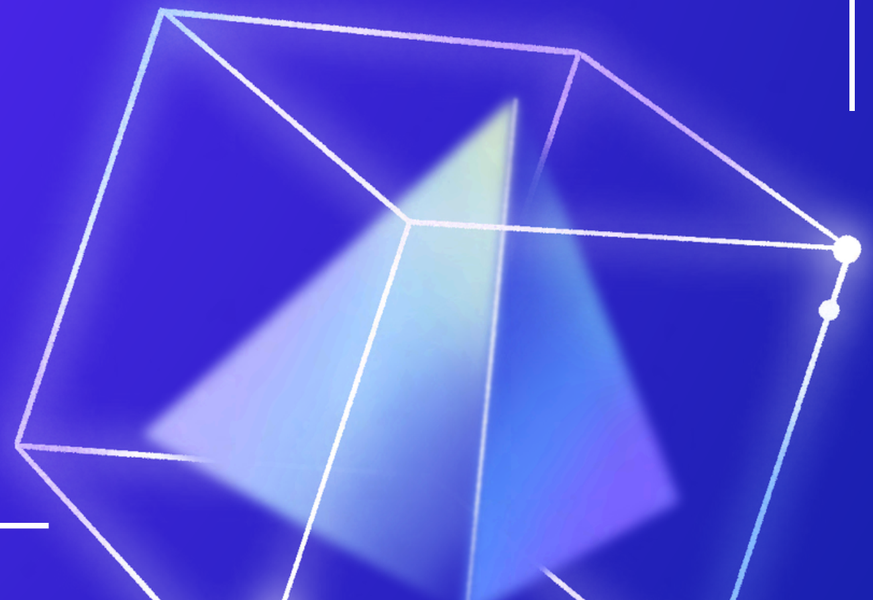
TABLE OF CONTENTS



INTRODUCTION

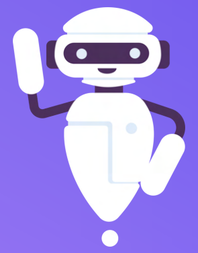
Welcome, cricket enthusiasts, to a fascinating journey into the world of data-driven predictions and cricket excitement!

In a cricket-crazy nation like ours, the Indian Premier League (IPL) holds a special place. It's not just a tournament; it's a celebration of cricketing prowess, strategies, and unpredictable outcomes.

PROBLEM STATEMENT

Predicting IPL match outcomes accurately has always been a challenge due to various dynamic factors like player form, team performance , and match conditions.

PROJECT OBJECTIVES



OBJECTIVE 01

The primary goal is to leverage historical match data, player statistics, team performance metrics, and other relevant features to predict which team is likely to win a particular match.

OBJECTIVE 02

Analyzing the strengths and weaknesses of the opposing team based on historical data can help teams plan their strategies, including team composition, batting order, and bowling rotations.

OBJECTIVE 03

To assist fantasy cricket players in selecting the most effective team for IPL fantasy leagues. An IPL Winning Predictor can contribute to more informed and balanced betting markets.

ADVANTAGES

STATISTICAL INSIGHTS

The predictor provides statistical insights and trends that contribute to a deeper understanding of player and team performances.

CRICKET ANALYTICS

The IPL Winning Predictor contributes to the broader field of cricket analytics and sports science research.

ENHANCED TEAM STRATEGIES

IPL teams can gain strategic insights into their opponents, leading to better team strategies.

ENTERTAINMENT

Cricket fans experience enhanced engagement and excitement as they anticipate match outcomes.

DATASETS

Delivery.csv(149578, 24)

	id	Season	city	date	team1	team2	toss_winner	toss_decision	result	dl_applied	winner	win_by_runs	win_by_wickets
0	1	IPL-2017	Hyderabad	05-04-2017	Sunrisers Hyderabad	Royal Challengers Bangalore	Royal Challengers Bangalore	field	normal	0	Sunrisers Hyderabad	35	0
1	2	IPL-2017	Pune	06-04-2017	Mumbai Indians	Rising Pune Supergiant	Rising Pune Supergiant	field	normal	0	Rising Pune Supergiant	0	7
2	3	IPL-2017	Rajkot	07-04-2017	Gujarat Lions	Kolkata Knight Riders	Kolkata Knight Riders	field	normal	0	Kolkata Knight Riders	0	10

Match.csv(756,18)

	match_id	inning	batting_team	bowling_team	over	ball	batsman	non_striker	bowler	is_super_over	...	bye_runs	legbye_runs	noball_runs	penalt
0	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	1	DA Warner	S Dhawan	TS Mills	0	...	0	0	0	
1	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	2	DA Warner	S Dhawan	TS Mills	0	...	0	0	0	
2	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	3	DA Warner	S Dhawan	TS Mills	0	...	0	0	0	

Final_df(71342, 10)

	batting_team	bowling_team	city	runs_left	balls_left	wickets	total_runs_x	crr	rrr	result
147177	Delhi Capitals	Chennai Super Kings	Chennai	95	30	2	187	6.133333	19.000000	0
127534	Mumbai Indians	Chennai Super Kings	Pune	33	16	8	177	8.307692	12.375000	1
87208	Punjab Kings	Chennai Super Kings	Abu Dhabi	168	98	9	206	10.363636	10.285714	1
144693	Chennai Super Kings	Royal Challengers Bangalore	Bengaluru	32	14	4	166	7.584906	13.714286	0

WORKFLOW

01

DATA EXTRACTION

Data extraction involves retrieving specific information or patterns from a dataset or a source.

02

EDA

EDA is the process of visually and statistically summarizing, interpreting, and understanding the main characteristics of a dataset.

03

DATA CLEANING

Data cleaning is the crucial process of enhancing the quality and reliability of our dataset.

04

DATA PREPROCESSING

It involves cleaning and transforming raw data into a format that is suitable for analysis or model training.

05

VISUALIZATION

Visualization is the representation of data through visual elements like charts, graphs, and maps to facilitate better understanding, analysis, and communication of patterns and insights.

MACHINE LEARNING MODELS

MODELS

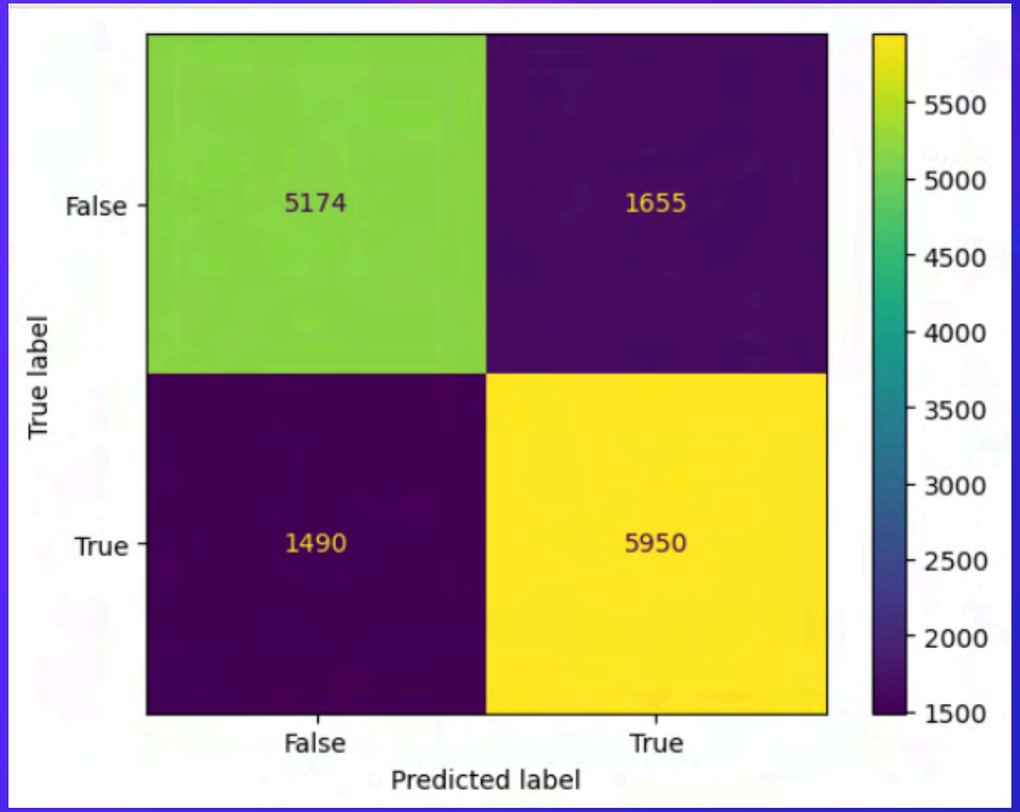
A machine learning model is a computational algorithm or system that is trained on data to make predictions or decisions without being explicitly programmed for the task. In other words, a machine learning model learns patterns and relationships within data and uses that knowledge to make predictions or decisions when presented with new, unseen data

There are seven machine learning classification algorithms that we have used for sentiment analysis:

- Linear Regression
- Logistic Regression
- KNN
- Gaussian Naïve Bayes
- Decision Tree
- Random Forest Classifier
- XGBoost Classifier

LINEAR REGRESSION

Linear regression is a statistical method used for modeling the relationship between a dependent variable and one or more independent variables. It assumes a linear relationship, aiming to find the best-fit line that minimizes the difference between predicted and actual values. The model predicts numerical outcomes based on input features.




```
Training Accuracy: 80.20%
Test Accuracy: 79.78%
Mean Squared Error: 0.20218655827317963
```

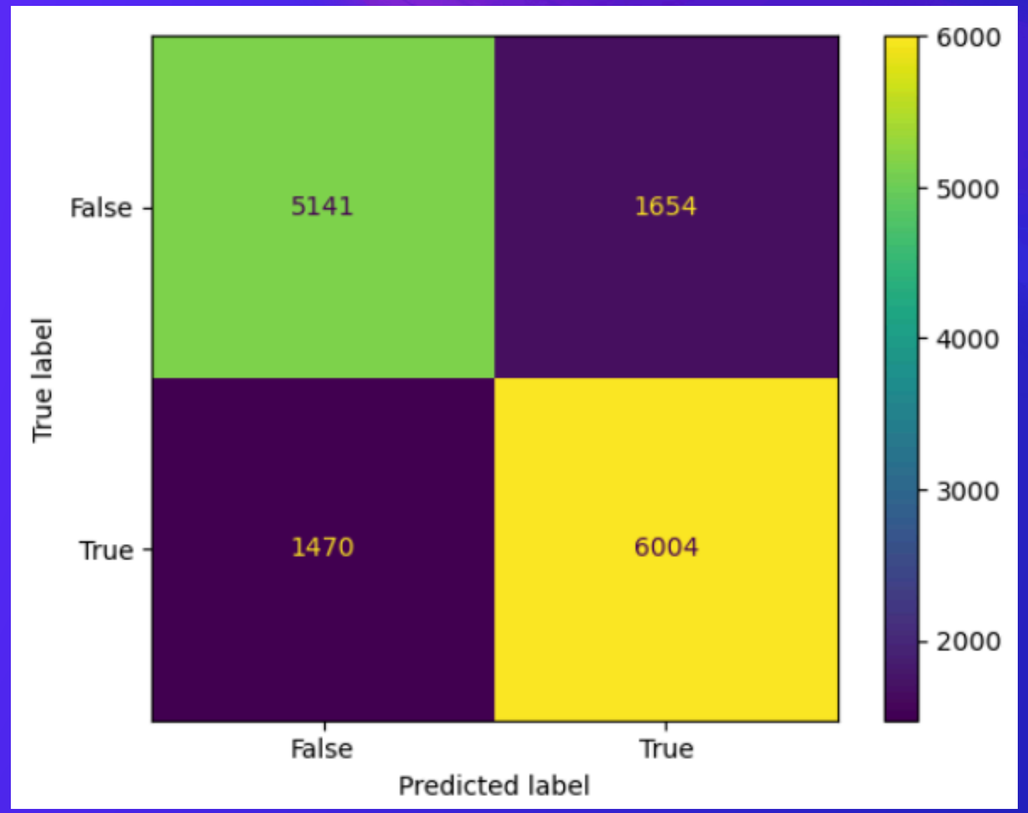
	precision	recall	f1-score
0	0.78	0.76	0.77
1	0.78	0.80	0.79

'0' (referred to as the negative class)

'1' (referred to as the positive class)

LOGISTIC REGRESSION

Logistic regression is a binary classification algorithm used to predict the probability of an instance belonging to a particular class. It models the relationship between independent variables and the log-odds of the dependent variable using a logistic function. The output is transformed to a probability between 0 and 1, making it suitable for classification tasks.




```
Training Accuracy: 78.78%
Test Accuracy: 78.11%
Mean Squared Error: 0.218936155301703
```

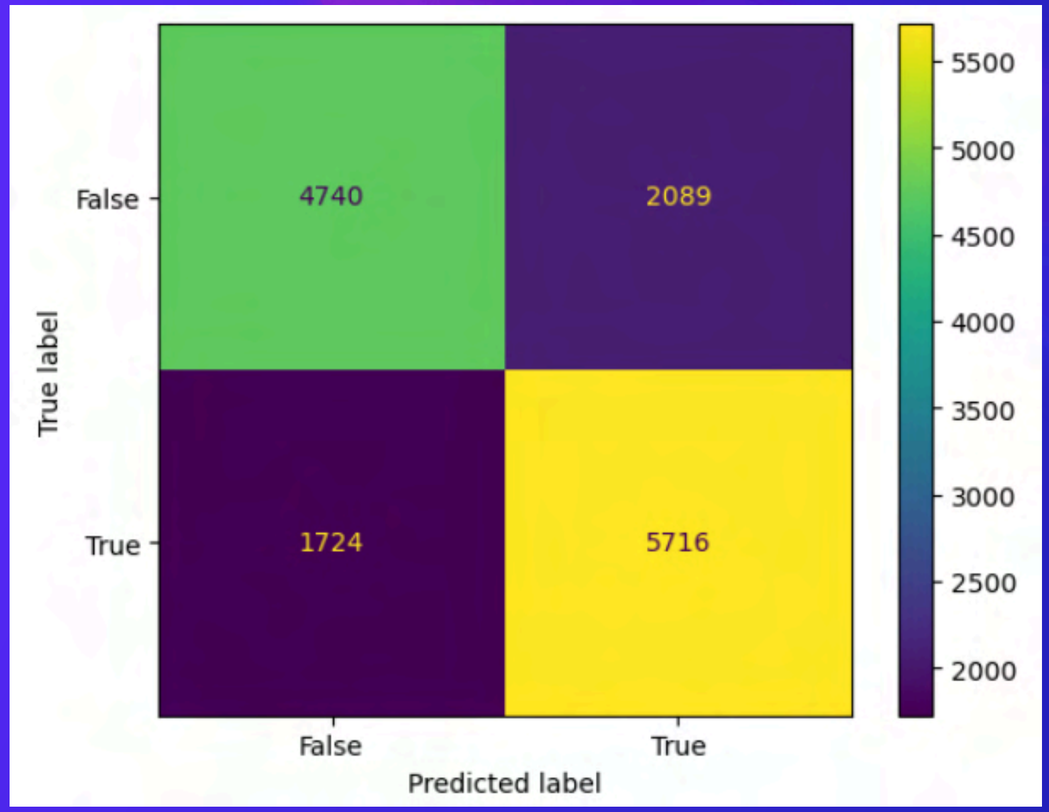
	precision	recall	f1-score
0	0.78	0.76	0.77
1	0.78	0.80	0.79

'0' (referred to as the negative class)

'1' (referred to as the positive class)

KNN CLASSIFIER

K-Nearest Neighbors (KNN) is a classification algorithm. In KNN, an object is classified by the majority class of its k nearest neighbors in the feature space. It works by calculating distances between data points and assigning a class label based on the most common class among its neighbors.




```
Training Accuracy: 73.84%
Test Accuracy: 73.28%
Mean Squared Error: 0.2672226505010863
```

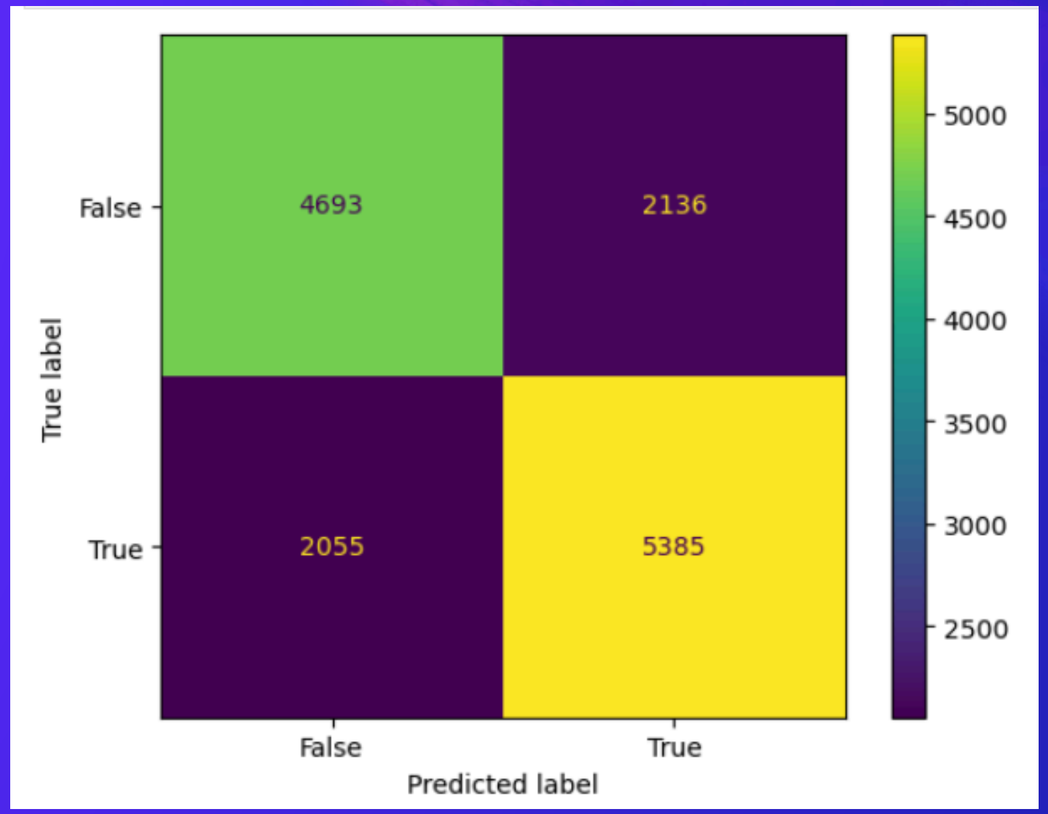
	precision	recall	f1-score
0	0.73	0.69	0.71
1	0.73	0.77	0.75

'0' (referred to as the negative class)

'1' (referred to as the positive class)

GAUSSIAN NAIVE BAYES

Gaussian Naive Bayes is a probabilistic classification algorithm based on Bayes' theorem. It assumes that features are normally distributed within each class. The model calculates the probability of a given instance belonging to each class and assigns it to the class with the highest probability. It's particularly effective for text and real-valued data.



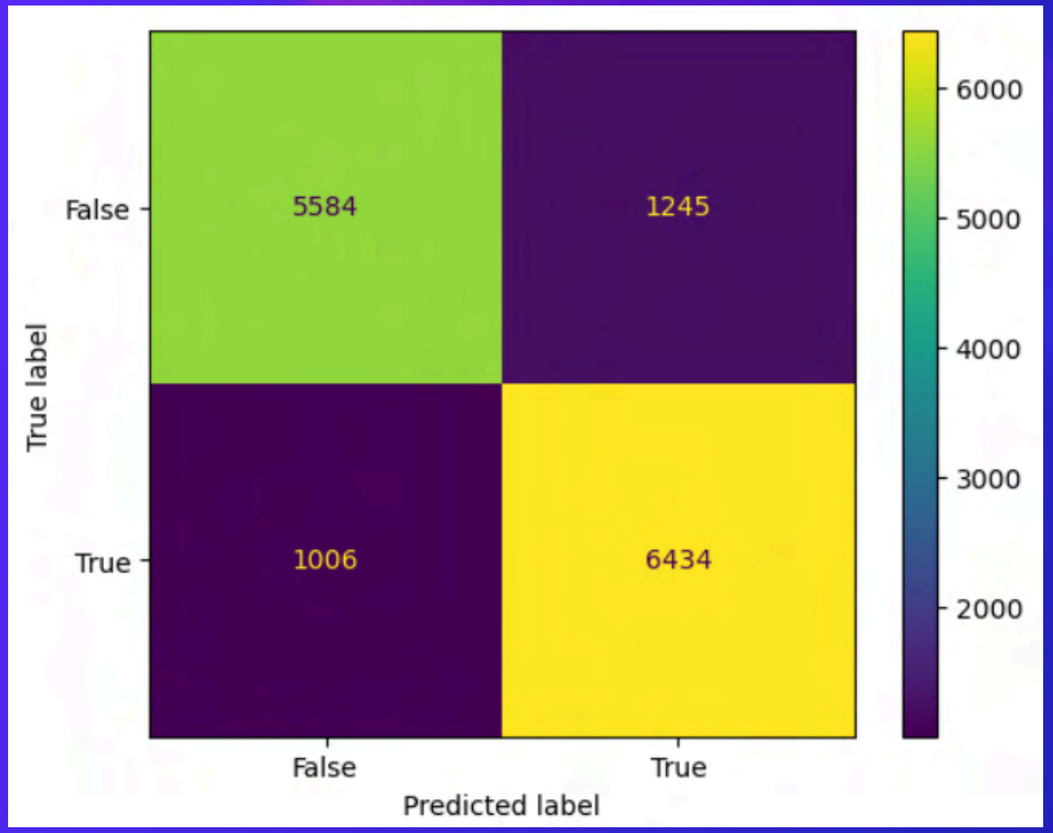

```
Training Accuracy: 70.67%
Test Accuracy: 70.63%
Mean Squared Error: 0.2937136449646086
```

	precision	recall	f1-score
0	0.70	0.69	0.69
1	0.72	0.72	0.72

'0' (referred to as the negative class)
'1' (referred to as the positive class

DECISION TREE

A Decision Tree is a machine learning algorithm that makes decisions by recursively partitioning the data into subsets based on feature values. It selects the best features at each node to maximize information gain or minimize impurity. The process continues until a stopping criterion is met, producing a tree-like structure. During prediction, new data traverses the tree, following decision paths to reach a leaf node, which provides the final outcome or prediction.



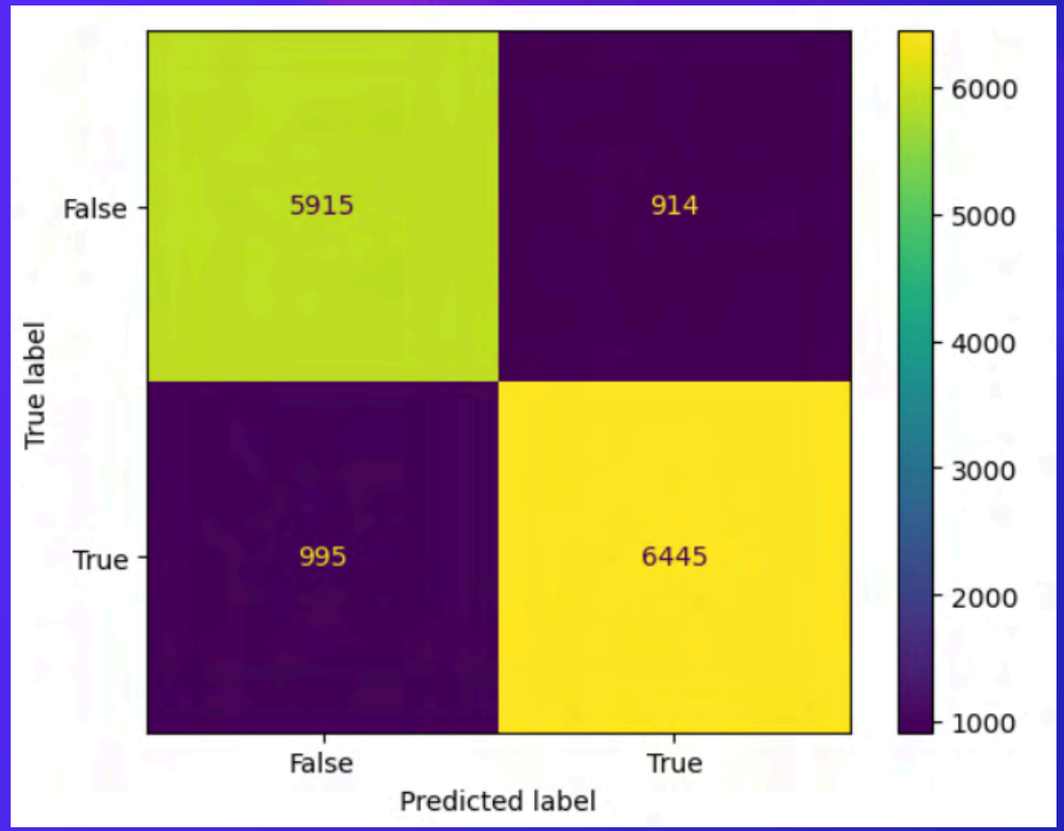

```
Training Accuracy: 85.47%
Test Accuracy: 84.22%
Mean Squared Error: 0.15775457285023478
```

	precision	recall	f1-score
0	0.85	0.82	0.83
1	0.84	0.86	0.85

'0' (referred to as the negative class) '
1' (referred to as the positive class

RANDOM FOREST

Random Forest is an ensemble machine learning algorithm that constructs multiple decision trees during training and outputs the mode (classification) or mean prediction (regression) of the individual trees. It works by randomly selecting subsets of features for each tree and combining their predictions. This randomness enhances robustness and reduces overfitting, resulting in a more accurate and stable model for classification or regression tasks.




```
Training Accuracy: 87.50%
Test Accuracy: 86.62%
Mean Squared Error: 0.1337865302
```

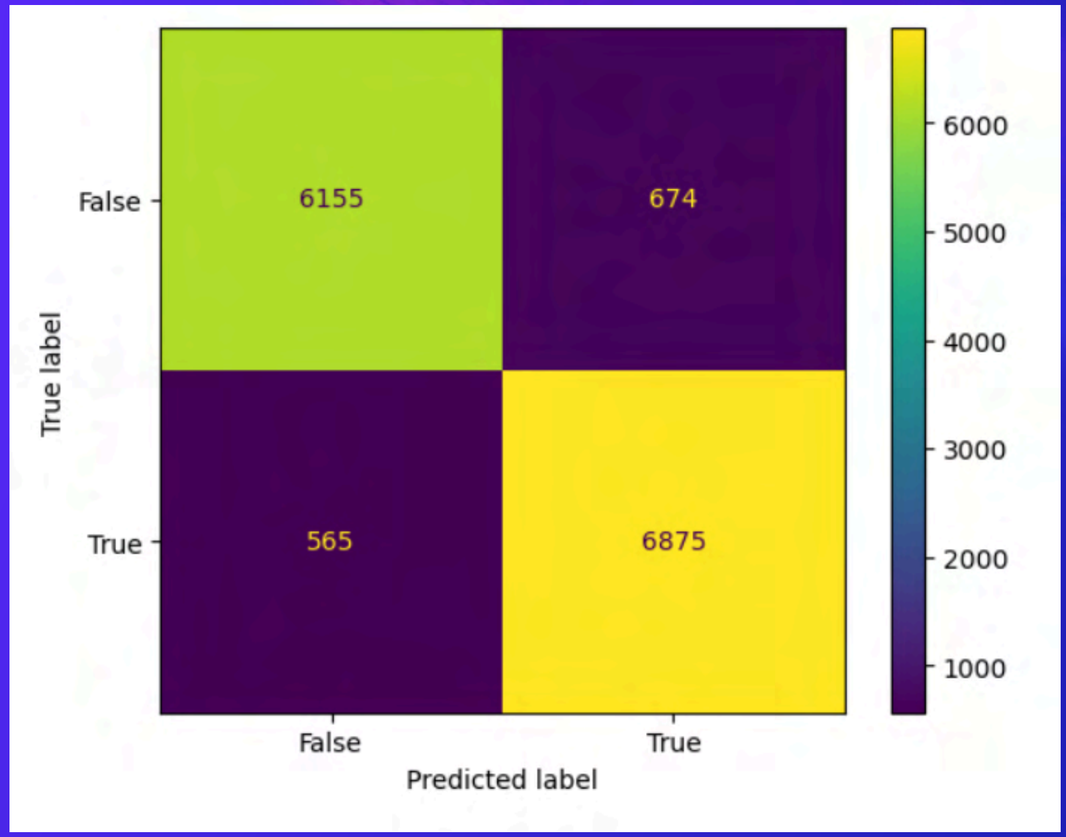
	precision	recall	f1-score
0	0.86	0.87	0.86
1	0.88	0.87	0.87

'0' (referred to as the negative class)

'1' (referred to as the positive class)

XG BOOST CLASSIFIER

XGBoost (Extreme Gradient Boosting) is a powerful machine learning algorithm known for its high performance. It's a gradient boosting framework that builds a series of decision trees sequentially, optimizing for errors in predictions. XGBoost combines the strengths of boosting and regularization, making it efficient, scalable, and capable of handling diverse data types. It minimizes prediction errors by iteratively adding weak models, optimizing their weights, and delivering accurate predictions through an ensemble of trees.



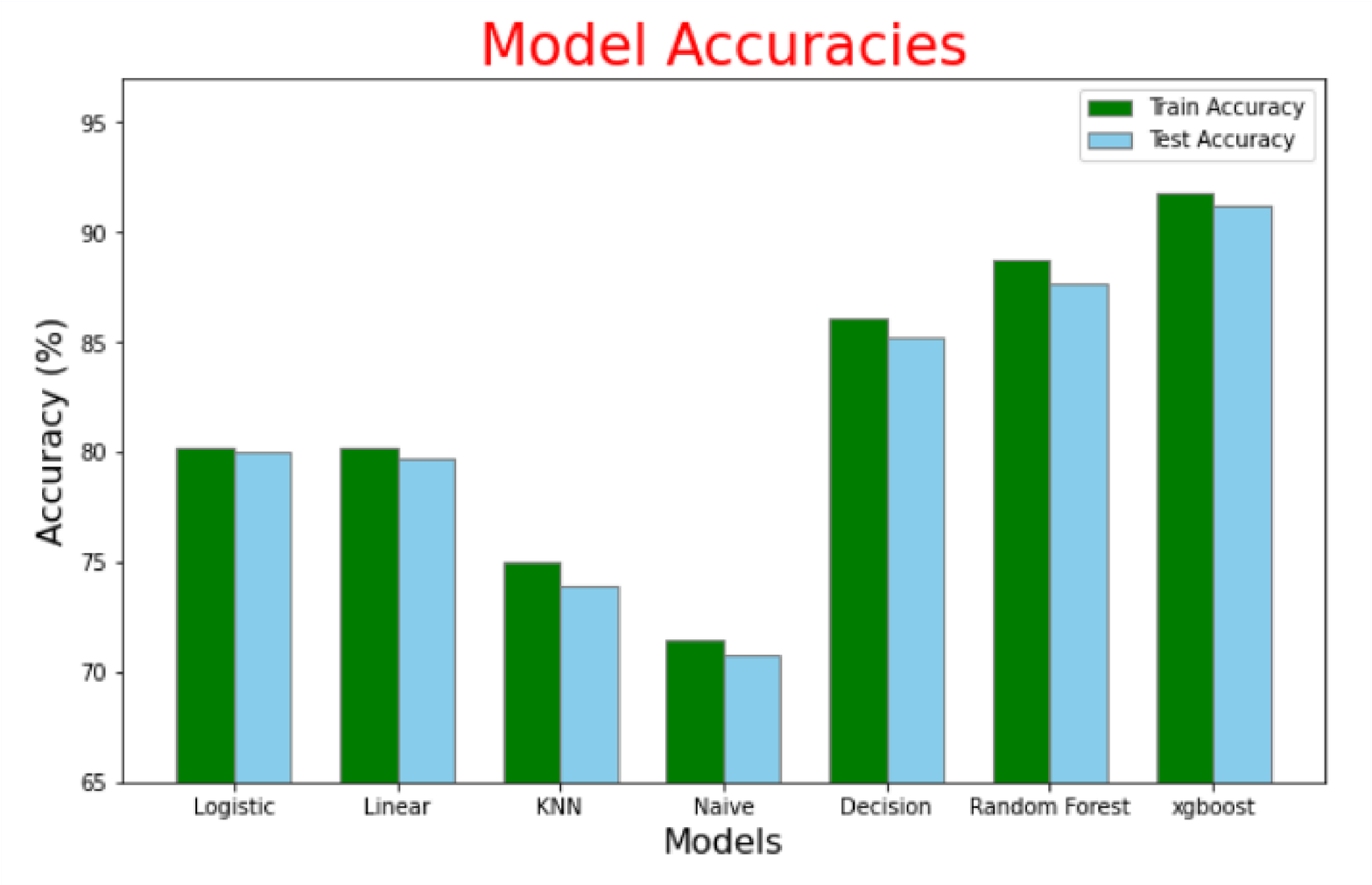

```
Training Accuracy: 91.58%
Test Accuracy: 91.32%
Mean Squared Error: 0.08683
```

	precision	recall	f1-score
0	0.93	0.90	0.91
1	0.91	0.94	0.92

'0' (referred to as the negative class)

'1' (referred to as the positive class)

GRAPHICAL REPRESENTATION OF TRAINING AND TESTING ACCURACY OF ALL MODELS



RESULTS AND ACHIEVEMENTS

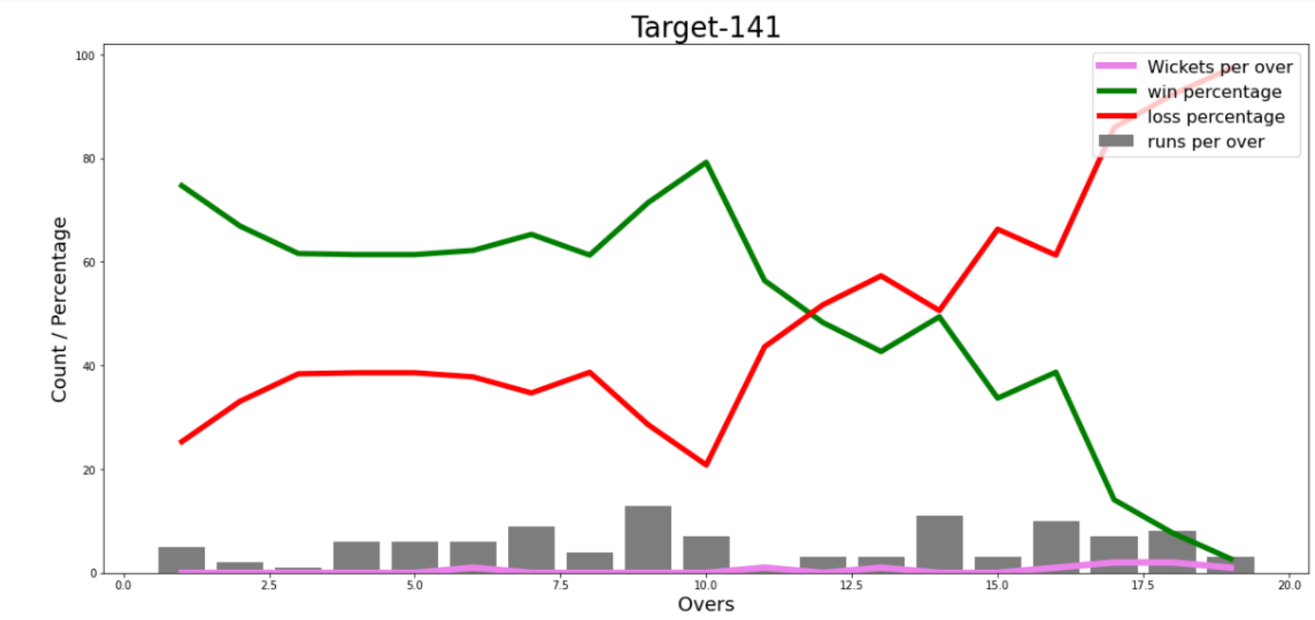
01

Target- 141

	end_of_over	runs_after_over	wickets_in_over	lose	win
69743	1	5	0	25.299999	74.699997
69749	2	2	0	33.099998	66.900002
69756	3	1	0	38.400002	61.599998
69762	4	6	0	38.599998	61.400002
69768	5	6	0	38.599998	61.400002
69774	6	6	1	37.799999	62.200001
69780	7	9	0	34.700001	65.300003
69786	8	4	0	38.700001	61.299999
69792	9	13	0	28.600000	71.400002
69798	10	7	0	20.799999	79.199997
69804	11	1	1	43.599998	56.400002
69810	12	3	0	51.700001	48.299999
69816	13	3	1	57.299999	42.700001
69823	14	11	0	50.599998	49.400002
69829	15	3	0	66.300003	33.700001
69835	16	10	1	61.299999	38.700001
69841	17	7	2	85.900002	14.100000
69847	18	8	2	92.300003	7.700000
69853	19	3	1	97.300003	2.700000

A match where the 1st inning score was 140 and target was 141 and this is the scenario of 2nd inning after every over

This is the graph of the match which include runs scored after each over, wickets fallen and both win and loss percentage of batting team



RESULTS AND ACHIEVEMENTS

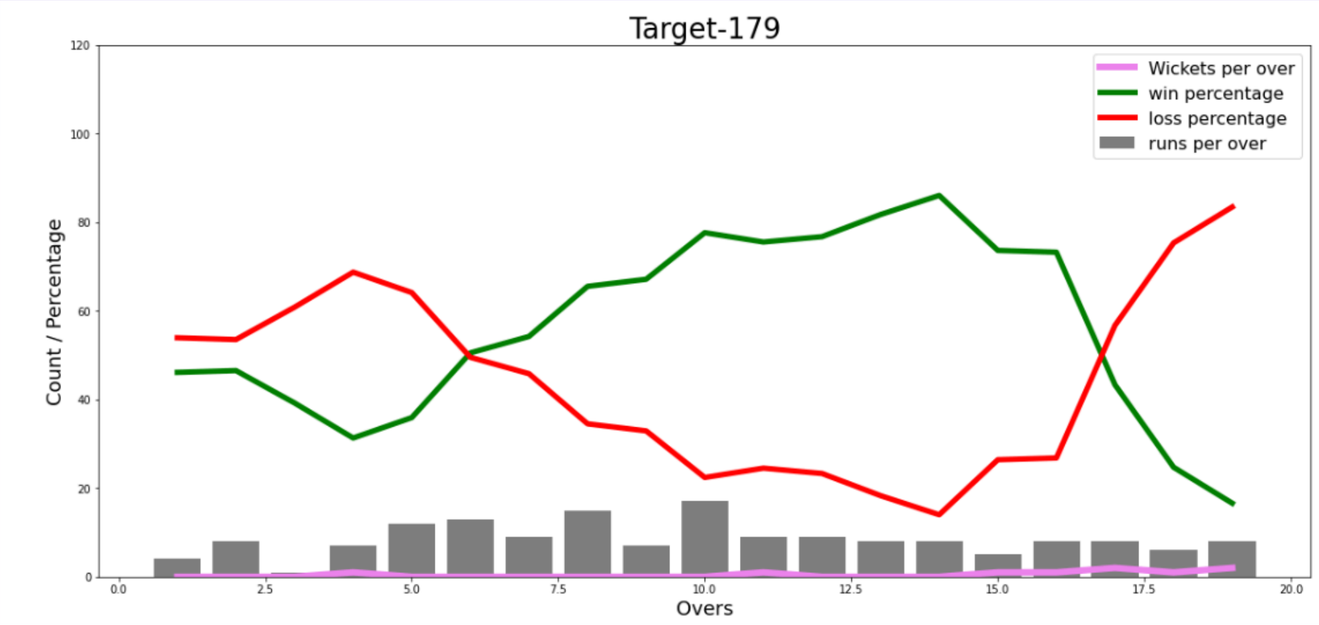
02

Target- 179

	end_of_over	runs_after_over	wickets_in_over	lose	win
10459	1	4	0	53.900002	46.099998
10467	2	8	0	53.500000	46.500000
10473	3	1	0	60.799999	39.200001
10479	4	7	1	68.699997	31.299999
10485	5	12	0	64.099998	35.900002
10491	6	13	0	49.500000	50.500000
10497	7	9	0	45.799999	54.200001
10505	8	15	0	34.500000	65.500000
10511	9	7	0	32.900002	67.099998
10518	10	17	0	22.400000	77.599998
10524	11	9	1	24.500000	75.500000
10530	12	9	0	23.299999	76.699997
10536	13	8	0	18.299999	81.699997
10542	14	8	0	14.000000	86.000000
10548	15	5	1	26.400000	73.599998
10555	16	8	1	26.799999	73.199997
10561	17	8	2	56.700001	43.299999
10567	18	6	1	75.300003	24.700001
10573	19	8	2	83.400002	16.600000

A match where the 1st inning score was 178 and target was 179 and this is the scenario of 2nd inning after every over

This is the graph of the match which include runs scored after each over, wickets fallen and both win and loss percentage of batting team



THANK YOU