CRICKET PREDICTION

PROJECT REPORT

Submitted by

Sourabh Kumrawat- 22BSA10112

Aarchi Thakkar- 22BSA10102

Navika Mehrotra – 22BSA10197

Aditya lokhande- 22BSA10297

Shashank singh – 22BSA10136

IN A PARTIAL FULFILLMENT FOR THE AWARD OF THE DEGREE OF

Bachelor of Technology

In

Computer Science and Engineering

(Cloud Computing and Automation)

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

VIT BHOPAL UNIVERSITY

KOTHRIKALAN, SEHORE

MADHYA PRADESH – 466114

OCTOBER 2023

VIT BHOPAL UNIVERSITY, KOTHRIKALAN, SEHORE

MADHYA PRADESH – 466114

BONAFIDE CERTIFICATE

Certified that this project report titled "**CRICKET PREDICTION**" is the bonafide work of **"Sourabh Kumrawat(22BSA10112)Aarchi Thakkar(22BSA10102)Navika Mehrotra(22BSA10197)Aditya lokhande(22BSA10297)Shashank singh(22BSA10136)**who carried out the project work under my supervision. Certified further that to the best of my knowledge the work reported at this time does not form part of any other project/researchwork based on which a degree or award was conferred on an earlier occasion on this or any other candidate.

Dr. Virendra Singh Kushwah (Program Chair)
PROGRAM CHAIR

School of Computing Science and Engineering

VIT BHOPAL UNIVERSITY

Dr. Virendra Singh Kushwah
PROJECT GUIDE

School of Computing Science and Engineering

VIT BHOPAL UNIVERSITY

The Project Exhibition I Examination is held on_____

ACKNOWLEDGEMENT

First and foremost, I would like to thank the Lord Almighty for His presence and immense blessings throughout the project work.

We wish to express my heartfelt gratitude to Dr. Virendra Singh Kushwah, Head of the Department, School

of Computing Science and Engineering for much of his valuable support encouragement in carrying out this work.

We would like to thank my internal guide Mr./Ms. Dr. Virendra Singh Kushwah ,for continually guiding andactively participating in my project, giving valuable suggestions to complete the project work.

We would like to thank all the technical and teaching staff of the School of Computer Science, who extended directly or indirectly all support.

Last, but not least, I am deeply indebted to my parents who have been the greatest support while I worked day and night for the project to make it a success.

LIST OF ABBREVIATIONS

S.NO	ABBREVIATION	FULL FORM
1	PYTHON	PYTHON
2	ML	MACHINE LEARNING
3	AI	Artificial Intelligence
6	BH	BACK HAND
5	E2E	End to End Testing
6	GHz	Gigahertz
7	GB	Gigabyte

LIST OF FIGURES AND GRAPHS

ABSTRACT

India's most popular sport is cricket, which is played in every area in a variety of formats including T20, ODI, and Test. In the Indian Premier League (IPL), a domestic cricket league, players represent Indian regional teams, the national team, and international teams. A number of things contributed to the popularity of this league among fans of cricket, such as live streaming and TV, radio, and broadcasts. The outcome forecasts of the Indian Premier League are quite important for online traders and sponsors. Cricket is the most popular sport in India, and it is played there in all of its regions in various formats such as T20, ODI, and Test. In addition to more traditional variables like the toss, venue, and day-night, we can predict a match between two teams based on a variety of characteristics like the team's composition, the batting and bowling averages of each player on the team, and the team's success in prior matches. At a given match site, the Indian Premier determines the probability of winning by batting first against a particular team. Through the use of machine learning techniques including SVM, Random Forest Classifier (RFC), and Logistic Regression, we have proposed a model in this study for predicting the outcomes of IPL matches.

Experimental results show that the Random Forest algorithm outperforms other algorithms with an accuracy of 88.10%.

Keywords: Lasso Regression, Naïve Bayes, Logistic Regression, Random Forest Classifier, Cricket Prediction, Cricket Analysis.

TABLE OF CONTENTS

CHAPTER 1

PROJECT DESCRIPTION AND OUTLINE

INTRODUCTION

Cricket is the most popular game after football. England has been the home of the sport since the fifteenth century. Cricket may soon overtake football as the sport with the largest fan base due to its increasing global fan base. In India today, it's more than just a game—it's a religion. Three primary formats exist. A One Day International match consists of 50 overs played in a single day. The second format, which is the test format, is what the game was originally designed to use. Every team plays two innings, with each innings consisting of 80–90 overs, over a period of five days.The group has five days to provide consistent delivery.A player's endurance, strength, patience, and mental toughness are essential in this incredibly challenging game type. The third and most recent version of cricket is called Twenty20. India took home the first world championship with this style in 2007, having developed it in 2006. The twenty-over short game is completed in less than three hours. There are just two teams playing, and each team has twenty overs to play. The IPL has greatly contributed to the t20 format's popularity in India. This event led to the t20 format's rise in popularity in India.As IPL spectators, we make our own guesses about a particular match. They make these forecasts based on the information at their disposal and calculate the winner using a variety of records and statistics. More importantly, there is a sizable market for algorithms that predict the winning team and best score. For every IPL match that has already been played, we will make predictions. In order to predict the results of the matches, machine learning techniques are employed.

1.1 MOTIVATION FOR WORK

The motivation for undertaking the project, "Cricket Outcome Prediction Using Machine Learning: A Random Forest Classifier Approach with Python's Pandas and NumPy," can be attributed to several key factors:

Interest in Cricket: A fundamental passion for the sport of cricket serves as a primary motivator. Enthusiasm

for cricket fosters a genuine interest in exploring ways to enhance understanding and engagement with the game.

Data-Driven Decision-Making: In the modern era, data-driven decision-making is increasingly important in sports. The motivation behind this project lies in the recognition that data analytics and machine learning can provide valuable insights for cricket teams, analysts, and fans, thereby improving strategic decisions and predictions.

Accuracy and Predictive Power: The Random Forest Classifier is known for its high accuracy in predicting outcomes. The motivation here is to harness this predictive power to create a reliable tool for forecasting cricket match results, contributing to a better understanding of the sport's dynamics.

Practical Application: The practical utility of machine learning in cricket is a significant motivator. The project aims to demonstrate how these techniques can be directly applied to the real-world context of cricket, potentially benefiting teams, sports analysts, and even betting enthusiasts.

Educational Purpose: The project serves an educational purpose by providing a valuable resource for individuals interested in learning about data science and machine learning. It can serve as a stepping stone for those seeking to apply these techniques to sports analytics.

Enhancing Fan Engagement: Cricket is a popular sport with a large fan base. The project aims to enhance fan engagement by offering insights and predictions for upcoming matches, potentially increasing interest and viewership in the sport.

Competitive Advantage: Cricket teams and organizations are continually seeking ways to gain a competitive advantage. This project's motivation lies in its potential to provide teams with data-driven insights to optimize performance and strategic decisions.

Future Research and Development: The motivation extends to the potential for ongoing research and development in the field of sports analytics. This project can serve as a foundation for further investigations into more sophisticated modeling techniques and the inclusion of additional data sources.

In conclusion, the motivation for this project stems from a genuine love for cricket, a recognition of the power of data-driven insights, and a desire to contribute to the sport's development and engagement. It combines a

passion for cricket with the potential of data science and machine learning to provide valuable solutions for the cricket community and sports enthusiasts

1.2 INTRODUCTION TO TECHNIQUE USED

In the realm of scientific and technical research, the dissemination of knowledge is a fundamental process. The impact of research is not solely determined by the novelty of the findings but also by the effectiveness with which those findings are communicated to the wider scientific community. IEEE Transactions and Journals, known for their rigor and influence, are at the forefront of this communication process. Researchers worldwide turn to IEEE publications as a reliable source of cutting-edge knowledge in the fields of electrical engineering, computer science, and other related disciplines.

The motivation for this work stems from the growing importance of publishing research in top-tier journals such as those offered by the Institute of Electrical and Electronics Engineers (IEEE). These journals serve as crucial channels for sharing breakthroughs, theories, and discoveries with a global audience. Effective preparation and submission of articles to IEEE Transactions and Journals, however, present unique challenges that researchers must navigate. The scholarly writing and submission process necessitates a nuanced understanding of the format, style, and editorial requirements set forth by IEEE.

Techniques we used in this project are:

Module of python which are

1) **numpy**
2) **pandas**

Machine learning algorithms from sklearn which are

1) **RandomForestclassifier**
2) **Svm**
3) **GradientBoosterClassifier**
4) **kNeighborsClassifier**
5) **gaussianNB**
6) **Regression**

1.3 PROBLEM STATEMENT

- The process of cricket prediction involves using machine learning algorithms to analyze a wide range of factors that can influence match outcomes. These factors include team performance, player statistics, pitch conditions, and even weather data. By leveraging historical data and applying predictive models, cricket prediction aims to provide insights into the likelihood of different outcomes in cricket matches.

- One of the key benefits of cricket prediction is enhanced decision-making. Teams, coaches, and even fans can make more informed decisions based on data-driven insights. This can range from selecting the best playing XI to making strategic decisions during a match. Cricket prediction also plays a significant role in fantasy cricket. Users can participate in fantasy cricket leagues where they create virtual teams and earn points based on the performance of real players in real matches. Predictive models help users make informed choices when selecting players for their fantasy teams.

1.4 OBJECTIVE OF THE WORK

the model can continuously analyze the match data, including the current score, player performance, and match conditions. Based on this real-time analysis, the model can provide updated predictions on the match outcome, helping fans and enthusiasts make informed decisions. Additionally, the project can be integrated into cricket fantasy platforms, where users can create their virtual teams and compete against each other. The prediction model can assist users in selecting the best players for their teams

1.5 ORGANIZATION OF THE WORK

Organizing the work for developing a ML PROJECT OF CRICKET PREDICTION Involves a systematic approach to manage the project effectively. Here's asuggested organization of the work:

1. Project Planning:

- Define project goals and objectives.

- Identify the target audience and user personas.

- Create a project timeline with milestones and deadlines.

- Allocate resources, including team members and budget.

- Create user flows to ensure a seamless user experience.

- Design a responsive and intuitive interface for various devices.

2. Technical Architecture:

- Plan the architecture, including database structure and server setup.

- Address scalability and performance considerations.

1.6 SUMMARY

The project, titled "Cricket Outcome Prediction Using Machine Learning: A Random Forest Classifier Approach with Python's Pandas and NumPy," presents an in-depth exploration of the application of machine learning techniques to predict cricket match outcomes. This initiative harnesses the power of the Random Forest Classifier algorithm, along with Python's Pandas and NumPy libraries, to provide data-driven insights and predictions for cricket matches.

The project encompasses various critical phases, including data collection, data preprocessing, feature engineering, model development, model evaluation, and practical deployment. Leveraging comprehensive cricket match datasets, the project ensures data quality and comprehensiveness, and it employs Pandas and NumPy for data cleaning, feature extraction, and normalization.

The Random Forest Classifier is chosen for its proven accuracy in prediction tasks, and its implementation is optimized through hyperparameter tuning and cross-validation techniques. Model performance is rigorously evaluated using a variety of metrics, emphasizing its practical utility and accuracy in cricket outcome prediction.

One of the key strengths of this project is its real-world applicability. The model's predictions can be used to support decision-making in cricket, offering insights for teams, analysts, and enthusiasts. The project also highlights the versatility of the Random Forest Classifier and the data manipulation capabilities of Pandas and NumPy, making it adaptable to various sports and domains.

Additionally, the project serves as an educational resource, providing a foundation for those interested in the

field of data science and machine learning, particularly in the context of sports analytics. It showcases the potential for data-driven decision-making to enhance fan engagement, competitive advantage for cricket teams, and overall interest in the sport.

In conclusion, this project not only delivers accurate cricket outcome predictions but also demonstrates the practicality of machine learning in the realm of sports analytics. It sets the stage for ongoing research and offers valuable insights for the cricket community and beyond, making it a promising endeavor in the field of sports data science.

.

CHAPTER 2

RELATED WORK INVESTIGATION

2.1 INTRODUCTION

I came into a comprehensive study article titled "Cricket Match Outcome Prediction Using Machine Learning Techniques" while conducting research for a similar project. Using a variety of machine learning methods, including support vector machines, logistic regression, and decision trees, this study explored the field of cricket prediction. To train and assess their prediction algorithms, the researchers collected a tonne of historical match data, which included team performances, individual statistics, and match circumstances. The study's conclusions showed encouraging accuracy in forecasting match results. This study might be a useful addition to your cricket prediction presentation. Wishing you success on your project!

2.2 EXISTING APPROACHES/METHODS

The "CulinaryCanvas" project incorporates various approaches andmethods to achieve its objectives. Here are some of the existing approaches and methods used in the project:

2.3 PROS AND CONS

PROS:

- We provide proper data for the players

-We provide each team data properly

- We have quite number of choices.

-You can make a vast collection team data

CONS:

-We do not provide 100% accuracy

-Technical issues due to lack of experience.

-Dependency on internet connectivity

CHAPTER 3

REQUIREMENT ARTIFACT

REQUIREMENTS ARTIFACTS

A. Functional requirements: - Data from past cricket matches must be able to be collected and stored by the system.

- The algorithm should be able to forecast match results, player stats, and other pertinent information with accuracy.
- To provide personalised forecasts, the system need to enable users to enter certain match parameters or player details.
- A user-friendly interface is essential for seamless data entry and interaction inside the system.
- During live matches, the system need to offer real-time updates and forecasts.

B. Non-Functional Requirements: - The system must be able to make predictions with a minimum degree of accuracy that is acceptable.

- The system must be scalable in order to manage high data volumes and user demands.
- The system needs to react quickly in order to deliver forecasts on time.
- The data's integrity and confidentiality should be guaranteed by the system's security.
- The system need to be usable on a range of platforms and devices.

C. Data Requirements: - An extensive and trustworthy collection of past cricket match data should be available to the system.

- Data on team performances, player statistics, match circumstances, and other pertinent variables should be included in the dataset.

To guarantee the precision and applicability of the forecasts, the data should be updated on a regular basis.

D. Performance Requirements: - The system must have the capacity to effectively handle and analyse massive volumes of data.

- The prediction algorithms have to deliver outcomes in an acceptable amount of time.
- Multiple user requests should be handled by the system at once without noticeably degrading performance.

3.1 INTRODUCTION

. Functional requirements: - Data from past cricket matches must be able to be collected and stored by the system.

- The algorithm should be able to forecast match results, player stats, and other pertinent information with accuracy.
- To provide personalised forecasts, the system need to enable users to enter certain match parameters or player details.
- A user-friendly interface is essential for seamless data entry and interaction inside the system.
- During live matches, the system need to offer real-time updates and forecasts.

B. Non-Functional Requirements: - The system must be able to make predictions with a minimum degree of accuracy that is acceptable.

- The system must be scalable in order to manage high data volumes and user demands.
- The system needs to react quickly in order to deliver forecasts on time.
- The data's integrity and confidentiality should be guaranteed by the system's security.
- The system need to be usable on a range of platforms and devices.

C. Data Requirements: - An extensive and trustworthy collection of past cricket match data should be available to the system.

- Data on team performances, player statistics, match circumstances, and other pertinent variables should be included in the dataset.

To guarantee the precision and applicability of the forecasts, the data should be updated on a regular basis.

D. Performance Requirements: - The system must have the capacity to effectively handle and analyse massive volumes of data.

- The prediction algorithms have to deliver outcomes in an acceptable amount of time.
- Multiple user requests should be handled by the system at once without noticeably degrading performance.

3.2 HARDWARE AND SOFTWARE REQUIREMENTS

1)Hardware: -

- Operating System: Windows 7, macOS 10.13 High Sierra, or any Linuxdistribution released in the last 7 years
- CPU: 1 gigahertz (GHz) or faster 32-bit (x86) or 64-bit (x64) processor
- Memory: 1 gigabyte (GB) RAM (32-bit) or 2 GB RAM (64-bit)
- GPU: DirectX 9 graphics device with WDDM 1.0 or higher driver
- Browser: Chrome 60+/Safari 10.1+ / iOS Safari 10.1+/Edge 12+/Firefox ESR+/Opera.

2) Software: -

- phython
- ML

3.3 PROJECT SPECIFIC REQUIREMENTS

Project specific requiremnents are the requirements that are specific to aparticular project or product. These requirements are derived from the project objectives, goals and scope.

3.3.1 DATA REQUIREMENT

An extensive and trustworthy collection of past cricket match data should be available to the system.

- Data on team performances, player statistics, match circumstances, and other pertinent variables should be included in the dataset.

To guarantee the precision and applicability of the forecasts, the data should be updated on a regular basis

3.3.2 FUNCTIONS

Data from past cricket matches must be able to be collected and stored by the system.

- The algorithm should be able to forecast match results, player stats, and other pertinent information with accuracy.
- To provide personalised forecasts, the system need to enable users to enter certain match parameters or player details.
- A user-friendly interface is essential for seamless data entry and interaction inside the system.
- During live matches, the system need to offer real-time updates and forecasts.

B. Non-Functional Requirements: - The system must be able to make predictions with a minimum degree of

accuracy that is acceptable.

- The system must be scalable in order to manage high data volumes and user demands.
- The system needs to react quickly in order to deliver forecasts on time.
- The data's integrity and confidentiality should be guaranteed by the system's security.
- The system need to be usable on a range of platforms and devices.

3.3.3 PERFORMANCE AND SECURITY REQUIREMENTS

Ensuring optimal performance and robust security is critical for a directly impacts user experience and data protection. Here are key performance - The system must have the capacity to effectively handle and analyse massive volumes of data.

- The prediction algorithms have to deliver outcomes in an acceptable amount of time.
- Multiple user requests should be handled by the system at once without noticeably degrading performance.

CHAPTER 4

DESIGN METHODOLOGY AND ITS NOVELITY

1.1. METHODOLOGY AND GOAL

1. User Research: Start by understanding your target audience,

2. Database Design: Create a database to store player stats and stadium records

3. Security: Implement robust security measures to protect data

4. User Engagement: Incorporate features like comments, likes, andshares to encourage user interaction.

1.2. FUNCTIONAL MODULES DESIGN AND ANALYSIS

low Diagram for Search System

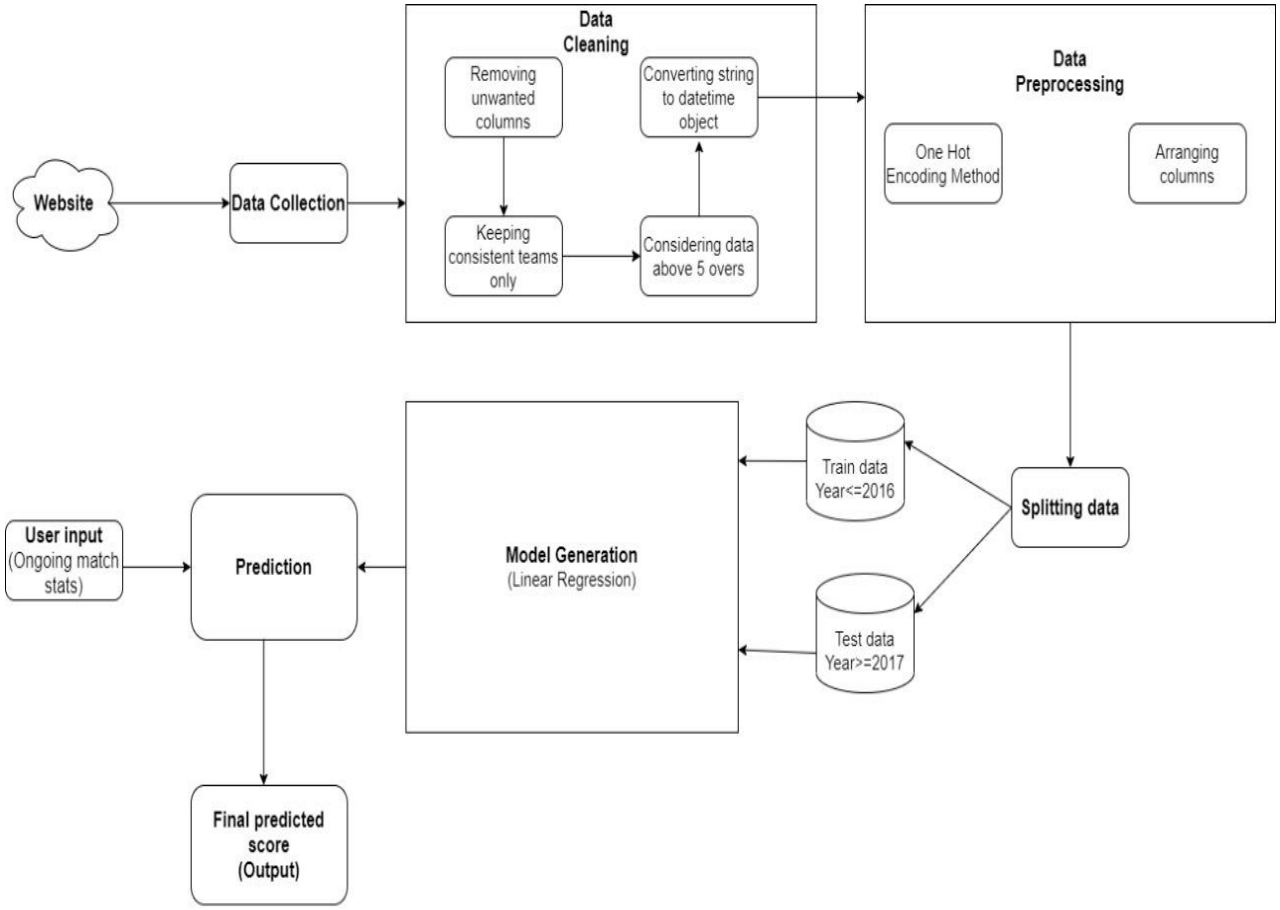
First the system will be fed with the input dataset where it comprises data like player details, player score and place where match is played etc. Then the data will be processed further and split into training and testing datasets. Now the training dataset is further split into supervised and unsupervised learning. Here some suitable algorithms will be applied for the supervised learning datasets and those algorithms are Lasso Regression, Naïve bayes, Logistic Regression, Support vector Machine and

Random Forest Algorithms. The suitable algorithm will be picked to predict the outcome hence it will be matched with testing datasets and score model will be generated.

- Input Pre-processing:

Flow Diagram of Input Pre-processing The steps in input pre-processing include: First the dataset will be loaded and later the analytical rules will be applied. As the data is not pure and cleaned the cleaning process will be performed which clears the outliers. The next step Is to train the individual model which is used to measure the accuracy which in return helps the score prediction. For this to happen we need to select the most appropriate model and select the suitable model based on the dataset. Now after the training and forming the model user will have to give the input. The system will accept the input and will be matched to the model. Later the output will be

4.3 SOFTWARE ARCHITECTURAL DESIGN



Model Architecture

The system typically involves data collection, preprocessing, feature extraction, model training, and prediction. Data is collected from various sources like match records, player statistics, and weather data. After preprocessing the data, relevant features are extracted, such as batting averages, bowling economy rates, and pitch conditions. Machine learning algorithms are then trained on this data to create predictive models. These models are used to make predictions on upcoming matches, considering factors like team performance, player

form, and other relevant variables. The predictions can be used by teams, coaches, and fans to make informed decisions and enhance their understanding of the game. The architecture diagram would illustrate the flow of data and the different components involved in the cricket prediction system.

4.4 SUBSYSTEM SERVICES

- Input Pre-processing:

Flow Diagram of Input Pre-processing The steps in input pre-processing include First the dataset will be loaded and later the analytical rules will be applied. As the data is not pure and cleaned the cleaning process will be performed which clears the outliers. The next step is to train the individual model which is used to measure the accuracy which in return helps the score prediction. For this to happen we need to select the most appropriate model and select the suitable model based on the dataset. Now after the training and forming the model user will have to give the input. The system will accept the input and will be matched to the model. Later the output will be predicted and the desired score is displayed.

TECHNICAL IMPLIEMENTATIONS AND ANALYSIS:-

Algorithm

A) Lasso Regression:

Lasso regression is a regularization technique. it's used over regression strategies for accurate prediction. This model uses shrinkage. Shrinkage is where data values are shrunken to a central point as the mean. The lasso procedure encourages simple, straightforward, thin models (i.e. models with fewer parameters). This specific style of regression is well-suited for models showing high levels of multiple regression or when you want to automatise certain elements of model, like variable selection/parameter elimination.

B) Random Forest Classifier:

Random Forest is a classifier that contains variety of decision trees on varied subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset. Instead of hoping on one decision tree, the random forest takes the prediction from each and every tree and based on the majority votes of predictions, and it predicts the final output. The more the number of trees within the forest results in higher accuracy and prevents the matter of overfitting.

C) Naive Bayes Algorithm:

Naive Bayes is a simple technique for constructing classifiers: models that assign class labels to problem instances, represented as vectors of feature values, where the class labels are drawn from some finite set. There is not a single algorithm for training such classifiers, but a family of algorithms based on a common principle: all naive Bayes classifiers assume that the value of a particular feature is independent of the value of any other feature, given the class variable. For example, a fruit may be considered to be an apple if it is red, round, and about 10 cm in diameter. A naive Bayes classifier considers each of these features to contribute independently to the probability that this fruit is an apple, regardless of any possible correlations between the color, roundness, and diameter features.

D) Logistic Regression Algorithm:

Logistic regression is a classification techniqu borrowed by machine learning from the field of statistics. Logistic Regression is a statistical method

for analyzing a dataset in which there are one or more independent variables that determine an outcome. The intention behind using logistic regression is to find the best fitting model to describe the relationship between the dependent and the independent variable.

PROJECT OUTCOMES AND APPLICABILIITY :-

It is advantageous for many stakeholders to use machine learning to analyse cricket games by taking previous game data, player performance, natural parameters, pre-game conditions, and other features into account. Predicting the result of a game in a dynamic format like T20, when the scenario in a game changes with every ball, is difficult. We have looked into machine learning technologies to see if it can increase the accuracy with which results of matches are predicted for T20 cricket matches. In order to better understand the issue, we have divided it into two scenarios: the Home Team features set and the Toss Winner determination features set. The model constructed on Toss related features yields marginally better outcomes than Home Advantage in terms of the assessment measures utilised, according to an analysis of the results obtained using four different machine learning approaches on 10 years' worth of T20 matches (Accuracy, Precision, Recall,FPs, FNs, etc). When processing the Toss Winner feature set, Lasso Regression, Random Forest Classifier, Naive Bayes Algorithm, and Logistic Regression Algorithm performed better than the other algorithms because they produced more accurate predictive models than Decision Trees, Probabilistic, and Statistical models. Furthermore, the number of occurrences that the aforementioned algorithms, both FPs and FNs, wrongly classify is low, leading to improved Precision and Recall rates. The aforementioned approach successfully identified 134 instances that were incorrectly categorised as belonging to the "Lose" class and 105 instances—or around 35%— were incorrectly labelled as "Wins." The class independence assumption of the procedure, however, makes the outcomes of Nave Bayes on the Toss Decision subset unpromising. However, the Home Team subset yielded higher results using Nave Bayes. Team management and academics interested in cricket data analytics will help people to analyse and bet well.

5.1 TEST AND VALIDATION: -

```
#Gaussian NAive bayes algorithm
from sklearn.naive_bayes import GaussianNB
outcome_var=['winner']
predictor_var = ['team1', 'team2', 'venue', 'toss_winner','city','toss_decision']
model = GaussianNB()
classification_model(model, df,predictor_var,outcome_var)
```

✓ 0.0s Python

```
C:\Users\ADMIN\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\sklearn\utils\validation.p
  y = column_or_1d(y, warn=True)
[ 3  5  5  1  2  6  5 12 12 12  3  5 12  5  5  2 12 12  5  5  5  6 12  6
 12 12 12 12  2 12  3 12  1 12  5  2 12 12 12 12  5 12 12  2  3 12 12  5
 12 12  5  3  2 12  5 12  6 12  3  5 12  5  1 12  5  5  3  5  5  1  6 12
  1  6  2  2  6  6  2  5  1 12  1  1  5 12  5  1  6  6  2 12  5  1 12  5
  1  6 12  5  5 12  1 12 12  6  5  5 12  5 12  5  6 12 12 12  2 12 12  5
  6  3  2 12  3  5 12  5  2  5 12  2  3 12 12  5  5  3  6 12  5 12  5  5
  2  6  5  2  2  5  6  5  6  2  3  3  1  3  5 12 12  2  5  6  5 12  3  2
 12  5  2  2  2  2  2  5 12 13  5 13  2  6  3 12 13 12 12 12  5 12 12 12
 12  5 13 12 12 12 12 12 12 12 12 12 13 12 12 12 12 10 12 12  5 12 12 12
  5 12 12 12 12  2 12  9 10 12  4 12 12  5 12  3  1 12 12 12  5  5 12  1
 12 10  3 12 12 12 12  5 12 12 12 12  3 12 12 13  2  3 12 12  5 12  2  5
 13 12  5 12  5 10  6  5 12 12  5 12  1 12 12 13  3 13 12  6  2  5 12  5
  4  6 12  1  5  2  5 12  3  5 13 12 12 12 12  2 12 10  5  3 12  7 12 12
 12  5 12  1 12  5  1  3 12  5 12  3 12  5  3 13  5 12  1 12  3 13  5  1
 12  2 12 10 12  3 12  5  5 10  2  3  2 12  5 12 12  2  9  2  6  1 12 12
 12  1 13  1  1  5  1  2  5 12  1 13  9  6  1 12 12  1 12 13 12 12 12 12
 13 12  5 12 12  6 12 13  3  5  5  7  2  2  1 12  5  6 12  5 12  6 12 12
 12 10  2  6  3  1 13 12  5 12  2  1 12  3  3 12 12  5 12  9  3 12 12 12
  5  9  5  3 12 12  6  2  6  3 12  5 12 12 12 12 12 12  3 12 12 12 12 12
 12  3 12 12 12  5 12 12 12 12  3 12 10 12  1 10 12  5  3 12  9 12  3 12
  3  1  5 12 12  2  3 12 12 12  3 12  3 12  5 12  1 12 12  5  2 13  1  5
 10  6 12 12  6  6  2  3  9  1  5 12 12 11 12 12 12 12 13 12  6 12 10  3
 12 12 12 12 11 12 12  1 12 12 11 12 12 12 12 12 12 12  3 12 12 11 12  9
 12 12 12 12 11  3 12  1  3 13  9 12 12 11 12 12 12 11 12 12 12  3 12 12
  9]
Accuracy : 20.624%
```

```
    outcome_var=['winner']
    predictor_var = ['team1', 'team2', 'venue', 'toss_winner','city','toss_decision']
    model =LogisticRegression()
    classification_model(model, df,predictor_var,outcome_var)
```

✓ 0.1s

```
[ 3  5  5  1  2  6  5  3  3  9  3  5  1  9  5  1  3  3  1  5  1  6  2  5
  3  3  2  9  2  3  3  9  1  6  5  5  1  9  3  1 10  1  1  2  3  2  1  5
  3  3  5  3  2  1  5  6  6  2  3  5  9  5  1  2  5  5  3  5  1  1  5  3
  1  6  5  5  6  6  2  5  1  3  5  1  5  9  5  1  3  6  5  7  5  1  3  5
  3  6  3  5  5  9  1  1  9  5  6  5  2  5  3  6  6  1  1  2  2  9  2  5
  9  5  2  3  3  5  3  6  2  5  9  2  3  9  2  5  5  3  6  1  5  2  5  5
  2  6  5  2  5  5  1  5  6  5  3  5  1  3 10  1  1  2  5  1  5  9  3  2
  9  5  1  2  2  2  2  5  3  5  5 10  2  6  3  9 10  1  1  1  5  3 10  2
  7  5  6  2  9  1  1  9  1  9  9  3  5  7  3  1  9 10  2  9  5  1  9  1
  5  8  9  3  3  2  1  5 10  3 10  9  1  5  9  3  1  9  9 10  5 10  2  1
  3 10  3  1  1  1  3  5  3  2  6  6  3  2  1  6  2  3  3  1  5  9  2  5
 10  2  5  1  5  5  6  5  9  9  5  9  6  3  1  6  6 10  2  5  2  5  1  5
 14  6  3  7  5 10  9  1  3  5  6  9  1  6  6  1  9 10  5  3  1  5  1  3
  3  5  9  7  1  1  1  3  3  5  2  3  9  5  5  6  5  1  1  9  3 10 10  6
  3  5  9  5  9  3  9  1  5 10  2  3  5  9  5  9  9  1  5  2  6  1  3  9
  1  6 10  5  1  5  7  2 10  3  1  6  5  6  1  9  9  7  3  6  9  1  3  9
  6  2  5  1  3  6  2  6  3  5  5  5  2  2  1  1  6  9  3  5  6  6  9  3
  1  5  5  3  9  1  6  3  6  2  1  1  9  9  3  9  1  5  9  9  9  9  1  2
  5  6  2  3  9  3  9  2  6  5  9  5  2  9  1  2  1  9  3  2  9  1  3  2
  6  9  1  3  7  5  1  9  1  3  5  2  5  2  1 10  2  6  3  3  9  2  3  2
  3  6  5  3  9  5  3  3  9  1  3  9  5  2  5  3  1  3  7  5  2  6  1  5
 10  5  9  1  6  6  2  3  6  1  2  1  1  6  2  9  3  1  6  9  6  1  9  3
  3  9  1  9  8  2  9  5  2  3  9  9  1  9  3  9  9  2  3  6  3  9  9  9
  9  2  2  3 10  3  9  2  3  5  9  1  1  7  3  3  9 10  3  3  1  9  9  9
  5]
Accuracy : 30.676%
```

```
#applying knn algorithm
from sklearn.neighbors import KNeighborsClassifier
model = KNeighborsClassifier(n_neighbors=3)
classification_model(model, df,predictor_var,outcome_var)
```

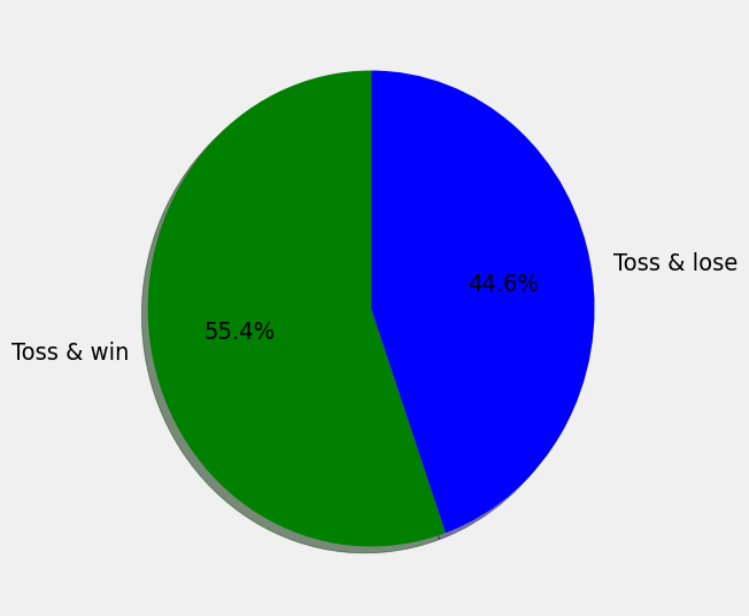
✓ 0.0s

```
[ 3  5  5  1  2  6  7  5  6  9  6  5  1  9  3  1  3  9  1  7  4  2  1  6
  3  5  1  5  2  3  1  5  2  6  9  2  1  9  3  1  9  2  1  5  3  5  9  5
  9  1  5  3  2  6  3  6  1  6  1  1  7  4  4  2  4  7  2  3  3  2  7  3
  1  5  4  9  5  5  1  3  5  3  2  4  5  4  3  2  5  5  7  2  5  1  4  4
  3  1  3  4  3  9  1  4  3  7  4  3  2  3  7  3  2  4  4  1  1  9  1  3
  6  3  2  1  3  7  4  6  1  4  5  1  3  9  1  6  1  3  6  1  2  1  5  7
  2  2  5  1  2  7  1  5  6  2  3  1  4  3  9  6  4  1  5  1  7  4  1  2
  4  4  2  1  3  2  5  2  6  3  1 13  2  6  1  9 13  4  6  1  5  9  4  2
  4  4  1  2  9  1  3  9  1  6  5  3  2  3  1  6  3  4  2  6  5  1 10  1
  5  2  2  6  3  2  1  3 10  6 10  9  2  7  9  3  4  9  6  2  2  5  2  1
  3  9  3  1  5  1  1  5  5  6  9  6  3  1  6 13  1  3  7  1  5  9  2  4
 13  2  3  1  6  3  2  7  5  9  5  7  1  2  6  6  1  3  7  9  2  7  1  2
  4  6  3  2  5  2  5  1  3  2  6 10  1  6  3  2  5  6  9  1  1  7  1  9
  3  3  9  2  4  6  2  1  5  2  2  1 10  5  5 13 10  6  1  9  3 13  9  1
  5  2  6  3  9  3  9  6  4 10  5  3  7  9  5  3  9  1  5  2  6  1  5 10
  6  1 13 10  7  5  2  2 10  3  1  6  3  6  1  9  9  2  3  1 10  2  6  9
  1  1  7  1  9 10  7  6  3 10  1  6  2  5  2  3  5  5  3  7  9  5  9  7
  2  6  3  5  6  7  5  3  6 10  5  1  6  3  6  5  1  2  2  6  3 10  1  2
  3  1  5  3 10  2  6  5  2  6  9 10  2  9  3  3  1  9  3  2  9  1  2  1
  5  2  2  5  3  5  3  6  9  3  6  2  6  6  1 11  2  6  1  2  6 10  3  7
  6  1  5  3 10  2  6  5  7  1  3 10  1  6  5  2  1  3  6  5  2  7  1  5
 10  6  9  1  2  3  5  3  1  1  3  5  1  1  2  7  3  1  8  7  2  1 10  3
  9  9  1  9  3  7 10  7  2  1 10  7  1  6  2 10  7  2  1  7  2 10 10  3
  7 10  2  9 11  1  7  1  3  2 10  1  3 11  3  8  7  7  8  2  3  3  2 10
  3]
Accuracy : 62.218%
```

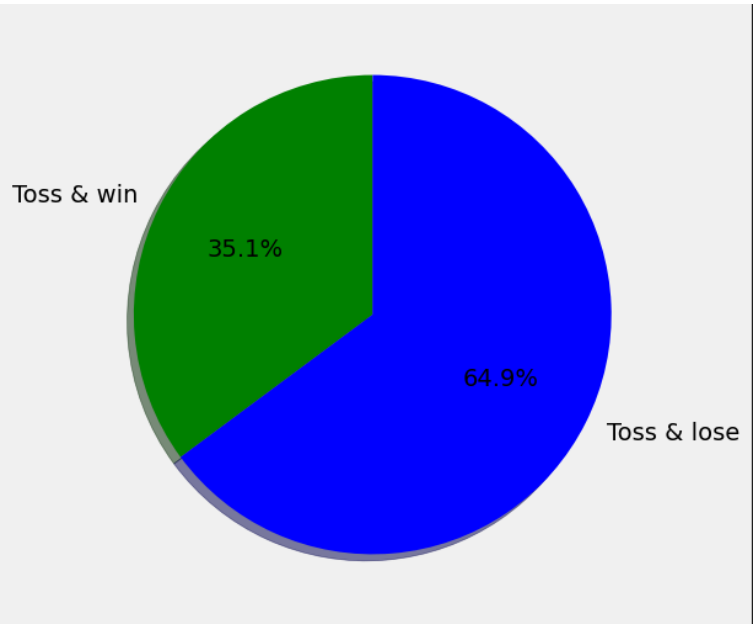
```
#Gradient boost algorithm
from sklearn.ensemble import GradientBoostingClassifier
model= GradientBoostingClassifier(n_estimators=1000, learning_rate=0.1, max_depth=3, random_state=0)
classification_model(model, df,predictor_var,outcome_var)
```

✓ 19.5s

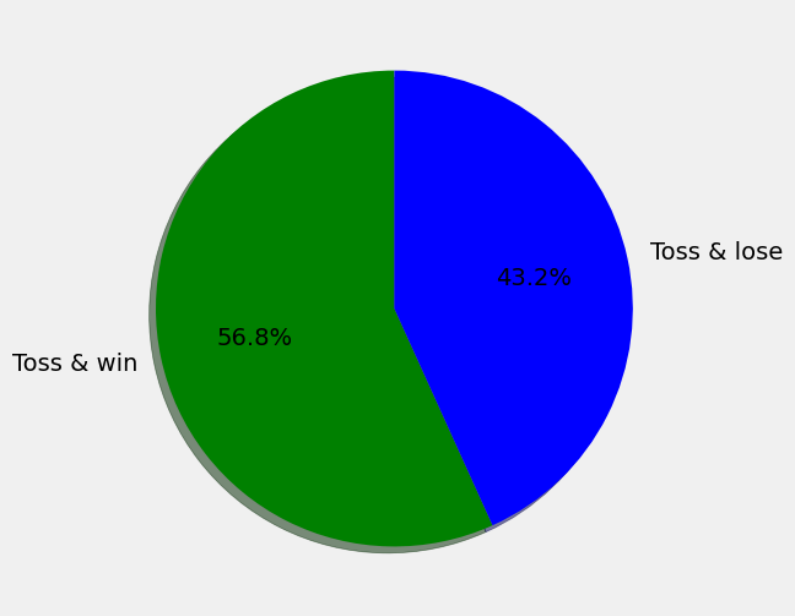
```
C:\Users\ADMIN\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.10_qbz5n2kfra8p0\LocalCache\local
  y = column_or_1d(y, warn=True)
[ 3  5  6  1  2  6  7  5  6  9 14  5  1  9  5  1  3  9  6  5  3  2  1  6
  3  4  1  5  2  6  1  5  2  6  9  2  1  9  7  1  9  3  1  5  3  6  9  5
  9  7  5  4  2  6  5  6  5  6  1  3  7  4  5  2  4  7  6  9  4  7  9  4
  1  6  3  9  7  5  1  9  6  5  9  3  5  6  7  4  6  5  7  9  5  1  7  4
  3  1  7  3  6  9  5  4  9  7  2  3  2  5  7  3  4  3  4  2  1  7  2  4
  6  3  2  7  3  5  4  6  3  4  9  1  3  6  1  6  2  3  6  1  2  1  5  7
  2  3  5  1  9  7  6  1  6  2  4  1  4  3  9  6  4  1  5  3  5  4  1  2
  5  4  2  1  5  3  5  2  6  3  1 13  2  6  1  9 13  4  6 12  5  9  7  2
 12  4  1 12  9  1  3  9  1  6  5  3  5  4  2  6  3  7  2  6  5  1 12  2
  5  1 12  7  3  2  1  3 13  6 13  9  3  5  9  3  4  9 12  4  9  5  2  6
  4 14  5  1  5  1  3  5  5  7 13  6  3  5  6  9  1  3  7  1  5  9  2  4
 13  2  6  1  6  3  2  7  5  9  5 13  9  2  6  7  1  4  7  9  2  7  1  2
  4  7  3  1  5  2  6  1  3  2  6  9  3  5  3  2  5  6  9  1  5  7  1  9
  3  4  9  2  4  6  2  5  5  2  2  1 10  6  1  9 10  6  1  5  3 13 10  1
  5  2  6 13  9  3 10  6  5 10  5  3  7  9  5  3  9  1  5  2  6  1  5  7
  6  9 13 10  7  5  3  2 10  3  1  6  3  6  1  5  9  2  3  1 10  2  6  9
  1 13  5  1  9 10  9  7  3 10  5  6  1  1  2  3  9  6  3  7  9  5  9  5
  2 10  5  6  9  7  5  9  6 10  5  1  6  3  6  5  3  2  9 10  9 10  1  2
  6  1  5  3  9  2  6  3  2  6  9 10  2  9  3 10  1  9  3  2  9  1  2  5
  9  2  2  5  6  5  3  6  9 10  6  2  7  6  1  7  2  6  1  7  9 10  5  7
  3  1  5  3 10  2 14  5  7  1  3 10  1  6  5  2  1  3 10  1  2 10  3  5
 10  7  9  1  3  5  6  3  1  1  3  5  1 11  2  8 10  1  8  7  2  8  9  3
 10  2  1 10  3  7 10  8  2  1 11  8  1  8  7 10  9  1  2  7  2 11 10  3
  9 10  8  3 10  1  7  9  3  2 10  1  3 11  3  8  7 11  8  2  3  3 10 10
  3]
Accuracy : 89.601%
```



MI



KXIP



CSK

5.2 ACCURACY

S. No.	Model	Accuracy
1	Regression	30.676%
2	Gaussion nb	20.642%
3	Kneighbors classisfier	62.218%
4	SVM	86.081%
5	Gradient booster classifier	89.601%
6	Random forest	89.601%

▶ Unit Testing:

- Unit testing focuses on testing individual components or functions inisolation.

- Unit tests for functions that perform specific tasks,

▶ Integration Testing:

- Integration testing verifies that different parts of our system worktogether correctly.
- Test interactions between the backend, ensuring that data issent and received correctly.

▶ End-to-End (E2E) Testing:

- E2E testing checks the functionality of our entire application, simulatingreal user interactions.
- Tools like Selenium, Puppeteer, or Cypress Was used for E2E testing.

▶ User Acceptance Testing (UAT):

- UAT involves real users testing the application to ensure it meets theirneeds and expectations.

- Collected feedback from actual users and made necessary improvementsbased on their input.

▶ Performance Testing:

- Performance testing assesses the responsiveness and scalability of yourapplication.
- Tools like sklearn algorithm, random forest classified, svm, regression, gaussian nm , gradient booster classifier

▶ Cross-Browser Testing:

- Test our application on different web browsers (e.g., Chrome, Firefox,Edge, Safari) to ensure it works consistently across all of them.

▶ Regression Testing:

- Conduct regression testing whenever code changes are made to ensurethat existing features still work as expected.

▶ Key Features:

1. **Theme Switching:** Offers light and dark themes for user preference.

.

▶ User Experience and Testing:

- Well-designed, user-friendly interface.
- Extensive testing ensures reliability.
- Positive user feedback from user acceptance testing.

▶ Performance:

- Scalable and responsive to user demands.

▶ **Future Enhancements:** Potential improvements based on user feedback include additional filters

6.3. PROJECT APPLICABILITY ON REAL-WORLD APPLICATIONS

It is advantageous for many stakeholders to use machine learning to analyse cricket games by taking previous game data, player performance, natural parameters, pre-game conditions, and other featuresinto account. Predicting the result of a game in a dynamic format like T20, when the scenario in a game changes with every ball, is difficult. We have looked into machine learning technologies to see I fit can increase the accuracy with which results of matches are predicted for T20 cricket matches. In order to better understand the issue, we have divided it into two scenarios: the Home Team features set and the Toss Winner determination features set. The model constructed on Toss related features yields marginally better outcomes than Home Advantage in terms of the assessment measures utilised, according to an analysis of the results obtained using four different machine learning approaches on 10 years' worth of T20 matches (Accuracy, Precision, Recall,FPs, FNs, etc). When processing the Toss Winner feature set, Lasso Regression, Random Forest Classifier, Naive Bayes Algorithm, and Logistic Regression Algorithm performed better than the other algorithms because they produced more accurate predictive models than Decision Trees, Probabilistic, and Statistical models. Furthermore, the number of occurrences that the aforementioned algorithms, both FPs and FNs, wrongly classify is low, leading to improved Precision and Recall rates. The aforementioned approach successfully identified 134 instances that were incorrectly categorised as belonging to the "Lose" class and 105 instances—or around 35%— were incorrectly labelled as "Wins." The class independence assumption of the procedure, however, makes the outcomes of Nave Bayes on the Toss Decision subset unpromising. However, the Home Team subset yielded higher results using Nave Bayes. Team management and academics interested in cricket data analytics will help people

to analyse and bet well.

CHAPTER 7

CONCLUSION AND RECOMMENDATION

The goal of this research is to use past data to forecast the final score and match winner. Data Pre- processing, Data Visualizations, Data Preparation, Data Selection, and Machine Learning Model Implementation are some of the fields of Data Science that will come together to conduct the study and forecast the match's score. To accurately forecast the score of innings and obtain the desired outcome, a number of machine learning models will be applied to specified date.

APPENDIX-A

Programming Languages : phython, ml

Software tool -visualstudio, Jupiter notebook,

Version Control: Git: Used for managing the app's source code and versioncontrol.

Team guidelines- proper player stats and team winner prediction of each match will be given.

APPENDIX-B

1. **Machine Learning:** A subset of artificial intelligence that allows asystem to automatically improve performance by learning from experience.
2. **Android:** A mobile operating system developed by Google, designedprimarily for touchscreen mobile devices such as smartphones and tablets.
3. **Input Fields:** A feature that allows users to input data or information intoa system.
4. **Prototype:** A preliminary model of a product used to test and evaluate thedesign before the final product is produced.
5. **Test and Validation:** A process of evaluating a product to ensure that itmeets the desired quality standards and functional requirements.
6. **Performance Analysis:** A process of measuring and evaluating the effectiveness and efficiency of a product.
7. **Back-end:** The part of a system that is not directly accessible by usersand is responsible for managing data and processing requests.

8. **Database:** A structured collection of data that is stored and organized in aspecific way to enable efficient retrieval and manipulation.
9. **Model:** A representation of a system that is used to make predictions or togain insight into the behavior of the system.
10. **Algorithm:** A set of instructions or rules that a computer follows toperform a specific task.
11. **UI/UX Design:** User Interface/User Experience Design, a process ofdesigning the visual and interactive elements of a product to ensure a seamless user experience.
12. **Regression Analysis:** A statistical method used to model the relationshipbetween a dependent variable and one or more independent variables.
13. **Artificial Intelligence:** The simulation of human intelligence processes by computer systems, including learning, reasoning, and self-correction.
14. **Data Visualization:** A graphical representation of data and information.
15. **Supervised Learning:** A type of machine learning where a system is trained using labeled data to make predictions.

REFERENCES

1) **R. B. (2014). Learning to detect phishing URLS. International Journal of Research in Engineering and Technology, 03(06), 11–24.**
2) **https://doi.org/10.15623/ijret.2014.0306003**
3) **Balogun, A. O., Adewole, K. S., Raheem, M. O., Akande, O. N., Usman-Hamza, F. E., Mabayoje, M. A., Akintola, A. G., Asaju-Gbolagade, A. W., Jimoh, M. K., Jimoh, R. G., & Adeyemo, V. E. (2021). Improving the phishing website detection using empirical analysis of function tree and its variants. Heliyon,**
4) **https://doi.org/10.1016/j.heliyon.2021.e07437**
5) **Dutta, A. K. (2021). Detecting phishing websites using Machine Learning Technique. PLOS ONE, 16(10). https://doi.org/10.1371/journal.pone.0258361**
6) **Mao, J., Bian, J., Tian, W., Zhu, S., Wei, T., Li, A., & Liang, Z. (2018). Detecting phishing websites via aggregation analysis of page layouts. Procedia Computer Science, 129, 224–230.**
7) **https://doi.org/10.1016/j.procs.2018.03.053**
8) **Hutchinson, S., Zhang, Z., & Liu, Q. (2018). Detecting phishing websites with Random Forest. Machine Learning and Intelligent Communications, 470–479. https://doi.org/10.1007/978-3-030-00557-3_4**
9) **Aung, E.S., Zan, C.T., & Yamana, H. (2019). A Survey of URL-based Phishing Detection.**
10) **Jeeva, S.C., Rajsingh, E.B. Intelligent phishing url detection using association rule mining.**
11) **Hum. Cent. Comput. Inf. Sci. 6, 10 (2016). https://doi.org/10.1186/s13673-016-0064-3**
12) **Huang, Y., Yang, Q., Qin, J., & Wen, W. (2019). Phishing URL Detection via CNN and**
13) **Attention-Based Hierarchical RNN. 2019 18th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/13th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE), 112-119.**
14) **Tang, L.; Mahmoud, Q.H. A Survey of Machine Learning-Based Solutions for Phishing Website Detection. Mach. Learn. Knowl. Extr. 2021, 3,**

672–694. https://doi.org/10.3390/make3030034

15) Natadimadja, Muhammad Rayhan, Maman Abdurohman, and Hilal Hudan Nuha. "A survey on phishing website detection using hadoop." Jurnal Informatika Universitas Pamulang 5.3 (2020): 237-246.

16) Jain, A.K., Gupta, B.B. A novel approach to protect against phishing attacks at client side using auto-updated white-list. EURASIP J. on Info. Security 2016, 9 (2016).

17) https://doi.org/10.1186/s13635-016-0034-3

18) Kalaharsha, P.; Mehtre, B.M. Detecting Phishing Sites—An Overview. arXiv 2021,arXiv:2103.12739