# **Machine Learning Project Task Document**

## **Project Title: IPL Score Prediction**

### **1. Project Overview**

This project aims to predict the total score of a cricket team at the end of their innings during an IPL (Indian Premier League) match. The model is built using historical data from IPL seasons 1 to 9 (2008-2016) and tested on data from season 10 (2017). The model uses various machine learning algorithms to predict the score based on input features such as the batting and bowling teams, overs, runs, wickets, and other match statistics.

### **2. Data Understanding**

#### **2.1 Data Collection**

The [dataset](https://drive.google.com/file/d/17I0Bb0i6UNq2PtI6Wtz0LRJFS1QkhVut/view?usp=sharing) used for this project is an IPL dataset (ipl.csv) containing match statistics for each ball bowled in a match. The dataset includes:

* **Batting team**
* **Bowling team**
* **Overs** (current over in the match)
* **Runs** scored in that over
* **Wickets** taken in that over
* **Runs in last 5 overs**
* **Wickets in last 5 overs**
* **Total score** of the batting team at the end of the innings
* **Date of the match**

#### **2.2 Data Exploration**

* The dataset contains multiple columns such as match ID, venue, batsman, bowler, striker, non-striker, etc.
* We perform basic data exploration, such as checking the shape of the dataset, data types, and the first few records, to understand the structure of the data.

### **3. Data Preprocessing**

#### **3.1 Data Cleaning**

* **Removing Unwanted Columns**: Columns that are not relevant for predicting the score, such as match ID, venue, batsman, bowler, striker, and non-striker, are removed.
* **Filtering Consistent Teams**: Only matches involving consistent teams (e.g., Chennai Super Kings, Mumbai Indians, etc.) are retained.
* **Removing Early Match Data**: Data from the first 5 overs of each match is removed to focus on more relevant match situations.
* **Date Conversion**: The 'date' column is converted from string format to a datetime object for easier analysis.

#### **3.2 Feature Selection**

* **Correlation Heatmap**: A heatmap is used to visualize the correlation between the features, ensuring the most relevant features are selected for the model.

#### **3.3 Encoding Categorical Features**

* **One-Hot Encoding**: The categorical variables, bat\_team and bowl\_team, are encoded into numerical features using one-hot encoding. This creates binary columns for each team, representing whether a team is batting or bowling.

#### **3.4 Splitting Data into Training and Testing Sets**

* The data is split into a **training set** (seasons 1-9, i.e., 2008-2016) and a **test set** (season 10, i.e., 2017). This ensures that the model is trained on historical data and tested on more recent data.

### **4. Model Building**

#### **4.1 Models Used**

The following regression models are used to predict the IPL match scores:

* **Linear Regression**
* **Decision Tree Regression**
* **Random Forest Regression**
* **AdaBoost Regressor** (with Linear Regression as the base learner)

#### **4.2 Model Training and Evaluation**

* The models are trained using the training dataset and evaluated using the test dataset.
* The evaluation metrics include:
  + Mean Absolute Error (MAE)
  + Mean Squared Error (MSE)
  + Root Mean Squared Error (RMSE)

#### **4.3 Model Comparison**

* After evaluating the models, it was found that **Linear Regression** outperformed other models, with the lowest error metrics.

### **5. Model Optimization with AdaBoost**

#### **5.1 AdaBoost with Linear Regression**

* The AdaBoost algorithm is applied to the **Linear Regression model** to improve its performance. However, it did not significantly reduce the error compared to the base Linear Regression model.

#### **5.2 Final Model Selection**

* **Linear Regression** is chosen as the final model due to its simplicity and better performance compared to other models.

### **6. Predictions**

#### **6.1 Predicting IPL Scores**

The final model (Linear Regression) is used to predict the total score of a team at the end of their innings for various matches in IPL seasons 11 and 12 (2018-2019). The prediction function takes the following inputs:

* **Batting team**
* **Bowling team**
* **Overs**
* **Runs scored**
* **Wickets taken**
* **Runs in the last 5 overs**
* **Wickets in the last 5 overs**

#### **6.2 Example Predictions**

The model makes predictions for various matches in IPL seasons 11 and 12, such as:

1. **Kolkata Knight Riders vs. Delhi Daredevils** (2018, Match 13)
   * Predicted score: 200/9
2. **Sunrisers Hyderabad vs. Royal Challengers Bangalore** (2018, Match 39)
   * Predicted score: 146/10
3. **Mumbai Indians vs. Kings XI Punjab** (2019, Match 59 - Eliminator)
   * Predicted score: 186/8
4. **Rajasthan Royals vs. Chennai Super Kings** (2019, Match 25)
   * Predicted score: 151/7

### **7. Conclusion and Future Work**

#### **7.1 Conclusion**

* The project successfully builds a predictive model for IPL scores based on various match-related features.
* Linear Regression provided the best performance in predicting the score compared to other models.

#### **7.2 Future Work**

* **Model Improvement**: Further optimization of the model, such as feature engineering and hyperparameter tuning, could potentially improve the accuracy.
* **More Data**: Including more data from newer IPL seasons could help the model generalize better.
* **Other Models**: Experimenting with advanced machine learning models such as XGBoost or Neural Networks might further enhance prediction accuracy.

### **8. Tools and Libraries Used**

* **Programming Language**: Python
* **Libraries**: pandas, numpy, scikit-learn, seaborn, matplotlib