

# SCORECRAFTERS

## SCORE PREDICTOR WITH PLAYER AND TEAM ANALYSIS

### **Problem Statement:**

A cricket score prediction system integrating a score predictor, individual player analysis and team performance metrics. Develop a user-friendly interface where users input live match details, including batting and bowling team, current score, overs, wickets, and runs in the past 5 overs. The system will offer insights into player statistics, featuring batting average, bowling average, and other relevant metrics. Additionally, implement team analysis, showcasing top 5 batsmen and bowlers, along with overall team performance visualizations. Build a machine learning model to predict the final score based on match-specific parameters and evaluate its accuracy.

### **The project includes:**

#### 1. Score Prediction Model:

- Developed a machine learning model that takes the match-specific inputs (batting team, bowling team, current score, overs, wickets, runs in past 5 overs) and predicts the final score depending on historical match data.

#### 2. Individual Player Analysis:

- Collected and analysed data for each player, including batting and bowling statistics and generated visualizations.

#### 3. Team Analysis:

- Implemented a feature where users can select a team, and the system displays the top 5 batsmen and top 5 bowlers for that team and provide visualizations.

## Goal:

The goal of this project is to provide cricket enthusiasts with a tool that not only analyses player and team performance but also predicts the likely outcome of a match based on the current game state.

## Use Cases

- Live Match Predictions, Team Strategy, Fan Engagement, Coaching and Training, Performance Assessment.

## Sources of your data:

<https://www.kaggle.com/datasets/veeralakrishna/cricsheet-a-retrosheet-for-cricket>

Structured ball-by-ball data for international and T20 League cricket matches.

The data was scraped from Kaggle. All data in the file is publicly available to everyone already. Please be noted that data is originally owned by Kaggle.

The dataset yaml file contains 1432 files of match data. The description of each data in each yaml field is as below:

Dates: The match conducted date

Gender: total male 966 and Female 466 matches

Match type: T20

Overs: 20

Player of match

Teams: The team played that match

venue: Harare Sports Club

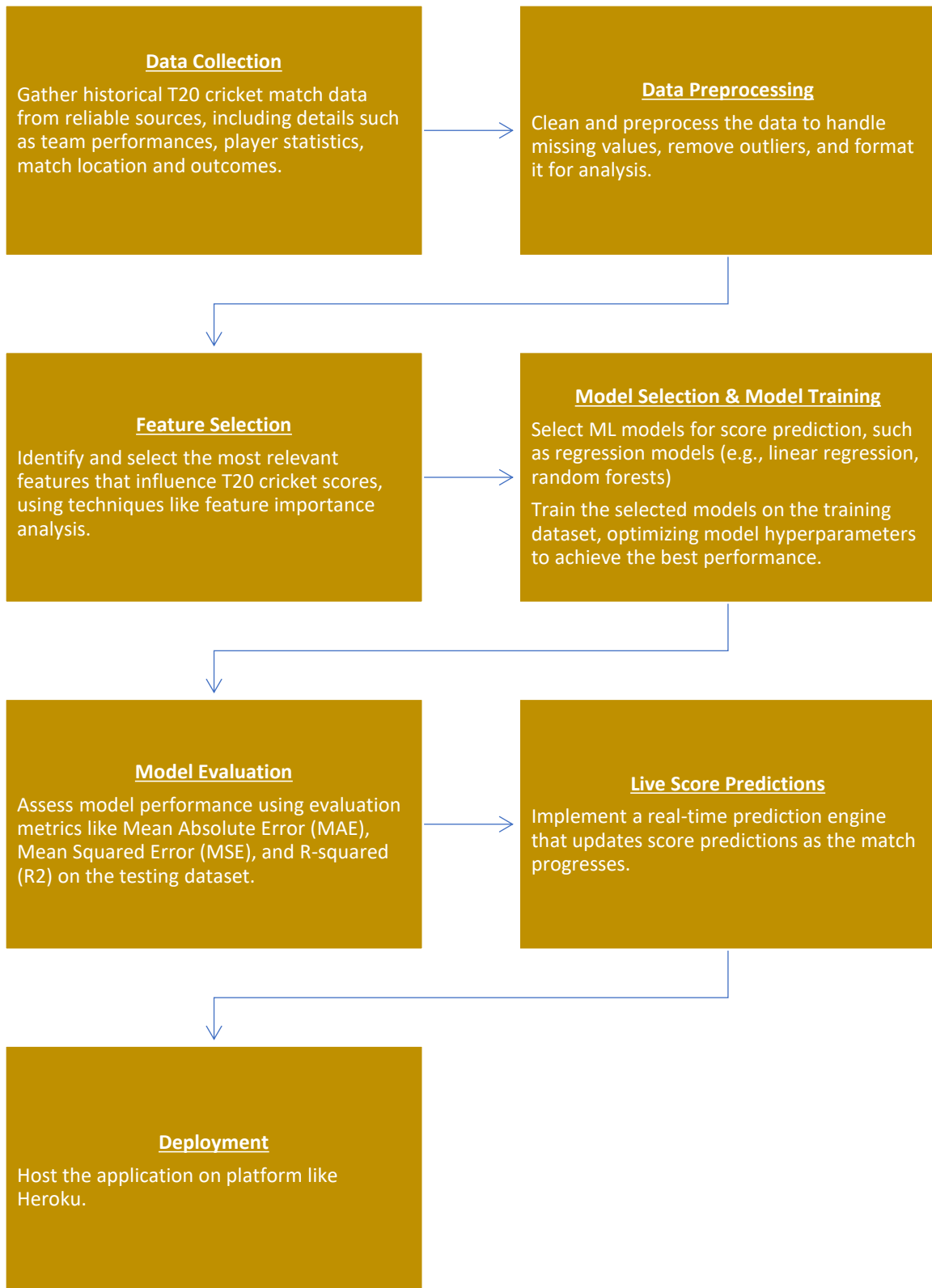
Innings: {1st innings: team: Zimbabwe

deliveries: each ball data {batsman: The name of the batsman

bowler: The name of the bowler

runs: Runs scored by the player by that ball}}

## PROJECT WORKFLOW



## Setup and Installation:

- Pandas
- Numpy
- Matplotlib
- Seaborn
- Xgboost
- Django
- Python

## Conclusion:

In summary, the project aimed to predict cricket scores using machine learning models based on a variety of input parameters, including individual player metrics and team performance. Here are the key findings and achievements:

### 1. Model Development:

- Developed machine learning models for score prediction, experimenting with various algorithms.
- Explored different model architectures.

Model	MSE	MAE	R2
Linear Regression	186.981	9.42245	0.592645
Polynomial	38.5906	3.80802	0.922588
Decision Tree	31.4541	0.934316	0.938685
Random Forest	9.97211	1.51429	0.98107
GradientBoosting	19.8164	2.4026	0.964971
SVR	38.9529	2.67269	0.93623
KNN	43.6981	2.65345	0.910901
XGBRegressor	6.85371	1.05628	0.988764

### 2. Individual Player Analysis:

- Successfully conducted a comprehensive analysis of individual player metrics, including batting and bowling averages, strike rates, wickets taken, and other relevant statistics.
- Presented the insights through visually appealing graphs, providing a clear understanding of player performance.

### 3. Team Analysis:

- Implemented team analysis, allowing users to select a team and view the top 5 batsmen and bowlers, along with additional graphical representations of team performance.

#### 4. Challenges and Lessons Learned:

- Data Quality: Dealing with inconsistent or incomplete data was a challenge. Lessons learned include the importance of data pre-processing and the impact it has on model performance.
- Model Complexity: Striking the right balance between model complexity and interpretability was challenging. It was crucial to ensure that the model not only performed well but also provided insights that could be understood by both technical and non-technical users.
- Feature Selection: Identifying the most influential features for prediction was a learning process. Iterative feature engineering and model evaluation were essential for refining the feature set.

#### **References :**

Python packages : Matplotlib, Pandas , Sklearn

StackOverflow for debugging code

Popular blogs such as Towards Data Science,Digital Ocean

Youtube channels : Krish Naik ,CodeBasics ,StatQuest etc