IPL Match Winner Prediction

The IPL dataset contains detailed information about cricket matches played in the Indian Premier League (IPL). Key columns include:

- **Season**: The year of the IPL season.
- **City**: The city where the match was played.
- **Date**: The date of the match.
- **team1** and **team2**: The two teams playing in the match.
- **toss_winner** and **toss_decision**: The team that won the toss and the decision they made (whether to bat or bowl).
- **result**: The outcome of the match (win or loss).
- **dl_applied**: Indicates if the Duckworth-Lewis method was applied due to interruptions.
- **winner**: The team that won the match.
- **win_by_runs** and **win_by_wickets**: The margin of victory by runs or wickets.
- **player_of_match**: The player awarded for exceptional performance.
- **venue**: The stadium where the match was held.
- **umpires**: The officials overseeing the match.

Objective: The aim of this project is to predict the winner of an IPL match based on various features such as the teams playing, toss winner, toss decision, and other match-related factors using different machine learning models like Logistic Regression, SVM, KNN, Decision Trees, Random Forest, and XGBoost. The best model must be selected after evaluating the performance and hyperparameters must be tuned for further improvements.

Instructions:

1. **Data Loading and Understanding:**
   - Load the IPL dataset into a suitable data structure.
   - Display the first few records to understand the data format.
   - Identify the columns and their data types.
   - Document any observations about the dataset.
2. **Data Cleaning:**
   - Handle missing or null values in the dataset.
   - Convert categorical features (like team names, toss decisions) into numerical values using techniques like label encoding or one-hot encoding.

   - Identify and remove any outliers or inconsistent data.
   - Document the data cleaning process.

3. **Exploratory Data Analysis (EDA):**
   - Visualize key patterns and relationships in the data using plots (e.g., bar charts, histograms, pie charts, etc.).
   - Explore correlations between features and the target variable (winner).
   - Create visualizations for team performance, toss outcomes, and other relevant statistics.
   - Document your observations from the visualizations.
4. **Data Preprocessing:**
   - Split the dataset into training and testing sets (e.g., 80% training and 20% testing).
   - Scale/normalize numerical features if required.
   - Prepare feature sets (X) and target labels (y).
5. **Model Training:**
   - Train multiple machine learning models (Logistic Regression, SVM, KNN, Decision Trees, Random Forest, XGBoost).
   - Evaluate the performance of each model using appropriate metrics like accuracy, precision, recall, and F1-score.
6. **Model Evaluation:**
   - Compare the performance of each model using metrics such as accuracy, confusion matrix, precision, recall, and F1 score.
   - Select the best-performing model based on evaluation results.
7. **Hyperparameter Tuning:**
   - Use grid search or random search to find the optimal hyperparameters for the best-performing model.
   - Tune the model for better performance.
8. **Model Comparison:**
   - Compare the performance of all models and justify which model performs the best. Save the final model in .pkl file.
   - Document the final model and its performance.
9. **Conclusion:**
   - Discuss how the model can be improved further.

  - Suggest additional features or data that can enhance the model.
  - Mention any challenges faced and how they were overcome.

10. **Documentation:**
    - Keep a detailed record of each step, observations, and decisions made throughout the project.
    - Ensure the documentation is clear and well-organized.
11. **Presentation:**
    - Prepare a PowerPoint presentation summarizing the entire project.

PowerPoint Presentation (PPT) Inputs:

1. **Title Slide:**
    - Project title, student's name, and date.
2. **Introduction:**
    - Overview of the problem.
    - Brief explanation of the dataset.
    - Objective of the project.
3. **Data Understanding:**
    - Description of the dataset and its columns.
    - Sample data rows.
4. **Data Cleaning:**
    - Steps taken for cleaning the data.
    - Handling missing values, encoding categorical features, etc.
5. **Exploratory Data Analysis (EDA):**
    - Key visualizations (graphs, charts).
    - Insights drawn from the data exploration.
6. **Modeling Process:**
    - List of models used (Logistic Regression, SVM, KNN, etc.).
    - Hyperparameter tuning approach (Grid Search or Random Search).
    - Evaluation metrics (accuracy, precision, recall, etc.).

7. **Model Comparison:**
    - Performance comparison of each model (charts/tables).
    - Best-performing model and justification for selection.
8. **Conclusion:**
    - How the model can be improved.
    - Possible future steps.
9. **Q&A Slide:**
    - Open floor for any questions.

Deliverables:

- **Project Documentation**: Detailed report documenting the process with necessary explanations and screenshots.
- **Code/Notebooks**: Jupyter notebooks or Python scripts containing all the code for data loading, cleaning, EDA, model training, and evaluation.
- **PowerPoint Presentation**: A well-structured presentation summarizing the project, findings, models used, and results.
- **Final Model**: The trained and tuned model saved in a usable format (e.g., .pkl file).

Evaluation Rubric:

Evaluation Criteria	Marks
Data Understanding and Cleaning	15
Exploratory Data Analysis (EDA)	20
Modeling and Model Selection	10
Hyperparameter Tuning	10
Model Comparison and Final Model Selection	10
Conclusion and Suggestions	5
Documentation	10
Presentation	20
Total	**100**

Note: Ensure to include screenshots in the documentation and presentation wherever applicable.