Basaveshwar Engineering College, Bagalkote

[An Autonomous Government Aided College, AICTE approved, Affiliated to VTU, Belagavi]

SYNOPSIS FOR MINI PROJECT (22UAI506P)

Department of Artificial Intelligence & Machine Learning. [2024-25]

TITLE:
PLAYER ROLE CLASSIFICATION USING ML MODELS

Submitted By		
Sl No.	**Name**	**USN**
01	Keerti Nandi	2BA22AI013
02	Ibrahim Indikar	2BA22AI010
03	Prashant R H	2BA22AI025
04	Vivek S Hosur	2BA22AI052

Project Guide	Project co-ordinator	Head of Department
(Dr. Bharati M. Reshmi)	(Prof. Lakshmi P. Kolur)	(Dr. A. D. Devanagavi)

ABSTRACT

In cricket, player roles such as Batsman, Bowler, and All-Rounder are fundamental to team composition and match strategy. Traditionally, player roles are assigned based on qualitative assessments and subjective judgments, which may lead to inconsistencies. The Player Role Classification project aims to automate this classification process using machine learning. By analyzing key player performance metrics, including batting average, strike rate, wickets taken, and economy rate, this project seeks to build a model that accurately categorizes players into their respective roles.

This project utilizes data collection from reliable sources, followed by preprocessing and exploratory data analysis (EDA) to uncover trends and patterns that define each player role. Several machine learning models, including Decision Trees and Random Forests, are then trained and evaluated to identify the most accurate classification model. Evaluation metrics such as accuracy, precision, and F1-score are used to assess model performance, while feature importance analysis provides insights into the metrics most indicative of each role.

The outcomes of this project will enable coaches, selectors, and analysts to make data-driven decisions regarding team selection and player development. By providing a structured, automated approach to role classification, this project offers a valuable tool for optimizing cricket strategies and enhancing player utilization. This model can potentially be deployed as an application or API, providing real-time classification capabilities for cricket enthusiasts and professionals alike.

*****

INDEX

INTRODUCTION

Cricket is a sport characterized by diverse player roles, each requiring unique skills and contributing differently to team performance. Broadly, players are categorized into three main roles:

- Batsman : Primarily focused on scoring runs.
- Bowler : Specializes in taking wickets and controlling the opposition's scoring rate.
- All-Rounder : Proficient in both batting and bowling, making versatile contributions to the team.
- Wicketkeeper : Wicketkeepers are integral to the team's defense, often serving as both skilled batsmen and agile fielders.

Accurate classification of these roles is vital in forming balanced teams, optimizing match strategies, and identifying areas for player improvement. Traditionally, player roles are determined based on qualitative insights, which may be subjective. This project aims to use machine learning to automate and standardize the player role classification process by analyzing player performance metrics. By leveraging key statistical data such as batting averages, strike rates, economy rates, and wickets taken, this project will develop a model that can accurately classify players into Batsman, Bowlers, and All-Rounders.

PROBLEM STATEMENT

In cricket, selecting the right combination of players for a team is crucial for achieving a balanced lineup and effective game strategy. Each player's role whether Batsman, Bowler, All-Rounder, or Wicketkeeper contributes differently to the team's overall performance. Traditionally, these roles are assigned based on a mix of qualitative judgments from coaches, selectors, and analysts. However, such manual classification may lead to inconsistent results due to subjective interpretation, potential biases, and varying criteria for role assignment across formats.

With the growth of cricket analytics and the availability of large datasets covering various player statistics, there is an opportunity to create a data-driven approach to player role classification. A well-structured, automated system can help classify players accurately and objectively, based on quantifiable performance metrics, such as batting and bowling averages, strike rates, economy rates, and other statistics. This can ensure more consistent role assignment, reduce human error, and provide insights that may not be immediately visible through traditional analysis.

The Player Role Classification project aims to solve this problem by developing a machine learning model that can classify players into one of the primary roles: Batsman, Bowler, All-Rounder, or Wicketkeeper. This model will analyze key performance metrics and use statistical patterns to assign each player to a role. By automating this classification process, we can enable coaches, analysts, and team selectors to make more informed, data-backed decisions when building teams, developing game strategies, or identifying potential areas for player improvement.

OBJECTIVES

1) **Classify Player Roles:**

- Build a model that can correctly identify a player's role based on their game data.

2) **Support Coaching Decisions:**

- Provide coaches with insights on player roles to help in strategy and training.

3) **Make the Model Easy to Understand:**

- Ensure the model's decisions are explainable to coaches and analysts.

4) **Adapt to Different Game Situations:**

- Create a model that works well across various game scenarios.

5) **Allow for Real-Time Role Updates:**

- Aim for a system that could provide player role updates during live matches.

6) **Contribute to Sports Analytics:**

- Add value to sports analytics by creating a useful tool for player classification.

METHODOLOGY

1} Data Collection:

- **Sources**: Use Kaggle datasets, CricAPI, and ESPNcricinfo to obtain player statistics.
- **Dataset Structure**: Collect data fields like player name, role (target label), runs, batting average, strike rate, wickets, bowling average, and economy rate.
- **Formats**: If available, consider gathering data across different formats for versatility.

2} Data Preprocessing:

- **Handling Missing Values**: Use methods such as mean/mode imputation for missing numerical data or remove players with incomplete data if necessary.
- **Outlier Detection**: Identify and manage outliers, which could skew model performance, particularly in features like runs or economy rate.
- **Normalization**: Standardize numerical features like batting and bowling metrics to ensure uniformity.

3} Exploratory Data Analysis (EDA):

- **Data Visualization**: Use visualization techniques (e.g., histograms, box plots) to analyze the distributions of batting and bowling metrics.
- **Correlations**: Identify correlations between features, such as the relationship between wickets taken and economy rate.
- **Feature Engineering**: Derive additional features, such as Batting/Bowling Ratio, which might help in distinguishing all-rounders from specialists.

4} Model Selection and Training:

- **Algorithms**: Experiment with multiple classification algorithms, including:
  - **Logistic Regression** for initial benchmarks.
  - **Decision Trees** for easy interpretability.
  - **Random Forests** for better generalization and handling feature importance.
- **Train-Test Split**: Use an 80-20 split for training and testing, and perform K-Fold Cross-Validation to improve robustness.

5} Model Evaluation and Optimization:

- **Metrics**: Evaluate models using accuracy, precision, recall, and F1-score.
- **Hyperparameter Tuning**: Use Grid Search or Randomized Search to fine-tune model parameters.
- **Confusion Matrix Analysis**: Analyze misclassified players to improve the model and understand any persistent classification issues.

6} Feature Importance Analysis:

- **Feature Ranking**: For models like Random Forest, examine feature importance scores to determine which metrics (e.g., runs, strike rate, economy rate) are most indicative of player roles.
- **Insights Generation**: Use these feature importance scores to generate insights for each role. For instance, high runs and batting average may indicate a Batsman, while a low economy rate could be characteristic of a Bowler.

Predicted Outcomes

- **Classification Model**: A machine learning model capable of accurately classifying players into Batsmen, Bowlers, and All-Rounders based on statistical data.
- **Data-Driven Insights**: Insights into the key statistical features that distinguish each role, helping team management make more informed decisions.
- **Role Prediction for Future Players**: The ability to use the model for real-time classification of new or upcoming players, potentially aiding scouting efforts.
- **Potential Deployment**: An optional interactive user interface or API that provides an accessible tool for role classification.

This project will serve as a foundation for applying machine learning in sports analytics and has the potential to be extended for additional roles (e.g., wicketkeepers) or formats (e.g., domestic leagues).

HARDWARE AND SOFTWARE REQUIREMENTS

1} Hardware:

- Standard computing device with a minimum of 8GB RAM and a decent CPU. For larger datasets or complex models, access to GPU resources can be beneficial.

2} Software:

- **Programming Language**: Python 3.x
- **Libraries**:
  - Pandas and NumPy for data manipulation and numerical operations.
  - scikit-learn for machine learning model development.
  - Matplotlib and Seaborn for data visualization and EDA.
  - Flask or Django (optional) for developing a web interface if deploying the model.
- **Environment**: Jupyter Notebook or Google Colab for experimentation and interactive development.

REFERENCES

- Kaggle hosts a wide variety of cricket-related datasets contributed by the data science community. These datasets include historical and recent player statistics, batting and bowling metrics, as well as team-level data. For this project, Kaggle datasets can provide an extensive set of pre-compiled and clean player statistics, which is essential for training the machine learning model.
- Kaggle datasets are especially useful for this project because they offer structured data that can be directly used in the data preprocessing phase.
- Dataset :- Player_Role

*****