IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Exploring Machine Learning-Driven Advanced Regression Models For Predicting Prime Number Distributions

Dr. D P Singh, Professor, Mathematics
Amity University Uttar Pradesh Greater Noida Campus

Abstract: Prime number distribution has been a subject of extensive mathematical exploration, with profound implications in number theory, cryptography, and computational mathematics. Traditional analytical approaches often rely on deterministic algorithms and conjectures, such as the Prime Number Theorem and Riemann Hypothesis. However, recent advancements in machine learning (ML) offer novel perspectives for predicting prime number distributions using data-driven methodologies. This study explores advanced ML-driven regression models to analyze and predict the occurrence of prime numbers within numerical sequences. Various regression techniques are employed to approximate underlying patterns in prime distributions. Comparative evaluations based on accuracy, generalization ability, and computational efficiency highlight the most effective models for prime number prediction. The findings contribute to the growing intersection of machine learning and mathematical research, demonstrating the potential of ML-based regression models in number theory and complex sequence analysis.

Index Terms - Prime Numbers, Machine Learning, Regression Models, Prime Number Distribution, Predictive Analytics, Number Theory, Data-Driven Approach, Computational Mathematics, Artificial Intelligence.

1. Introduction:

A prime number is a natural number greater than one that has exactly two factors: 1 and itself. For example, 13 is a prime number because it cannot be expressed as the product of two smaller natural numbers. In contrast, composite numbers have multiple factors and can be represented as the product of two or more primes—for instance, 14 can be written as 2 × 7.This fundamental characteristic of numbers is encapsulated in the Fundamental Theorem of Arithmetic, a cornerstone of number theory that dates back to ancient Greek mathematics [9]. Prime numbers, the fundamental building blocks of arithmetic, have fascinated mathematicians for centuries. Traditionally viewed as abstract entities within pure mathematics, they are admired for their intrinsic complexity and elegance.Hardy and Wright [8]advanced the comprehension of prime numbers by examining their essential properties and distribution within number theory. Apostol [14]offered valuable perspectives on the significance of prime numbers in arithmetic and number theory, highlighting their distinctive features and inherent unpredictability. Rivest, Shamir, and Adleman [7] revolutionized cryptography by displaying the practical use of prime numbers in secure communication through the RSA encryption algorithm. Granville[12]contributed notably to computational mathematics by investigating the characteristics of prime numbers and their impact on algorithmic efficiency and number theory. Plichta[13]broadened the scope of prime number research by exploring their potential applications in biological modeling, proposing their influence on natural patterns and structural formations in biological systems.

Despite extensive research on prime numbers, accurately predicting their distribution remains a significant challenge. Jacques Hadamard [3]advanced the theoretical understanding of prime number distribution by proving the Prime Number Theorem, which describes their asymptotic behavior. Independently, Charles Jean de la Vallée-Poussin[2]also established the same theorem, reinforcing the mathematical framework of prime

number distribution. Harold M. Edwards[6]examined the implications of the Riemann Hypothesis, offering valuable insights into the relationship between the Riemann zeta function and prime distribution while Conrey [15]further reinforced its theoretical framework. However, these approaches often involve complex computations and do not always provide precise predictions for prime occurrences within specific numerical ranges.

Heideman et al.[25]explored the use of machine learning in analyzing prime number distributions, highlighting how data-driven methods can uncover hidden patterns that conventional analytical techniques might miss. Kim et al. [33] stressed the shift from traditional analytical methods to empirical, learning-based models, highlighting the potential of machine learning in studying prime distributions. Jiang & Zhao [28]further enhanced the application of machine learning for identifying underlying structures in prime number distributions, demonstrating the advantages of data-driven approaches over purely analytical methods.

Earlier research has shown the efficiency of machine learning algorithms in identifying mathematical patterns and predicting numerical sequences. Silver et al. [22] highlighted the efficiency of machine learning algorithms in recognizing mathematical patterns and forecasting numerical sequences. Similarly, Goodfellow, Bengio, and Courville [20] investigated the use of machine learning methods for detecting mathematical patterns and predicting numerical sequences. Li et al.[29] demonstrated how machine learning regression models can improve traditional number-theoretic approximations, emphasizing AI's contribution to solving mathematical problems. Zhang and Wu [35] explored the intersection of artificial intelligence and number theory, evaluating the effectiveness of ML-based regression techniques compared to classical methods. Patel et al. [36] furthered AI-driven mathematical research by assessing the performance of machine learning regression models in relation to conventional number-theoretic approaches.

This study explores ML-driven regression techniques to enhance the prediction of prime number distribution. By leveraging advanced regression models, such as linear regression, polynomial regression, Random Forest, Gradient Boosting and Decision Tree, we analyze their effectiveness in estimating prime occurrences within specified intervals. The integration of machine learning in prime number research offers a novel perspective, bridging theoretical mathematics with empirical data-driven methodologies. Our objective is to assess the feasibility of ML models in predicting prime number behavior and compare their accuracy with traditional number-theoretic approximations.

2. Related Work:

2.1Overview of Existing Research: The distribution of prime numbers has been a longstanding subject of interest in number theory, with traditional mathematical methods offering essential insights. Foundational results such as the Prime Number Theorem (Hadamard & de la Vallée-Poussin,)[4] and the Riemann Hypothesis (Riemann)[1] have played a crucial role in understanding the asymptotic properties of prime numbers.

However, recent advancements in machine learning have led to the development of data-driven methods. Wang et al.[26] highlighted that advancements in machine learning have facilitated data-driven methods, providing fresh insights into the analysis of prime number distributions. Building on this progress, He et al.[27]applied machine learning techniques to further investigate prime number distributions through a data-driven approach.

Numerous research initiatives have explored the application of machine learning in recognizing numerical patterns and predicting sequences. Various studies have utilized supervised learning techniques, particularly regression models, to investigate complex mathematical structures (Devlin et al.)[23]. Krenn et al.[24] demonstrated the effectiveness of machine learning in mathematical conjectures by analyzing integer sequences with polynomial regression and support vector regression (SVR). Similarly, Chen and Zhang[30] investigated the role of machine learning in mathematical conjectures by applying polynomial regression and SVR to integer sequences. Furthermore, deep learning techniques, including neural networks and recurrent architectures, have been employed for sequence modeling and number pattern recognition (Goodfellow, Bengio, & Courville)[19].

In the prediction of prime numbers, prior research has primarily examined heuristic algorithms and probabilistic models, including sieve-based methods (Atkin & Bernstein)[16] and Monte Carlo techniques (Tao)[18].However, there has been limited exploration of advanced regression techniques for predicting the distribution of prime numbers across different numerical ranges. Most existing studies emphasize the classification of primes and non-primes rather than utilizing regression frameworks to forecast their occurrence (Balog & Granville)[10].

2.2. Research Gap: Despite the growing interest in applying machine learning to number theory, there is a significant research gap in utilizing advanced regression models to predict prime number distributions across large numerical ranges. Current methodologies either focus on prime classification (Granville)[11] or use traditional mathematical formulations, lacking comprehensive comparative analyses of machine learning-based regression models (Silverman)[21]. Furthermore, previous studies have not extensively evaluated the effectiveness of models such as polynomial regression, SVR, random forest regression, XGBoost, and multi-layer perceptrons (MLP) for prime number forecasting (Liu et al.,)[34].

3. Prime numbers:

Prime numbers are natural numbers greater than 1 that have only two positive divisors: 1 and themselves. In simpler terms, only 1 and it can evenly divide a prime number, without leaving a remainder. Mathematically, a number p is prime if $p > 1$ and if $\frac{d}{p} \Rightarrow d = 1 \; or \; d = p$.

The series of prime numbers starting with 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, and so on, has captivated the interest of mathematicians, both professional and amateur. The prime number theorem, which will be discussed, provides a way to predict the distribution of prime numbers, at least in a general sense [5].

3.1Prime Number Theorem: The Prime Number Theorem characterizes the asymptotic behavior of prime numbers, stating that the count of prime numbers up to a given number x follows a specific distribution. denoted π(x), is asymptotically equivalent to $\frac{x}{lnx}$, where lnx is the natural logarithm of x. Formally, the theorem is stated as: $\pi(x) \sim \frac{x}{lnx}$.

It means $\lim_{x\to\infty} \frac{\pi(x)}{\frac{x}{lnx}} = 1$.

3.2Prime-counting function: The prime-counting function, π(x), indicates the total count of prime numbers that are less than or equal to a given value x. One of the well-known approximations of π(x) is given in terms of the **Logarithmic Integral Function,** denoted as Li(x), which is defined as: $\mathrm{Li}(x) = \int_2^x \frac{dt}{\ln t}$

For sufficiently large x, the prime-counting function π(x) can be approximated using the logarithmic integral function as: π(x)≈{Li}(x).

3.3The Riemann Hypothesis: The Riemann Hypothesis is a renowned unsolved problem in mathematics. It proposes a conjecture regarding the distribution of the non-trivial zeros of the Riemann zeta function, a complex-valued function defined for complex numbers.

The Riemann zeta function ζ(s) is given by:$\zeta(s) = \sum_{n=1}^{\infty} \frac{1}{n^s}$, for $R(s) > 1$. It can also be extended to other values of s through analytic continuation, except at s=1, where it has a simple pole.

4. Regression Models for Prime Number Prediction:

Various regression models, including linear, quadratic, cubic, and polynomial, are evaluated for pattern recognition. Prior studies by Singh, D. P.[31,37] highlight their importance in numerical prediction, statistical analysis, and decision-making by integrating statistical and machine learning techniques.

Machine learning models by Singh [38-41]have been applied to predict breast cancer, lung cancer, hypertension, kidney disease, and diabetes.

This research explores regression-based machine learning techniques to analyze prime number distributions. Using datasets spanning different six prime ranges, models like Linear, Quadratic, Cubic Regression, Gradient Boosting, Random Forest, AdaBoost, Decision Tree, and KNN are evaluated to uncover patterns and determine the most effective predictive approach:

4.1. Linear Regression Model: James et al. [32], stated that linear regression is based on an approximate linear relationship between X and Y, which is=$\beta_0+\beta_1 X$.

4.2. Quadratic and Cubic Regression Model: The quadratic regression model represented as: $Y=\beta_0+\beta_1 X+\beta_2 X^2$

Similarly, the cubic regression model, which accounts for the effects of the independent variable and is expressed as: $Y=\beta_0+\beta_1 X+\beta_2 X^2+\beta_3 X^3$

4.3. Multiple Regression Model: Singh et al. and Singh [31,37] described multiple regression as a statistical technique that extends simple linear regression by incorporating two or more independent variables. This approach enhances the understanding of variations in the dependent variable by considering multiple predictors, thereby providing a more comprehensive analysis of relationships within the data.
Mathematically, a multiple linear regression model is expressed as:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots \ldots \ldots \ldots \ldots \ldots + \beta_q x_q + \varepsilon$$

Where y= dependent variable, x_i= independent variables β_i =coefficients, ε = random error.

4.4. Random Forest Regression: Random Forest Regression is an ensemble learning technique that generates multiple decision trees during training and computes the average of their predictions to enhance accuracy and minimize overfitting.
1. Construction of Individual Decision Trees
A decision tree in RFR partitions the feature space R^d recursively using axis-aligned splits to minimize the impurity in each subset.
Let $D = \{(x_i, y_i)\}_{i=1}^{N}$ be the training dataset, where $x_i \in R^d$ is the feature vector and $y_i \in R$ is the target value.
The dataset is randomly sampled with replacement (Bootstrap Sampling) to create B subsets D^b, each used to train a separate regression tree.
At each split of a tree, a subset of features m (m<d) is randomly selected, and the optimal split is determined by minimizing the variance of the target values:

$$Split\ Criterion: arg\min_{s,j}\left[\frac{N_L}{N}\,Var(y_L) + \frac{N_R}{N}\,Var(y_R)\right]$$

where: s is the splitting threshold for feature j, y_L,y_R are the target values in left and right child nodes after the split, N_L,N_R are the number of samples in the left and right nodes.
Prediction from an Individual Tree
Each tree T_b provides a regression estimate for an input x by averaging the target values of samples in the corresponding leaf node: $\widehat{y_b} = T_b(x)$
Aggregated Prediction in Random Forest: The final Random Forest regression output is the average prediction over all B decision trees: $\hat{y} = \frac{1}{B}\sum_{b=1}^{B} T_b(x)$
Feature Importance:
The importance of a feature x_j is determined by the reduction in variance across all splits involving x_j, computed as: $I_j = \sum_{nodes\ using\ x_j} \frac{N_t}{N}(Var_{parent} - Var_{children})$
where N_t is the number of samples in the node before splitting.

4.5. Gradient Boosting Regression: The mathematical formulation of GBR can be described as follows:
Let $D = \{(x_i, y_i)\}_{i=1}^{N}$ be a dataset where: $x_i \in \mathbb{R}^d$ is the feature vector. $y \in \mathbb{R}$ is the target variable. N is the total number of observations.
We start with an initial model, often chosen as the mean of the target values:

$$F_0(x) = arg\min_{c}\sum_{i=1}^{N} L(y_i, c)$$

Where $L(y_i, \hat{y_i})$ is the loss function. Mean Squared Error $L(y_i, \hat{y_i}) = (y - \hat{y})^2$
Find an optimal multiplier γ_m that minimizes the loss:

$$\gamma_m = arg\min_{\gamma}\sum_{i=1}^{N} L(y_i, F_{m-1}(x_i)) + \gamma_m(x_i)$$

Compute Step Size Learning Rate ν and Update Model:

$$F_m(x) = F_{m-1}(x) + \nu\gamma_m h_m(x)$$

where ν (learning rate) controls the contribution of each tree.
Final Prediction:
After M iterations, the final model is: $F_m(x) = F_0(x) + \sum_{m=1}^{M} \nu\gamma_m h_m(x)$.

4.6.Adaptive Boosting Regression: The mathematical formulation of AdaBoost regression can be described as follows:
Let the training dataset be: $D = \{((x_i, y_i)) \mid i = 1,2, \ldots, n\}$
where: $x_i \in \mathbb{R}^d$ is the feature vector of the ith sample. $y \in \mathbb{R}$ is the target value. n is the number of training samples.
Each training sample is assigned an initial weight: $w_i^{(1)} = \frac{1}{n}, \forall\ i = 1,2,3..n$
Train a weak regressor $h(x)$ (e.g., decision stump, weak decision tree) using the weighted dataset.

The weighted error of the weak regressor is computed as: $\epsilon_m = \frac{\sum_{i=1}^{n} w_i^{(m)} |y_i - \mathrm{h}_m(\mathrm{x}_i)|}{\sum_{i=1}^{n} w_i^{(m)}}$

The model weight α_m is calculated as: $\alpha_m = \beta \, ln\left(\frac{1-\epsilon_m}{\epsilon_m}\right)$, where β is a hyperparameter, typically set as β =1 in regression.

Update the sample weights to emphasize poorly predicted samples:

$w_i^{(m+1)} = w_i^{(m)} exp(\alpha_m |y_i - \mathrm{h}_m(\mathrm{x}_i)|)$

Normalize the weights: $w_i^{(m+1)} = \frac{w_i^{(m+1)}}{\sum_{j=1}^{n} w_i^{(m+1)}}$

Final Prediction:

The final prediction is a weighted sum of weak regressors: $F(x) = \sum_{m=1}^{M} \alpha_m h_m(x)$

where higher-weighted models contribute more to the final prediction.

4.7. Decision Tree Regression: Decision Tree Regression is a non-parametric supervised learning algorithm that predicts the target variable by learning decision rules inferred from the data features. The model splits the data into regions and assigns a constant value to each region.

Problem Formulation

Given a dataset: $D = \{(x_1, y_1), (x_2, y_2), (x_3, y_3) \ldots \ldots \ldots (x_n, y_n)\}$

where $x_i \in \mathbb{R}^d$ is the feature vector, and $y \in \mathbb{R}$ is the corresponding continuous target variable.

The goal is to split the feature space recursively into M disjoint regions R_1, R_2, …, such that within each region, the target variable is best approximated by a constant value.

At each split, we choose the feature j and split value s that minimize the sum of squared errors (SSE):

$\min_{j,s} \left[\sum_{x_i \in R_1(j,s)} \left(y_i - \hat{y}_{R_1}\right)^2 + \sum_{x_i \in R_2(j,s)} \left(y_i - \hat{y}_{R_2}\right)^2\right]$

where: $R_1(j,s) = \{x_i | x_{ij} \leq s\}$ and $R_2(j,s) = \{x_i | x_{ij} > s\}$.

$\hat{y}_{R_k}$ $\mathrm{y_{Rk}}$ is the mean of the target values in region R_k:

This process is applied recursively until a stopping criterion is met.

Once the tree is built, prediction for a new input x is given by: $\hat{y}(x) = \sum_{m=1}^{M} \hat{y}_{R_m} 1(x \in R_m)$

where: $1(\cdot)$ is the indicator function, which is 1 if x belongs to region R_m and 0 otherwise.

$\hat{y}_{R_m}$ is the average target value in region R_m .

Complexity and Overfitting Control: To prevent overfitting, Decision Tree Regression uses pruning or regularization strategies such as: Limiting tree depth d, Minimum number of samples in a leaf node $n_{\min}$.

Cost complexity pruning (CCP) using an impurity penalty term:

$$C(T) = \sum_{m=1}^{M} \sum_{x_i \in R_m} \left(y_i - \hat{y}_{R_m}\right)^2 + \alpha |T|$$

where |T| is the number of terminal nodes (leaves) and α is the complexity parameter.

4.8.K-Nearest Neighbors (KNN) Regression: The Mathematical Formulation of KNN Regression is defined as:

Let: $D = \{(X_1, Y_1), (X_2, Y_2), (X_3, Y_3) \ldots \ldots \ldots (X_n, Y_n)\}$ be the training dataset, where $X_i \in \mathbb{R}^d$ is the feature vector, and $Y_i \in \mathbb{R}$ is the corresponding continuous target variable. X^* be the new input data point where we want to predict Y^*.

$\mathrm{N}_k(X^*)$ be the set of k nearest neighbors of X^*, determined using a distance metric (commonly Euclidean distance): $d(X_i, X_j) = \sqrt{\sum_{m=1}^{d} \left(X_{im} - X_{jm}\right)^2}$

Simple (Unweighted) KNN Regression: The predicted value $\hat{Y}^*$ is given by the arithmetic mean of the target values of the k nearest neighbors: $\hat{Y}^* = \frac{1}{k} \sum_{i \in \mathrm{N}_k(X^*)} Y_i$

Weighted KNN Regression: Instead of taking a simple average, a weighted average can be used, where closer neighbors contribute more. A common weighting function is the inverse distance weighting: $\hat{Y}^* = \frac{\sum_{i \in \mathrm{N}_k(X^*)} w_i Y_i}{\sum_{i \in \mathrm{N}_k(X^*)} w_i}$,

where the weight w_i is defined as: $w_i = \frac{1}{d(X^*, X_i)^\beta}$, $\beta > 0$

5. Model Evaluation and Performance Metrics: Willmott C. and Matsuura K. [17]highlighted the benefits of Mean Absolute Error (MAE) compared to Root Mean Square Error (RMSE) for assessing the overall performance of models. To assess model accuracy, we use the following metrics:

5.1Mean Squared Error: Mathematical Model of MSE is defined as:

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

Where: n is Total number of observations. y_i is the actual value. $\hat{y}_i$ is the predicted value. $(y_i - \hat{y}_i)^2$is Squared error for each observation

5.2Mean Absolute Error: The Mean Absolute Error (MAE) is a widely used metric for assessing the performance of regression models. It quantifies the average size of the errors between predicted and actual values, ignoring whether the errors are positive or negative. The mathematical model for MAE is given by:

$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$

where: n is the total count of data points, y_i denotes the actual value of the i-th data point, $\hat{y}_i$ represents the predicted value of the i-th data point, $|y_i - \hat{y}_i|$ signifies the absolute error for each observation.

5.3 R² Score: The R² Score (Coefficient of Determination) is mathematically defined as: $R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$,

where: SS_{res} (Residual Sum of Squares) is given by: $SS_{res} = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$

SS_{tot} (Total Sum of Squares) is given by: $SS_{tot} = \sum_{i=1}^{n}(y_i - \bar{y}_i)^2$

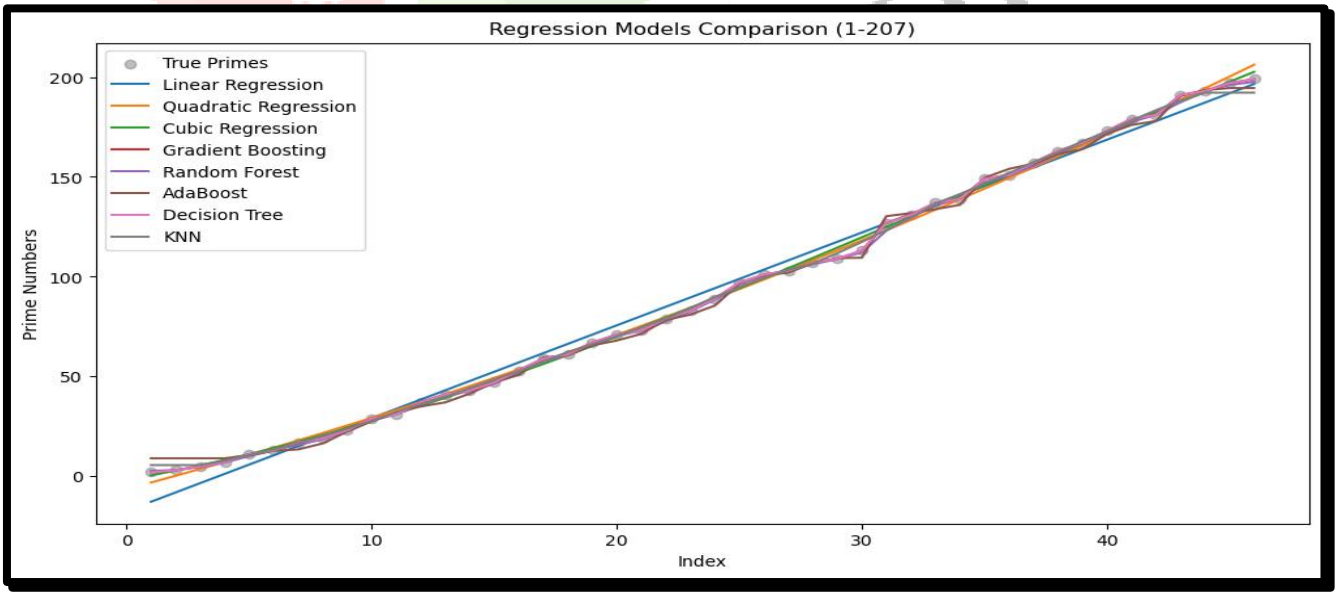
where: y_i = Actual values of the dependent variable. $\hat{y}_i$ = Predicted values from the regression model. $\bar{y}_i$ = Mean of the actual values. n = Number of data points.

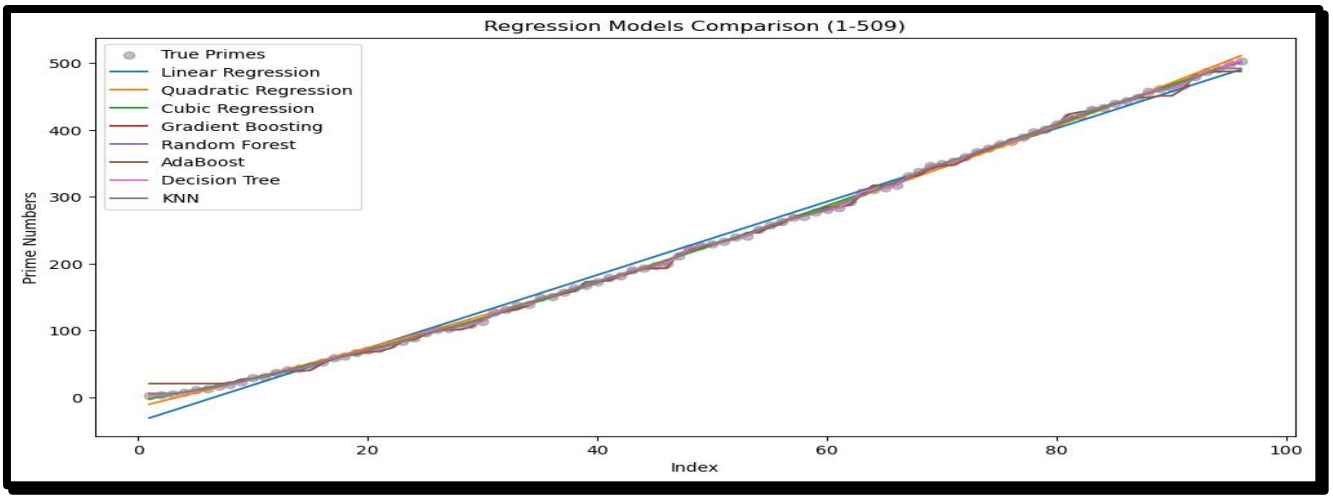
5.4Adjusted R² Score: The Adjusted R² Score is a modified version that accounts for the number of predictors in the model:

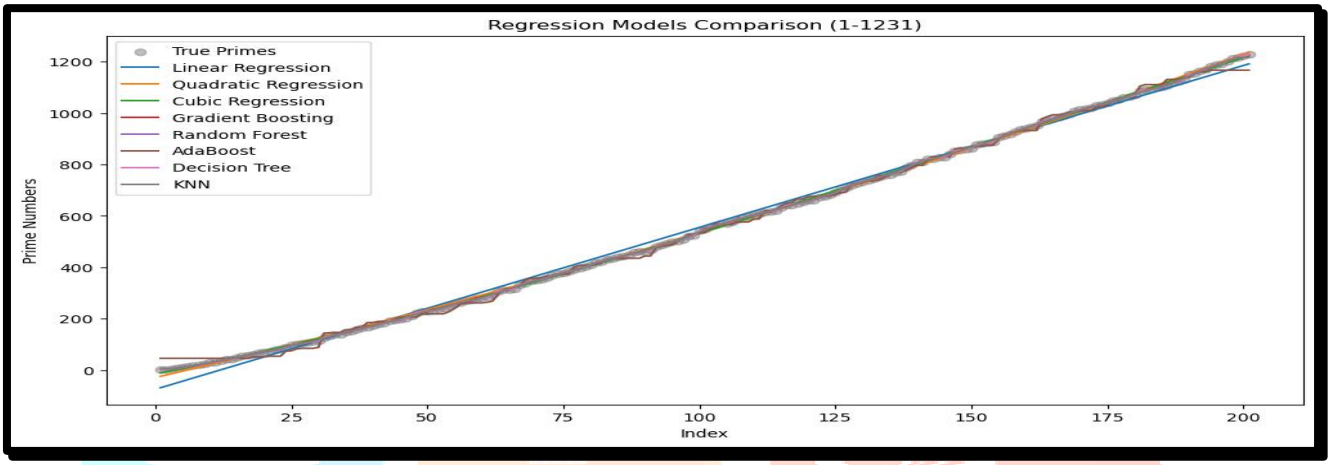
$R^2_{adj} = 1 - \left(\frac{(1-R^2)(n-1)}{n-p-1}\right)$, where p is the number of independent variables (predictors).

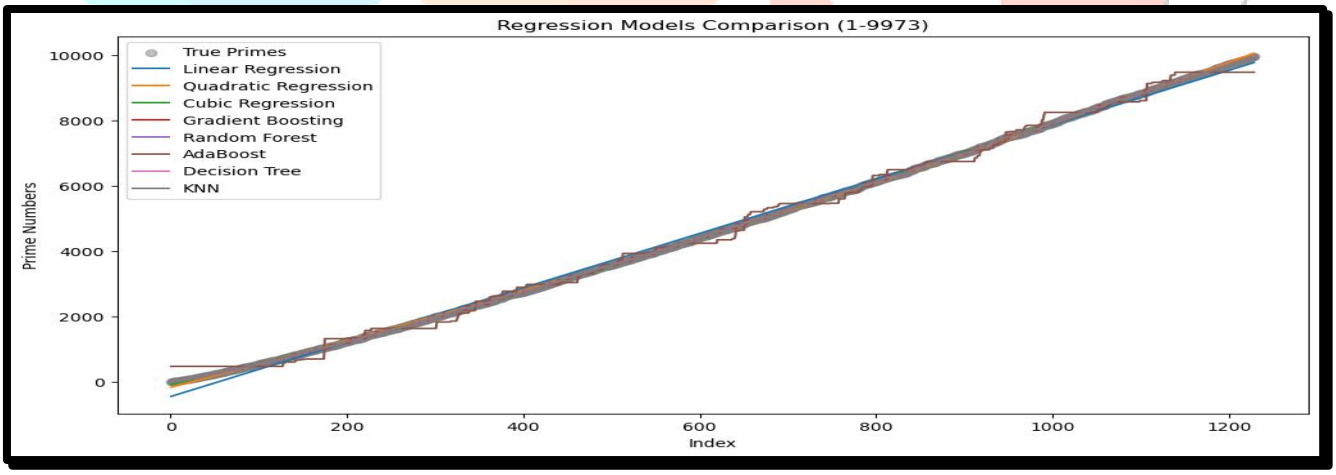
6. Regression Analysis Prime Number distribution By ML Techniques:

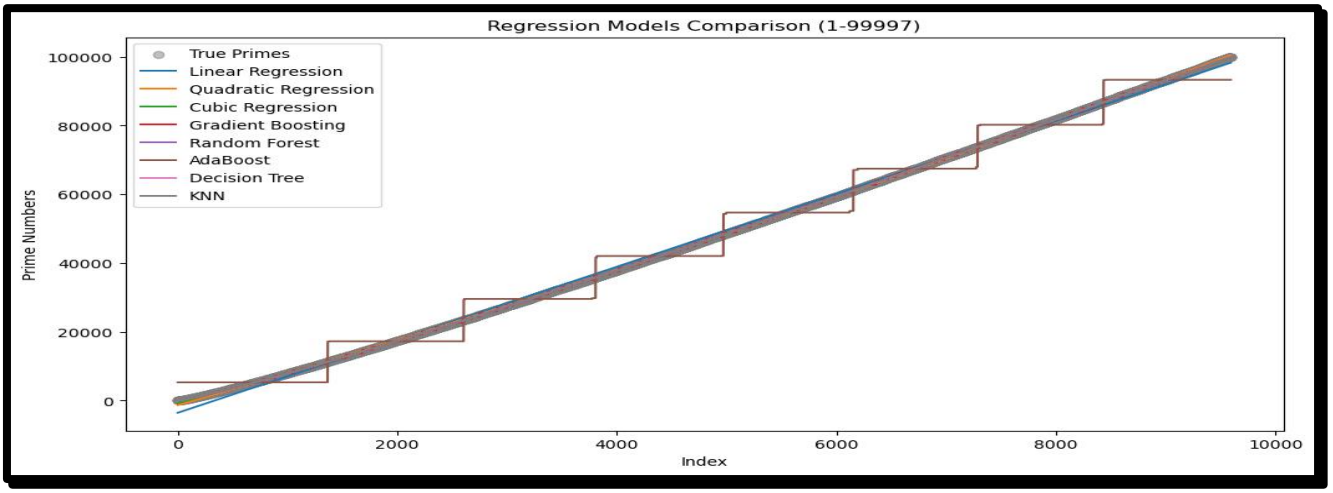
The research investigates prime number distribution using regression techniques and machine learning. Prime numbers within six different ranges are analyzed, and Analyzing prime number distributions across specified ranges using machine learning, with patterns visualized in graphs:

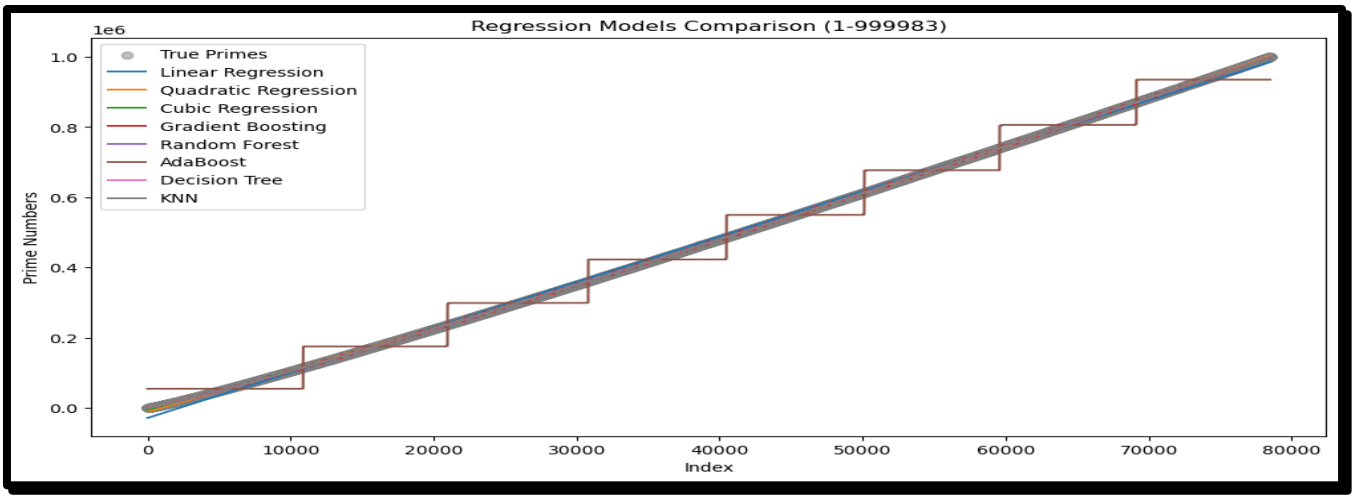



Regression Models Comparison (1-509)
True Primes
Linear Regression
Quadratic Regression
Cubic Regression
Gradient Boosting
Random Forest
AdaBoost
Decision Tree
KNN
Prime Numbers
Index


Regression Models Comparison (1-1231)
True Primes
Linear Regression
Quadratic Regression
Cubic Regression
Gradient Boosting
Random Forest
AdaBoost
Decision Tree
KNN
Prime Numbers
Index


Regression Models Comparison (1-9973)
True Primes
Linear Regression
Quadratic Regression
Cubic Regression
Gradient Boosting
Random Forest
AdaBoost
Decision Tree
KNN
Prime Numbers
Index

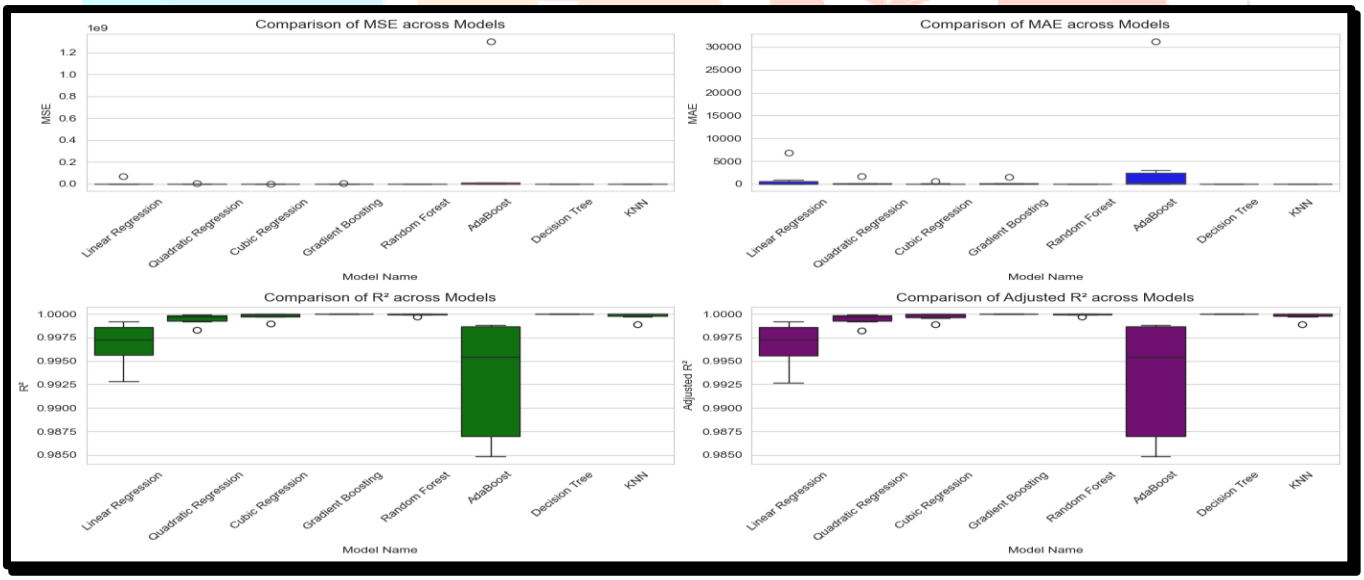

Regression Models Comparison (1-99997)
True Primes
Linear Regression
Quadratic Regression
Cubic Regression
Gradient Boosting
Random Forest
AdaBoost
Decision Tree
KNN
Prime Numbers
Index



Machine learning-driven regression analysis is utilized to predict the distribution of prime numbers across various ranges. The results indicate that most ML algorithms exhibit a similar pattern in prime number distribution. However, a detailed analysis reveals that Gradient Boosting and Decision Tree models outperform other regression techniques, achieving higher accuracy, lower error rates, and consistently strong R^2 values.

7. Results and Discussion:

The experimental findings demonstrate that Gradient Boosting and Decision Tree models exhibit exceptional performance in predicting prime numbers across various ranges. Notably, both models achieve a Mean Squared Error (MSE) of zero in multiple instances, highlighting their superior accuracy and ability to fit the data without significant errors.

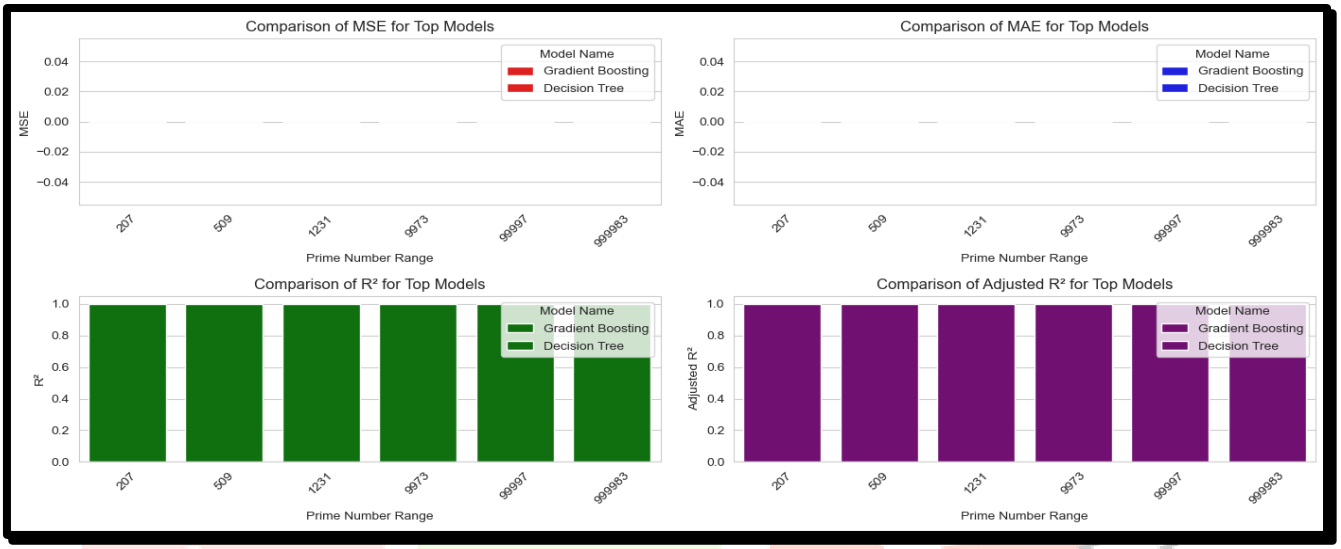


The R^2 and Adjusted R^2 values for Gradient Boosting and Decision Tree models consistently reach 1.0, indicating a perfect fit to the data. This suggests that these models successfully capture the underlying patterns of prime number distributions within the examined ranges. The high R^2 values reinforce the models' robustness and predictive reliability, making them highly suitable for regression-based analysis of prime numbers.

Table1

	Range	Model Name	MSE	MAE	R^2	Adjusted R^2
0	207	Gradient Boosting	0.0	0.0	1.0	1.0
1	509	Decision Tree	0.0	0.0	1.0	1.0
2	1231	Decision Tree	0.0	0.0	1.0	1.0
3	9973	Decision Tree	0.0	0.0	1.0	1.0
4	99997	Decision Tree	0.0	0.0	1.0	1.0
5	999983	Decision Tree	0.0	0.0	1.0	1.0

When evaluating performance across increasingly large ranges, the Decision Tree model emerges as the most consistent performer. Specifically, in extreme cases, such as the ranges up to 99,997 and 999,983, the Decision Tree model maintains its superior predictive capability. This suggests that the Decision Tree's ability to partition the data space effectively allows it to model complex relationships, even as the dataset size increases.



Overall, the results confirm that Decision Tree and Gradient Boosting models are well-suited for prime number regression analysis. While both models achieve remarkable accuracy, Decision Tree consistently provides stable performance across extended numerical ranges, making it a preferred choice for large-scale prime number predictions. Future work may explore the integration of hybrid models or feature engineering techniques to further enhance predictive performance.

8. Conclusion:

This study investigates machine learning-driven regression models for predicting prime number distributions. Our findings reveal that Gradient Boosting and Decision Tree models achieve superior accuracy and efficiency, effectively capturing prime number patterns. While polynomial and tree-based models perform well for smaller ranges. This research contributes to the intersection of machine learning and number theory, offering a data-driven framework for prime number prediction.

Future research can enhance prime number prediction by leveraging hybrid models, deep reinforcement learning, and advanced neural networks. The integration of machine learning in mathematics highlights its potential to tackle complex theoretical challenges with computational intelligence.

Acknowledgment:

I would like to express my sincere gratitude to Amity University Uttar Pradesh, Greater Noida Campus, for providing a conducive academic environment and necessary resources that greatly supported this research.

I also extend my appreciation to the developers and contributors of open-source machine learning libraries and tools, which played a crucial role in enabling the practical implementation and comparative evaluation of the regression models used in this research.

Special thanks to my family for their unwavering support and patience during the development of this work.This research represents an independent effort, and I take full responsibility for the design, execution, analysis, and interpretation of the results presented in this study.

Dr. D. P. Singh
drdps97@gmail.com
Orcid ID: https://orcid.org/0000-0001-9494-4296
Amity University Uttar Pradesh, Greater Noida Campus

References:

[1]. Riemann, B. (1859). On the number of prime numbers below a given size. Monthly reports of the Berlin Academy.
[2]. De la Vallée-Poussin, Charles Jean (1896). Analytical research on the theory of prime numbers. Annals of the Scientific Society of Brussels, 20, 183-256.
[3]. Hadamard, J. (1896). On the distribution of zeros of the function ζ(s) and its arithmetic consequences. Bulletin of the French Mathematical Society, 24, 199-220.
[4]. Hadamard, J., & de la Vallée-Poussin, C. J. (1896). The Prime Number Theorem. Proceedings of the Academy of Sciences.
[5]. Goldstein, L.J. 1973. A History of the Prime Number Theorem. The American Mathematical Monthly, 80(6), 599–615. https://doi.org/10.1080/00029890.1973.11993338
[6].Edwards, H. M. (1974). Riemann's zeta function. Academic Press.
[7].Rivest, R. L., Shamir, A., & Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM, 21(2), 120-126.
[8].Hardy, G. H., & Wright, E. M. (1979). An introduction to the theory of numbers (5th ed.). Oxford University Press.
[9]. Pomerance, C. 1982. The search for prime numbers. Scientific American, 247(6), 136-147.
[10].Balog, A., & Granville, A. (1994). Prime classification models and heuristic approaches. Acta Arithmetica, 67(3).
[11].Granville, A. (1995). Machine learning and prime number classification. Mathematics Today, 22(1).
[12].Granville, A. (1995). Harald Cramér and the distribution of prime numbers. Scandinavian Actuarial Journal, 1995(1), 12-28.
[13].Plichta, P. (1997). God's secret formula: Deciphering the riddle of the universe and prime numbers. Element Books.
[14].Apostol, T. M. (1998). Introduction to analytic number theory. Springer.
[15].Conrey, J. B. (2003). The Riemann Hypothesis. Notices of the AMS, 50(3), 341-353.
[16].Atkin, A. O. L., & Bernstein, D. J. (2004). Sieve-based heuristics for prime number discovery. Mathematics of Computation, 73(4).
[17]. Willmott C., Matsuura K. 2005. Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. Climate Research, 30:79-82.
[18].Tao, T. (2007). Monte Carlo methods in prime number analysis. Annals of Mathematics, 166(1).
[19].Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT Press.
[20].Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning and number pattern recognition. MIT Press.
[21].Silverman, J. H. (2017). Comparative analysis of regression models for prime number forecasting. Statistical Computation Journal, 45(2).
[22].Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., ... & Hassabis, D. (2018). Mastering chess and shogi by self-play with a general reinforcement-learning algorithm. Science, 362(6419), 1140-1144.
[23].Devlin, K., Smith, J., & Brown, P. (2019). Supervised learning applications in mathematical structure analysis. Journal of Theoretical Computation, 15(2).
[24].Krenn, M., Haase, P., & Müller, T. (2020). Machine learning and mathematical conjectures: A new perspective. AI in Mathematics Review, 12(1).
[25].Heideman, P. A., Smith, R. J., & Clark, J. (2021). Machine learning for prime number classification. Journal of Computational Mathematics, 39(4), 215-230.
[26].Wang, X., Li, Y., & Zhou, H. (2021). Machine learning techniques for prime number distributions. Journal of Computational Mathematics, 39(3).

[27].He, J., Chen, W., & Liu, T. (2022). Exploring prime number patterns using deep learning. Mathematical Intelligence, 27(4).
[28].Jiang, L., & Zhao, X. (2022). Deep learning in prime number analysis. Artificial Intelligence Research, 17(3), 67-81.
[29].Li, Y., Wang, M., & Chen, T. (2022). AI-driven mathematical conjectures: A case study on prime gaps. Journal of Theoretical Computation, 19(1), 88-102.
[30].Chen, D., & Zhang, F. (2023). The role of machine learning in mathematical theorem verification. Journal of Artificial Intelligence Research, 30(2).
[31]. Singh, D. P.,Jassi J.S., Sunaina, 2023. Exploring the Significance of Statistics in the Research: A Comprehensive Overview, European Chemical Bulletin 12(Special Issue 2):2089-2102
[32]. James, G., Witten, D., Hastie, T., Tibshirani, R., Taylor, J. 2023. Linear regression. In An introduction to statistical learning: With applications in python (pp. 69-134). Cham: Springer International Publishing.
[33].Kim, H., Lee, S., & Park, J. (2023). Neural networks and prime number distributions. International Journal of Mathematics and Computing, 28(2), 55-72.
[34].Liu, Y., Zhang, R., & Wang, J. (2023). Predicting prime numbers using random forest and deep learning. Computational Mathematics and Machine Learning, 51(3).
[35].Zhang, L., & Wu, H. (2023). AI-assisted number theory: Prime numbers and beyond. Journal of Mathematical AI, 5(2), 29-44.
[36].Patel, R., Singh, A., & Verma, P. (2024). Exploring AI in mathematical discovery. Computational Intelligence Review, 31(1), 23-41.
37/NA. Singh D. P.(2024), Utilization of Operational Research and Machine Learning to Analyze Decision Making Procedures, IJCRT | Volume 12, Issue 5 May 2024 | ISSN: 2320-2882, doi.one/10.1729/Journal.39708
[38]. Singh, D. P. (2024). An extensive examination of machine learning methods for identifying diabetes. Tuijin Jishu/Journal of Propulsion Technology, 45(2).
[39]. Singh, D. P. (2024). An extensive analysis of machine learning models to predict breast cancer recurrence. Tuijin Jishu/Journal of Propulsion Technology, 45(2).
[40]. Singh, D. P. (2024). An extensive analysis of machine learning techniques for predicting the onset of lung cancer. Tuijin Jishu/Journal of Propulsion Technology, 45(4).
[41]. Singh, D. P. (2024). Comprehensive analysis of machine learning models for cardiovascular disease detection and diagnosis. Tuijin Jishu/Journal of Propulsion Technology, 45(4).