Fig. 5.2. Scatter plot illustrating the correlation between actual vs predicted values (Random Forest Regression)

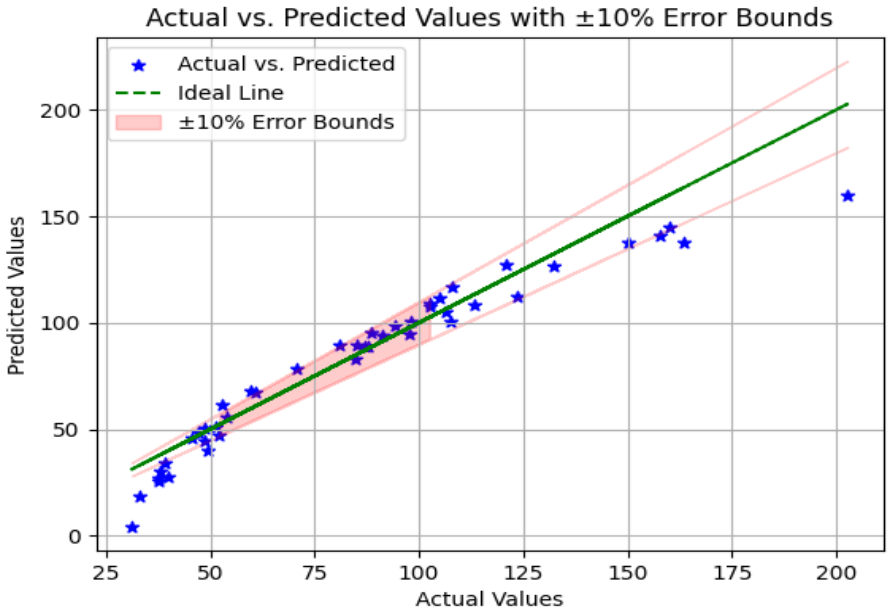

Fig. 5.3. Scatter plot illustrating the correlation between actual vs predicted values (SGD Regression)

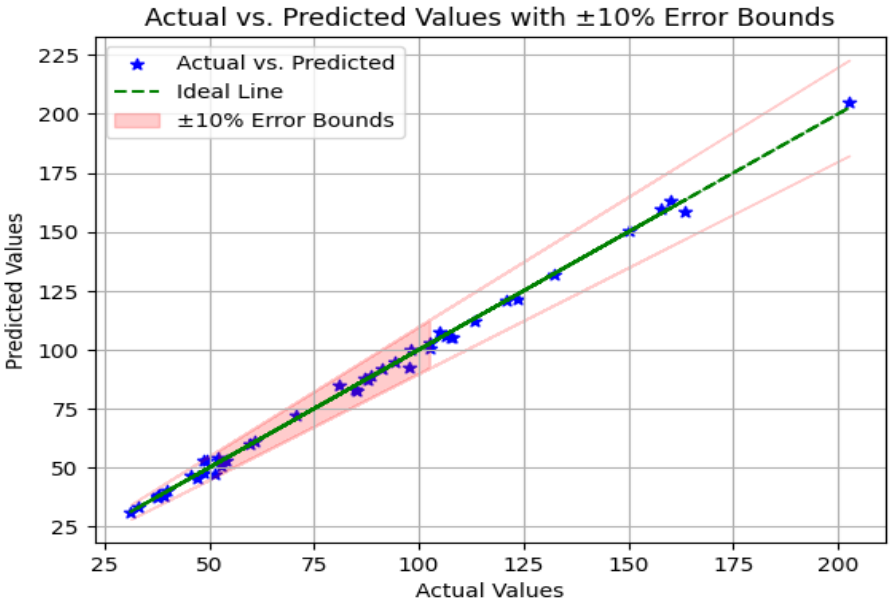

Fig. 5.4. Scatter plot illustrating the correlation between actual vs predicted values (XGBoost Regression)

5.2. Comparative analysis of Performance Metrics:

In this section the average determined on the basis of ten iterations that were considered for all the metrics.

5.2.1. R2 Score:

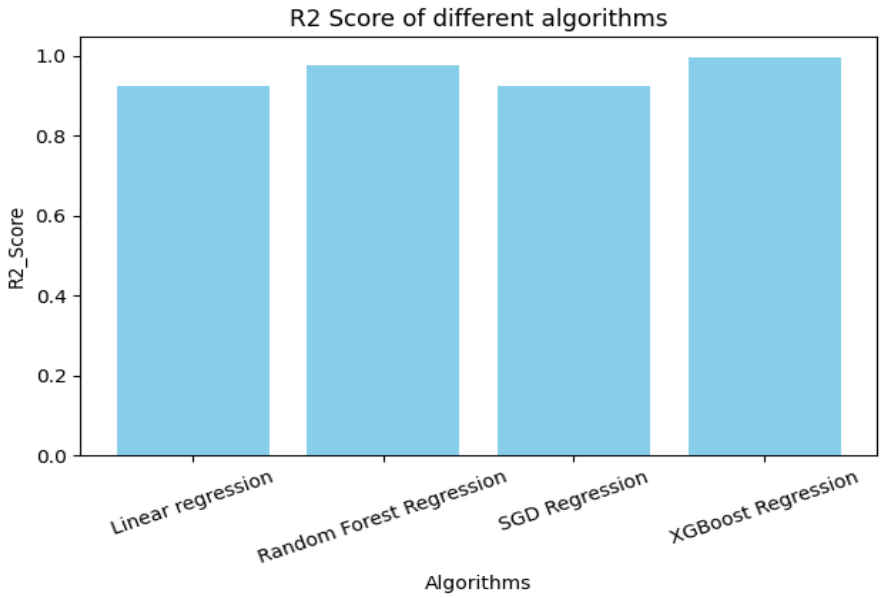

Fig. 5.5: Bar plot of R2 Score obtained from different model

Figure 5.5 shows the R2 score of the 10-fold stratified cross-validation obtained by the Simple Linear Regressor is 0.923, followed by the Random Forest Regressor with 0.977, then SGD Regressor with the 0.923 R2 score and finally XGBoost Regressor with 0.997 R2 score.

From figure 5.5, it can be seen that XGBoost Regressor is the best performer with 0.997 R2 score compared to other methods, and Simple Linear Regressor and SGD Regressor are the poor performer with same R2 score of 0.923.

5.2.2. Mean Absolute Error:

Figure 5.6 shows the average MAE of the 10-fold stratified cross-validation obtained by Simple Linear Regressor is 8.099 error, followed by Random Forest Regressor with 3.429 error, then SGD Regressor with 8.098 error, and finally XGBoost Regressor with 1.638 error respectively. From figure 5.6, it can be shown that XGBoost Regressor is the best performer with less error relative to other approaches.

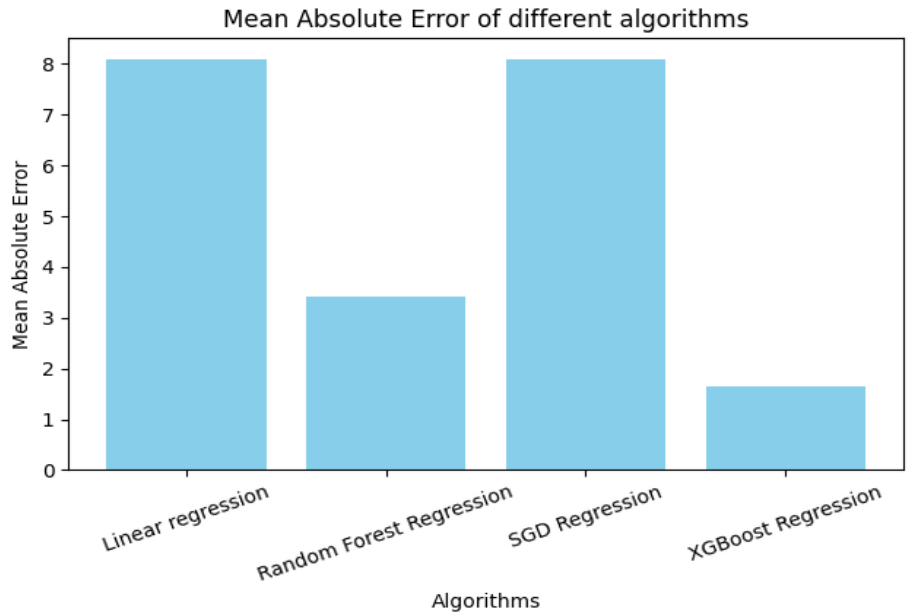

Fig. 5.6: Bar plot of MAE obtained from different model

5.2.3. Mean Squared Error:

Figure 5.7 shows the average MSE of the 10-fold stratified cross-validation obtained by Simple Linear Regressor is 127.720 error, followed by Random Forest Regressor with 31.914 error, then SGD Regressor with 127.653 error, and finally XGBoost Regressor with 4.602 error respectively. From figure 5.7, it can be shown that XGBoost Regressor is the best performer with less error relative to other approaches.

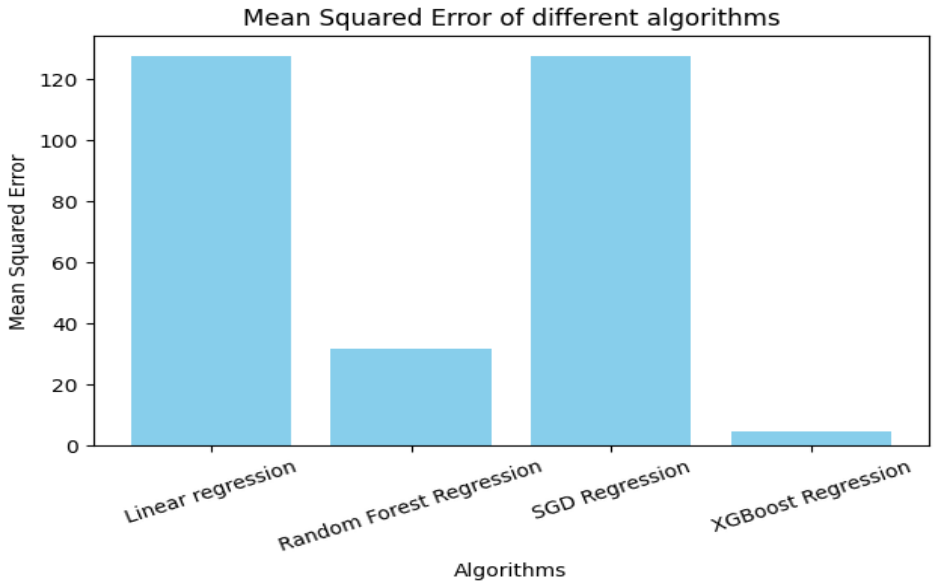

Fig. 5.7: Bar plot of MSE obtained from different model

5.2.4. Max Error:

Figure 5.8 shows the average max error of the 10-fold stratified cross-validation obtained by Simple Linear Regressor is 42.773 error, followed by Random Forest Regressor with 21.450 error, then SGD Regressor with 42.803 error, and finally XGBoost Regressor with 5.371 error respectively. From figure 5.8, it can be shown that XGBoost Regressor is the best performer with less error relative to other approaches.

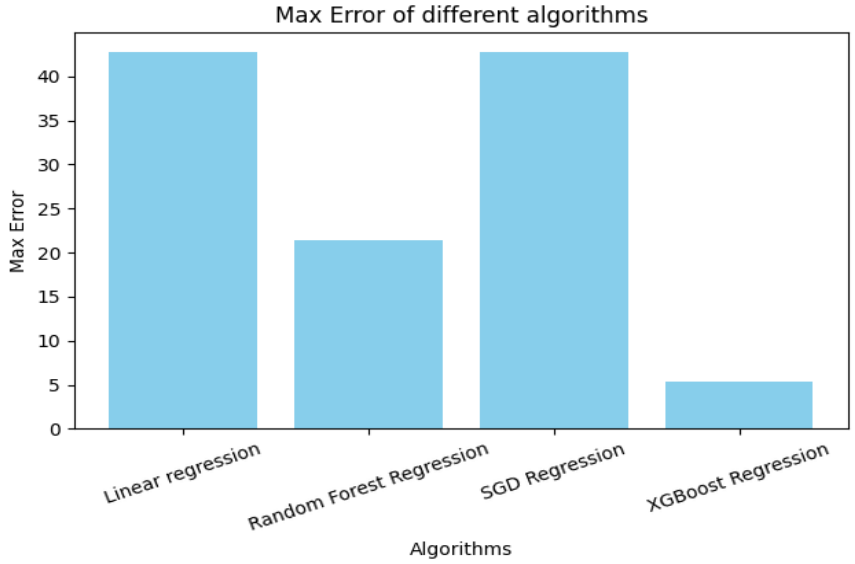

Fig. 5.8: Bar plot of max error obtained from different model

5.2.5. Mean Absolute Percentage Error:

Figure 5.9 shows the average max error of the 10-fold stratified cross-validation obtained by Simple Linear Regressor is 0.117 error, followed by Random Forest Regressor with 0.0379 error, then SGD Regressor with 0.117 error, and finally XGBoost Regressor with 0.021 error respectively. From figure 5.9, it can be shown that XGBoost Regressor is the best performer with less error relative to other approaches.

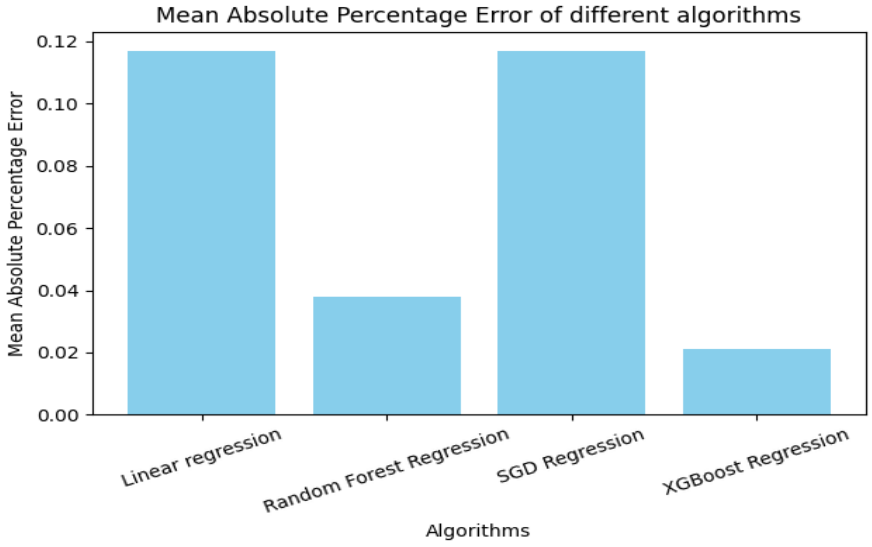

Fig. 5.9: Bar plot of MAPE obtained from different model

5.3. Comparison of ML Models:

Table 5.1 provides a summary of the evaluation results for the ML models. Based on our results, the Simple Linear Regressor and SGD Regressor model exhibited lower accuracy, as evidenced by a lower R2 score on. The Linear Regressor model recorded R2 score of 0.923, MAE of 8.099, MSE of 127.720, ME of 42.773 and MAPE of 0.117 respectively. Additionally, Spider charts were utilized to visualize and assess each model's efficiency relative to others (Fig.5.10). The spider chart shows that the Linear Regressor and SGD Regressor model significantly diverged towards higher MAE, MSE, ME and MAPE in comparison to other ML models. The Random Forest Regressor model perform moderate with the moderate value of R2 score, MAE, MSE, ME and MAPE. The XGBoost model outperform all the models in terms of R2 score, MAE, MSE, ME and MAPE. XGBoost model have lower error values and higher R 2 scores, indicating higher accuracy and better performance in predicting uplift load from the input features.

Table 5.1: Summary of evaluation results for each ML model

Model	R2 Score	MAE	MSE	ME	MAPE
Simple Linear Regressor	0.923	8.099	127.720	42.773	0.117
Random Forest Regressor	0.981	3.429	31.914	21.450	0.039
SGD Regressor	0.923	8.098	127.653	42.822	0.117
XGBoost Regressor	0.997	1.638	4.602	5.371	0.021

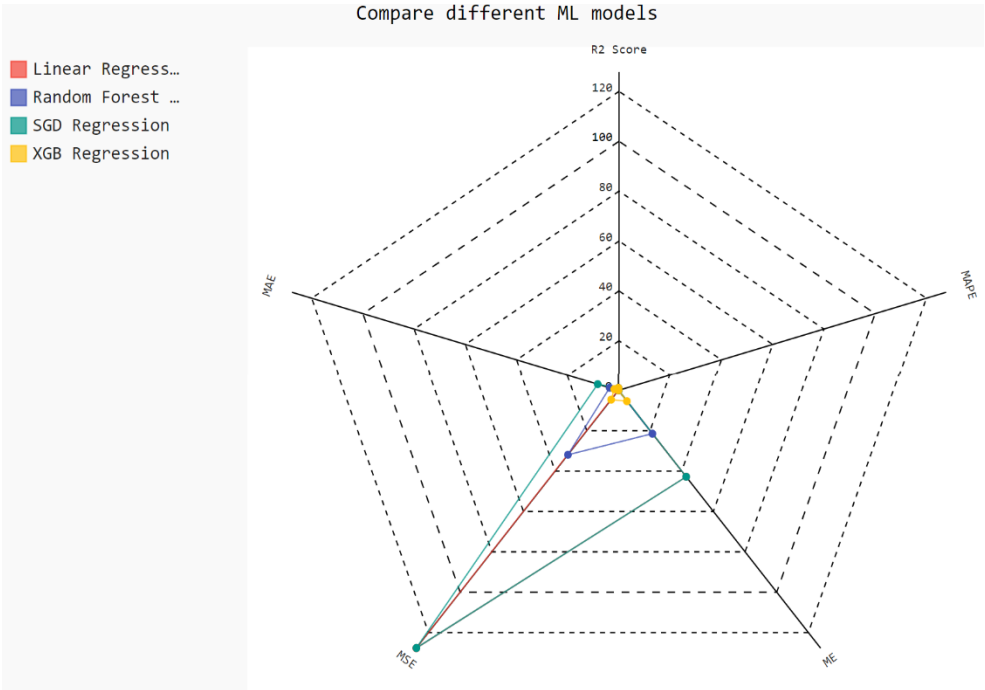

Fig. 5.10: Spider plot showing the performance metrics of the different models.

5.4. Conclusions:

This study utilized various ML algorithms, including Simple Linear Regressor, Random Forest Regressor (RF), Stochastic Gradient Descent Regressor (SGD) and XGBoost Regressor, to predict uplift load from the anchor soil data. To train and test these ML models, we used a previously published open-source CPT dataset. The hyperparameters of each ML model were fine-tuned through various technique like Bayesian optimization, Random Search CV and Grid Search CV technique with 10-fold cross-validation process. Five performance metrics, namely R2 score, Mean Absolute Error (MAE), Mean Squared Error (MSE), Maximum Error (ME) and Mean Absolute Percentage Error (MAPE) provided quantitative evaluation of the models. Based on our results, the following conclusion can be drawn:

• The XGBoost model outperformed the other ML models, achieving the lowest error metrics. Specifically, it achieved R2 score of 0.997, MAE of 1.638, MSE of 4.602, ME of5.731 and MAPE of 0.021% respectively.

• The Random Forest model exhibited poor performance, with higher error metrics compared with the XGBoost model. Random Forest model ranked second, following the RF model, which achieved the highest performance. However, the Linear Regression and SGD model performed poorly, with higher error rates and uncertainty in predicting uplift load.

• The XGBoost model demonstrated its overall superior performance and high accuracy in predicting uplift load, even when trained with minimal input features. Hence, owing to its excellent performance across multiple metrics, the XGBoost model can be integrated into a software package for rapid and accurate prediction of uplift load.

• In summary, while this study relied solely on the numerical analysis data for training ML models, it is important to recognize the limitations of the dataset, because the dataset has limited input features. For any other criteria this model will not perform. To further enhance the application of ML models in uplift load prediction, future research should consider incorporating experimental results and data with more variables.

REFERENCES

Bhattacharya, P., Kumar, J. (2016). “Uplift Capacity of Anchors in Layered Sand Using Finite-Element Limit Analysis: Formulation and Results”. International Journal of Geomechanics / Volume 16 Issue 3 - June 2016

Mistri, B. and Singh, B., (2011). “Pullout Behavior of Plate Anchors in Cohesive Soils”, Bund.K,EJGE, Volume 16, pp.1173-1184

Majumder, A., Roy, R., Banerjee. S., Mukherjee, S., Biswas, S., (2019) “Pullout behavior of plate anchors in geotextile reinforced soft clay”.

Singh, B., & Mistri, B., (2011) “A study on load capacity of horizontal and inclined plate anchors in sandy soils,” International Journal of Engineering Science and Technology (IJEST) Vol. 3 No. 9 September 2011.

Majumder, A., Roy, R., Banerjee. S., Mukherjee, S., Biswas, S., (2019) “Pullout behavior of plate anchors in geotextile reinforced soft clay”.

Biradar, J., Banerjee, S., Shankar, R., Ghosh, P., Mukherjee, S., Fatahi, B. (2019) “Response of square anchor plates embedded in reinforced soft clay subjected to cyclic loading”. Geomechanics and Engineering, Volume 17, Issue 2, Pages.165-173.

Thorne, C. P., Wang, C. X., Carter, J. P.,(2004), “Uplift Capacity of rapidly loaded strip anchors in uniform strength clay”, Geotechnique Volume 54, Issue 8, pp. 507-517.

Aydın, Y.; I¸sıkda ˘g, Ü.; Bekda¸s, G.; Nigdeli, S.M.; Geem, Z.W. Use of Machine Learning Techniques in Soil Classification. Sustainability 2023, 15, 2374.

Carvalho, L.O.; Ribeiro, D.B. A Multiple Model Machine Learning Approach for Soil Classification from Cone Penetration Test Data. Soils Rocks 2021, 44, 1–14.

Demir, S.; Sahin, E.K. An Investigation of Feature Selection Methods for Soil Liquefaction Prediction Based on Tree-Based Ensemble Algorithms Using AdaBoost, Gradient Boosting, and XGBoost. Neural Comput. Appl. 2023, 35, 3173–3190.

Samui, P.; Sitharam, T.G. Machine Learning Modelling for Predicting Soil Liquefaction Susceptibility. Nat. Hazards Earth Syst. Sci. 2011, 11, 1–9.

Wang, L.; Wu, C.; Tang, L.; Zhang, W.; Lacasse, S.; Liu, H.; Gao, L. Efficient Reliability Analysis of Earth Dam Slope Stability Using Extreme Gradient Boosting Method. Acta Geotech. 2020, 15, 3135–3150.

Zhang, W.; Zhang, R.; Wu, C.; Goh, A.T.C.; Wang, L. Assessment of Basal Heave Stability for Braced Excavations in Anisotropic Clay Using Extreme Gradient Boosting and Random Forest Regression. Undergr. Space 2022, 7, 233–241.

Xiao, L.; Zhang, Y.; Peng, G. Landslide Susceptibility Assessment Using Integrated Deep Learning Algorithm along the ChinaNepal Highway. Sensors 2018, 18, 4436.

Guido Van Rossum et al. Python programming language. In USENIX annual technical conference, volume 41, page 36, 2007.

Travis E Oliphant. A guide to NumPy, volume 1. Trelgol Publishing USA, 2006.

Wes McKinney. Pandas, python data analysis library. see http://pandas. pydata. org, 2015.

Niyazi Ari and Makhamadsulton Ustazhanov. Matplotlib in python. In 2014 11th International Conference on Electronics, Computer and Computation (ICECCO), pages 1–6. IEEE, 2014.

Raul Garreta and Guillermo Moncecchi. Learning scikit-learn: machine learning in python. Packt Publishing Ltd, 2013.

Seaborn documentation. https://seaborn.pydata.org/introduction. html). Accessed: 2020-04-26.

Cross validation documentation. https://towardsdatascience.com/ cross-validation-explained-evaluating-estimator-performance-e51e5430ff85). Accessed: 2020-04-28.

Sebastian Raschka. Model evaluation, model selection, and algorithm selection in machine learning. arXiv preprint arXiv:1811.12808, 2018.

Accuracy documentation. https://medium.com/thalus-ai/ performance-metrics-for-classification-problems-in-machine-learning-part-i-b085 Accessed: 2020-05-01.

scilearn max error. https://scikit-learn.org/stable/modules/model_ evaluation.html#max-error. Accessed: 2020-05-10.

scilearn mean absolute error. https://scikit-learn.org/stable/modules/ model_evaluation.html#mean-absolute-error. Accessed: 2020-05-10.