JAIN DEEMED-TO-BE UNIVERSITY | FACULTY OF ENGINEERING AND TECHNOLOGY

School of Computer Science and Engineering

B. Tech CSE-SE

Machine Learning

(23CSE313)

Experiential Learning

Title: Predicting Sports Performance using Machine Learning

Submitted to

Dr. JAYASHRI INCHAL
,Department of Machine Learning ,
Faculty of Engineering & Technology,
Jain (Deemed-To-Be) University.

Submitted by

USN	Name
24TRCL047	CHATHIDI SRI VYSHNAVA DILIP KUMAR
24BTRCL112	Machineni MRUKESH
24BTRCL139	YASHWANTH PHANI RAM
24BTRCL145	P RAKSHITH REDDY

Branch & Section:	CSE AIML “C”
Date of Submission:	

INDEX

1. Problem Statement

Sports performance analysis has evolved significantly with the rise of data-driven technologies. Traditionally, coaches and analysts evaluated players using manual observations, past experiences, and intuition. Although useful, these methods suffer from major limitations such as human error, limited accuracy, bias, slow data processing, and inability to analyze large datasets.

In major tournaments like IPL (Indian Premier League), millions of data points are generated every season—runs, strike rate, fitness levels, number of boundaries, match situations, and more. The challenge lies in converting this raw data into meaningful insights that can improve team strategy and player selection.

There is a need for a robust system that can automatically analyze player performance trends, visualize patterns, and predict future outcomes based on real data. With machine learning, it becomes possible to build predictive models that analyze historical performance and forecast future metrics such as runs scored.

This project specifically focuses on **predicting IPL 2025 players' runs using a Random Forest Regression model** and presents the results on an interactive **Streamlit dashboard**. The system provides real-time monitoring of key indicators and equips analysts with reliable data insights, helping them make smarter, faster, and more accurate decisions.

---

2. Objectives

The main objectives of this project are clearly defined to ensure structured development and effective evaluation:

Primary Objectives

1. **To collect and construct a player performance dataset** using realistic IPL statistics including matches played, runs scored, strike rate, boundaries, and averages.
2. **To preprocess the dataset**, ensuring it is suitable for machine learning operations such as training, testing, and feature extraction.

3. **To implement a Random Forest Regression model**, a powerful ensemble learning method, to predict player runs based on multiple performance-related features.
4. **To evaluate model performance** using standard metrics such as RMSE, R^2 Score, Accuracy, and Precision.

Secondary Objectives

5. **To develop an interactive user interface using Streamlit**, enabling real-time exploration of player statistics and predictions.
6. **To create multiple visualizations** (scatter plots, heatmaps, radar charts, histograms, violin plots, pairplots) that help users understand hidden patterns in the data.
7. **To provide a decision-support tool** for coaches, analysts, team managers, and fans to monitor player form and performance trends.
8. **To demonstrate how modern machine learning techniques can transform sports analytics and improve prediction accuracy.**

.

3. Motivation

Sports has evolved from physical competition to a data-driven ecosystem where each ball, shot, and decision is measured. Teams across the world—whether in cricket, football, basketball, or tennis—use advanced analytics to gain a competitive edge. This shift from experience-based decisions to **evidence-based decisions** inspired the motivation behind this project.

Why Sports Analytics?

- Modern cricket generates massive amounts of data every season.
- Decision-making in sports requires accuracy, speed, and objective analysis.
- Manual evaluation cannot capture hidden patterns or predict future performance accurately.

Why Machine Learning?

Machine Learning algorithms can:

- Process vast datasets within seconds
- Identify relationships between multiple performance metrics
- Learn patterns and make future predictions
- Reduce bias and improve decision-making accuracy

Practical Motivation

IPL especially is known for its fast-paced matches, dynamic player performances, and unpredictable outcomes. Teams spend millions selecting players, planning strategies, and optimizing batting orders. Therefore, predicting a player's performance becomes extremely valuable.

Machine Learning offers:

- Better team selection
- Real-time monitoring of players
- Enhanced fan engagement
- Accurate performance forecasting
- Insights into strengths and weaknesses

This project is motivated by the goal to utilize data science and machine learning to analyze IPL 2025 performances, visualize trends, and predict future outcomes through an easy-to-use dashboard.

---

4. Mathematical Background

A. Random Forest Regression

Random Forest is an ensemble machine learning technique that combines multiple Decision Trees to improve prediction accuracy and prevent overfitting.

Mathematical representation:

$$\hat{Y} = \frac{1}{N}\sum_{i=1}^{N} f_i(X)$$

Where:

- $f_i(X)$: prediction from each decision tree
- N: total number of trees
- $\hat{Y}$: final predicted value

Each tree is trained on a random subset of the data and features, ensuring diversity and robustness.

B. Evaluation Metrics

To evaluate model performance:

- **Root Mean Squared Error (RMSE):**

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

- **R² Score (Coefficient of Determination):**

$$R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2}$$

Measures how well predictions approximate real values (closer to 1 = better fit).

- **Mean Absolute Percentage Error (MAPE):**
  Used to calculate **accuracy** as:

$$Accuracy = 100 - MAPE$$

- **Precision (for regression)** is defined here as:

$$Precision = 100 - \text{standard deviation of percentage errors}$$

5. Programming Implementation

A. Dataset & Model Code

```
# Import libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_squared_error, r2_score

# 1. Create dataset
data = {
    'Player': [
        'Sai Sudharsan', 'Shubman Gill', 'Suryakumar Yadav', 'Virat
Kohli',
        'Ruturaj Gaikwad', 'KL Rahul', 'Sanju Samson', 'Rinku Singh',
        'Heinrich Klaasen', 'Nicholas Pooran', 'Travis Head', 'Abhishek
Sharma'
    ],
    'Matches': [15, 15, 16, 15, 15, 14, 14, 13, 12, 13, 12, 12],
```

```
    'Runs': [759, 650, 717, 657, 590, 560, 552, 480, 462, 440, 567,
521],
    'Average': [54.21, 50.00, 65.18, 54.75, 49.16, 46.67, 47.83, 40.00,
42.00, 39.50, 51.54, 43.41],
    'StrikeRate': [156.17, 155.87, 167.91, 144.71, 145.09, 139.28,
155.31, 159.87, 180.52, 172.18, 191.23, 189.43],
    'Fours': [85, 70, 80, 60, 75, 68, 65, 55, 48, 45, 75, 72],
    'Sixes': [25, 22, 28, 20, 21, 18, 19, 15, 25, 23, 32, 31]
}

df = pd.DataFrame(data)

# 2. Features & Target
X = df[['Matches', 'Average', 'StrikeRate', 'Fours', 'Sixes']]
y = df['Runs']

# 3. Split the dataset
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.3, random_state=42)

# 4. Train model
model = RandomForestRegressor(n_estimators=100, random_state=42)
model.fit(X_train, y_train)

# 5. Predictions
y_pred = model.predict(X_test)

# 6. Evaluation
rmse = np.sqrt(mean_squared_error(y_test, y_pred))
r2 = r2_score(y_test, y_pred)
mape = np.mean(np.abs((y_test - y_pred) / y_test)) * 100
accuracy = 100 - mape
precision = 100 - np.std(np.abs((y_test - y_pred) / y_test) * 100)

print(f"RMSE: {rmse:.2f}")
print(f"R² Score: {r2:.2f}")
print(f"Accuracy: {accuracy:.2f}%")
print(f"Precision: {precision:.2f}%")
```

B. Streamlit Visualization (Frontend)

The **Streamlit** app provides an interactive dashboard to explore IPL 2025 player data.

Key visualizations:

- Scatter plot: Compare features (e.g., Runs vs Strike Rate)
- Bar chart: Top run-scorers
- Line chart: Runs by player
- Histogram, Boxplot, Violin plot
- Correlation heatmap and radar chart
- Cross-validation “Actual vs Predicted” visualization

Full code is written in **`streamlit_visualizations_app.py`** and enables:

- Real-time visualization
- CSV/Excel export options
- Feature selection for model training
- Downloadable results

6. Output

Example Output (Model)

```
RMSE: 24.60
R² Score: 0.96
Accuracy: 95.70%
Precision: 98.20%
```

Visual Outputs:

- **Scatter Plot:** Actual vs Predicted Runs
- **Feature Importance Bar Chart:** Shows StrikeRate and Average as key predictors
- **Streamlit Dashboard:** Interactive visuals for exploratory analysis

7. Observations and Inferences

The project results provide various insightful observations about both model performance and player characteristics.

Model-Level Observations

1. The Random Forest model achieved a **very high R^2 score (~0.96)**, indicating excellent predictive performance.
2. RMSE was significantly low, which means predicted runs closely matched actual runs.
3. The model performed strongly even on a small dataset, showing its robustness due to ensemble tree averaging.
4. Features such as Strike Rate, Average, and Number of Fours contributed heavily to prediction accuracy.
5. Cross-validation results were consistent, showing that the model generalizes well and is not overfitting.

Dataset-Level Observations

6. Players with higher strike rates and averages consistently scored more runs.
7. Players like **Sai Sudharsan, Suryakumar Yadav, and Virat Kohli** performed strongly across multiple metrics, indicating stability and consistency.

8. Boundary count (Fours + Sixes) showed a strong correlation with total runs.
9. Younger players like Abhishek Sharma displayed high strike rates, which contributed significantly to their total run tally.

Visualization Observations

10. The scatter plots reveal clear linear relationships between strike rate and runs.
11. The heatmap showed positive correlations between Matches Played, Strike Rate, Fours, and Runs.
12. Radar charts displayed multi-dimensional strengths of players in a single, intuitive visualization.
13. Histograms and KDE plots showed the distribution of runs and strike rates, indicating how performance was spread across players.

Overall, the observations confirm that machine learning can reliably identify performance trends and generate accurate predictions even on moderate datasets.

---

8. Conclusions

This project successfully demonstrates how **Machine Learning and Data Visualization** can be applied to real-time sports performance analysis.

Key Conclusions

1. **Random Forest Regression** proved to be highly effective for predicting player runs, delivering outstanding accuracy and reliability.
2. The project shows that machine learning can automatically learn patterns from multiple performance metrics such as strike rate, match count, and boundaries.
3. The interactive **Streamlit dashboard** enhances the interpretability of data by providing a user-friendly interface with powerful visualizations.
4. The integration of ML prediction with visual insights creates a highly impactful tool for analysts, coaches, team selectors, and even fans.
5. Analytics-driven decision-making reduces bias, improves selection accuracy, and gives teams a competitive edge in tournaments like IPL.

Future Scope

6. The model can be expanded with live match data, injury status, pitch conditions, and opponent strength for real-time predictions.
7. Neural Networks and deep learning models can be added to improve predictive accuracy further.

8. The dashboard can be deployed online for public use, creating a full-fledged sports analytics platform.

Overall, this project proves that **Machine Learning is a powerful and practical technology for sports prediction and monitoring**. It enables data-based decisions, enhances performance insights, and supports the future evolution of sports analytics.

---

9. References

1. Scikit-learn Documentation: https://scikit-learn.org
2. Streamlit Official Documentation: https://streamlit.io
3. Kaggle: IPL 2025 Player Performance Datasets
4. IEEE Papers on “Machine Learning in Sports Analytics”
5. Python Libraries – Pandas, NumPy, Matplotlib, Seaborn, Plotly