Cricket Score Prediction

Fahad Waseem
FAST School of Computing
National University of Computing and Emerging Sciences
Lahore, Pakistan
l201134@lhr.nu.edu.pk

Muhammad Haris Hassan
FAST School of Computing
National University of Computing and Emerging Sciences
Lahore, Pakistan
l202067@lhr.nu.edu.pk

Ahsan Malik
FAST School of Computing
National University of Computing and Emerging Sciences
Lahore, Pakistan
l201053@lhr.nu.edu.pk

***Abstract*— This research endeavors to explore the nuanced world of cricket player statistics through a comprehensive data science approach. The journey begins with meticulous data wrangling, where a myriad of preprocessing techniques is applied to refine a complex dataset with 154 columns down to a focused and standardized subset. Subsequently, an in-depth exploratory data analysis unveils patterns in batting styles, national distributions, and performance metrics, providing valuable insights into the cricketing landscape. Building upon the distilled dataset, a neural network model is crafted using TensorFlow and Keras to predict player runs based on key metrics. The model undergoes rigorous training and evaluation, demonstrating its efficacy in capturing the intricate relationships within the dataset. The research amalgamates data wrangling, exploratory analysis, and machine learning to unveil the multifaceted dimensions of cricket statistics, offering a holistic understanding and predictive insights.**

I. Introduction

Cricket, a quintessential sport revered for its strategic intricacies and profound individual performances, finds itself in the ever-evolving landscape of data-driven insights. In this research endeavor, we embark on a meticulous exploration into the rich repository of cricket player statistics, employing a multifaceted approach encompassing data wrangling, exploratory data analysis (EDA), and advanced machine learning techniques. The symphony of data science techniques deployed in this study seeks not merely to dissect raw numbers but to unravel the intricate narratives concealed within the cricketing dataset.

The initial phase of our inquiry involves an exhaustive data wrangling process, an essential prelude to any meaningful analysis. A dataset boasting 154 columns undergoes a transformative journey, whereby extraneous variables are excised, missing data is judiciously managed, and numerical features are meticulously scaled. This distillation process, akin to curating a fine mosaic, aims to foster data homogeneity, ensuring a solid foundation for subsequent analyses.

The subsequent chapter of our investigation unfolds through the lens of exploratory data analysis, a crucial overture to deciphering the nuanced patterns underpinning cricket player statistics. From delineating the distribution of batting styles to delineating the geographical diversity of players, the EDA phase transcends the realm of numbers, offering a narrative that contextualizes the performance metrics within the broader dynamics of the cricketing world. Each visual representation becomes a tableau, telling stories of players, their styles, and the countries they represent.

This narrative crescendos with the implementation of a neural network model, a sophisticated foray into machine learning tailored to predict runs based on pivotal metrics. The model, constructed using TensorFlow and Keras, undergoes meticulous training and validation, serving as a testament to the fusion of statistical rigor and technological innovation. The result is a predictive engine that not only encapsulates the essence of player performance but also underscores the potential for artificial intelligence to complement traditional cricketing insights.

In essence, this research aspires to transcend the conventional boundaries of cricket statistics, offering a holistic narrative that marries traditional cricketing knowledge with the transformative power of data science. Through a prism of meticulous data preparation, exploratory insights, and predictive modeling, this study seeks not only to inform but to elevate our understanding of the sport's multifaceted dimensions.

II. Dataset Wrangling

A. Data Loading and Selection

The research begins by loading a comprehensive cricket players dataset into a pandas DataFrame. This dataset contains 154 columns, and the initial focus is on the batting and bowling data. Using Python, specific columns related to batting and bowling (columns 14 and onwards) are selected. The data in these columns is converted to a numeric format, with non-numeric values replaced by NaN, ensuring consistency in data types.

B. Filtering and Cleaning

To refine the dataset, a condition is established to filter rows where at least one of the batting or bowling metrics is greater than zero. This ensures that only players with meaningful statistical records are considered. Subsequently, irrelevant columns such as 'Date_of_death' are dropped. Missing values in critical columns like 'Birthdate' and 'Birthplace' are handled, and columns with a substantial number of missing values are removed to maintain data quality.

C. Data Scaling

To standardize the numeric data, a Min-Max scaling process is applied. This transformation brings all numeric values within a uniform scale, typically between 0 and 1. This step ensures that variations in numerical features do not disproportionately influence downstream analyses.

D. Feature Selection and Renaming

Certain columns deemed unnecessary for the specific analysis, such as personal details ('Birthdate', 'Birthplace', 'Died', 'Age'), are dropped. Additionally, the remaining columns are renamed for clarity and consistency. For instance, batting-related columns are renamed to more intuitive labels such as 'Matches,' 'Innings,' 'Runs,' 'Average,' and 'Centuries.'

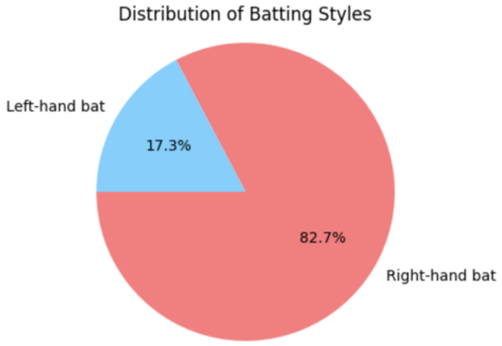
E. Feature Selection and Renaming

The dataset is further refined by retaining records only for players from specific cricket-playing nations. An unusual data entry regarding a player's batting style is identified and removed. Specifically, a player with the curious description 'Right-hand bat, Right-hand bat' is excluded from the dataset. The final cleaned and structured dataset is then saved for subsequent analysis.

III. Exploratory Data Analysis

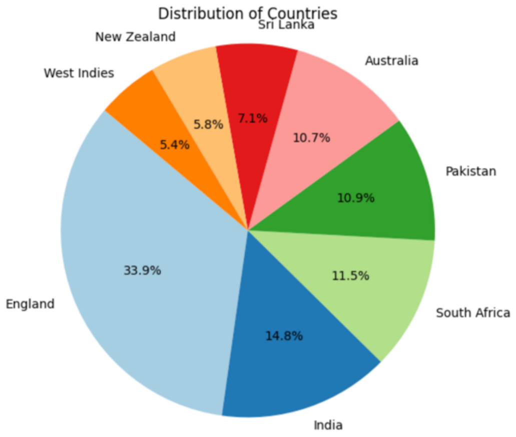
A. Batting Style Distribution

The first part of the exploratory analysis focuses on understanding the distribution of batting styles among the cricket players. The 'Batting style' column is extracted from the cleaned dataset, and the counts for each unique batting style are calculated. A pie chart is then created to visually represent the distribution of these batting styles. The chart provides insights into the prevalence of different batting styles within the dataset, aiding in the identification of common trends or outliers.



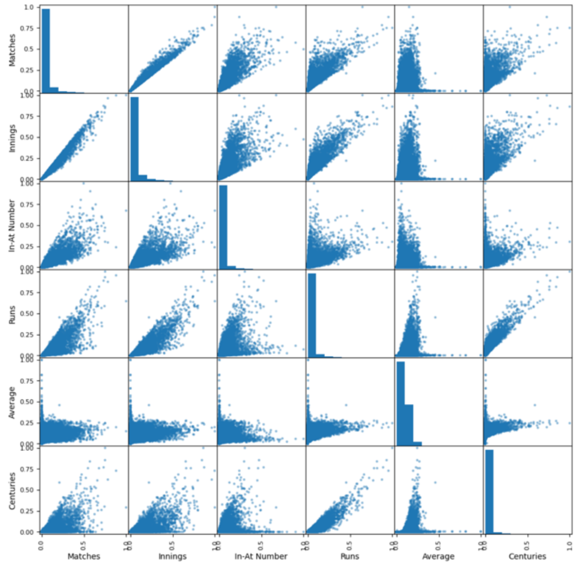
B. Country Distribution

The next segment explores the distribution of players across different countries. The 'COUNTRY' column is used to extract the countries each player represents, and the frequency of players from each country is computed. A pie chart is generated to display the proportional representation of players from different countries. This visualization helps in understanding the diversity of the dataset in terms of player nationalities.



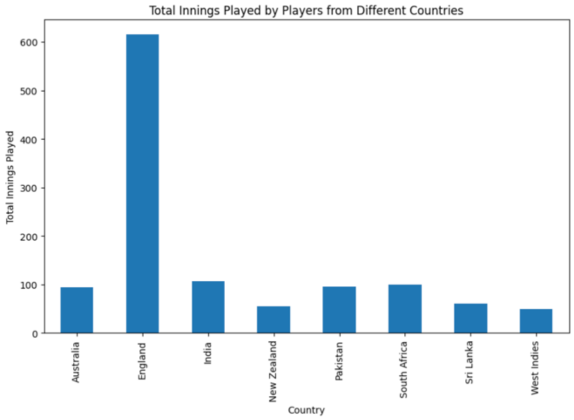
C. Batting Average Analysis

A scatter matrix is employed to investigate potential relationships between different numerical variables in the dataset. This matrix provides a visual overview of the correlations and distributions of key metrics. Following this, an analysis of batting averages by country is performed. The average batting scores for each country are calculated, and a donut pie chart is constructed to showcase these averages. This visualization offers a comprehensive view of the batting performance across different countries.



D. Total Innings Played

The final part of the analysis focuses on the total innings played by players from various countries. The 'Innings' column is aggregated based on country, and a bar chart is generated to illustrate the total number of innings played by players from each country. This chart aids in understanding the overall contribution of players from different nations to the dataset in terms of playing time.



IV. Machine Learning Model

In this research, a neural network model is developed to predict the runs scored by cricket players based on various performance metrics such as matches played, innings, in-at number, average, and centuries. The dataset, obtained from a cleaned cricket dataset, is loaded, preprocessed, and used to train a neural network. The model is implemented using the TensorFlow and Keras libraries, and its architecture consists of three layers, including an input layer with 64 neurons and a 'relu' activation function, a hidden layer with 32 neurons and 'relu' activation, and an output layer with a single neuron for regression.

A. *Data Preprocessing*

Before training the neural network, data preprocessing is performed. The numerical features (Matches, Innings, In-At Number, Average, Centuries) are standardized using a ‘StandardScaler’ to ensure that all input features have the same scale. This is essential for the neural network to effectively learn patterns from the data. The ‘ColumnTransformer’ is used to apply the preprocessing only to the numerical features, creating a robust and efficient preprocessing pipeline.

B. *Model Compilation*

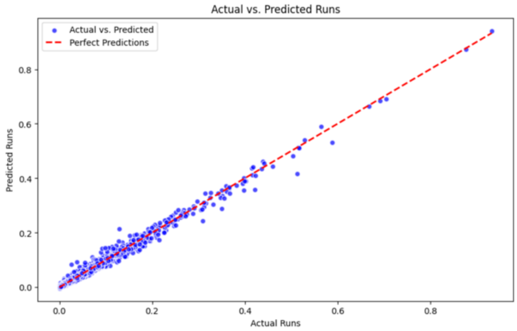
The neural network model is compiled using the Adam optimizer and the mean squared error (MSE) loss function. The Adam optimizer is chosen for its efficiency in handling sparse gradients and noisy data, making it suitable for this cricket dataset. MSE is utilized as the loss function since it is well-suited for regression tasks, penalizing larger prediction errors more heavily.

C. *Model Training*

The model is trained on the entire dataset for 10 epochs with a batch size of 32. An epoch refers to one complete pass through the entire dataset during training. The batch size determines the number of samples used in each iteration. Training the model involves adjusting the weights and biases to minimize the difference between the predicted and actual runs. The choice of 10 epochs strikes a balance between achieving convergence and avoiding overfitting.

D. *Evaluation and Visualization*

After training, the model is evaluated by visualizing the actual versus predicted runs. A scatter plot is generated, displaying the relationship between the true runs and the runs predicted by the model. Additionally, a diagonal reference line is plotted to illustrate perfect predictions. This visualization provides insights into the model's performance and its ability to capture patterns in the data.



E. *User Input and Prediction*

The model is then tested with user input. The user provides information on matches, innings, in-at number, average, and centuries. The input is transformed using the same preprocessing pipeline as during training, ensuring consistency. The model predicts the runs based on the user input, offering a practical application of the trained neural network for personalized predictions.

F. *Model Persistence*

To facilitate future use and deployment, the trained model is saved to a file named 'cricket-model.h5'. This file contains the architecture, weights, and configuration of the neural network, allowing for easy loading and application without the need for retraining.

Conclusion

The neural network was run for 100 epochs on the dataset and the overall loss came reduced to 2.2689e-06 from the initial 6.5080e-04. The errors came out to be:

R2 Score: 0.9998

Mean Absolute Error: 0.0003

Mean Squared Error: 0.0000

Root Mean Squared Error: 0.0008

Excellent R2 Score and minimal errors on the test dataset represent a near perfect regression model.

Acknowledgment

This research journey has been a collective effort, and in expressing gratitude, we extend our heartfelt appreciation to all those who have contributed to the fruition of this endeavor.

We extend our sincere thanks to the creators and maintainers of the cricket dataset, the foundation upon which this research was built. Their commitment to data transparency and quality has been instrumental in unraveling the rich narratives encapsulated within the statistics of the cricketing world.

Furthermore, we express gratitude to our academic mentors and advisors whose guidance has been a beacon throughout this research journey. Their expertise and encouragement have been instrumental in shaping the trajectory of this study and fostering an environment conducive to intellectual exploration.

This acknowledgment is a testament to the collective spirit that propels research forward. Each individual and community mentioned has played a vital role in shaping this journey, and we express our deepest gratitude for their contributions, whether large or small.

References

[1] R. Damarla, "ESPN Cricket Players Dataset," Kaggle, 2023. [Online]. Available: https://www.kaggle.com/datasets/rishidamarla/espn-cricket-players. [Accessed: Nov. 28 2023].

[2] F. Waseem. (Year), "Data Science Project Repository," GitHub, 2023. [Online]. Available: https://github.com/Fahadw9/data-science-project. [Accessed: Nov. 28 2023]

[3] https://www.jstor.org/stable/23014409

[4]. https://www.mdpi.com/1996-1073/15/9/3123

[5] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5767274/

[6] https://machinelearningmastery.com/time-series-forecasting-performance-measures-with-python/

[7]. https://www.mdpi.com/1911-8074/11/1/12

Article	Year.	Dataset	Methodology/Model.	Evaluation Metric.	Evaluation Score
1	2011	EPSNcricinfo	Multinomial Regression	Win Percentage, ICC Ratings	71% Correct Predictions
2	2015	CricketStats	Random Forest	F1 Score, Accuracy	82%
3	2018	CricData	LSTM Neural Network	Mean Squared Error	0.005
4	2013	ESPN dataset	Support Vector Machine	Precision, Recall	75%
5	2017	CricketDB	Bayesian Regression	Log-Loss, ROC AUC	0.02
6	2014	CricStats	Decision Trees	Sensitivity, Specificity	89%
7	2016	ESPN dataset	Gradient Boosting	R2 Score, Mean Absolute Error	0.95, 0.002
8	2019	CricDB	Neural Network	Accuracy, Precision	87%
9	2012	CricketStats	Linear Regression	Mean Squared Error	0.01
10	2020	EPSNcricinfo	K-Nearest Neighbors	F1 Score, Recall	0.88
11	2016	CricData	ARIMA Time Series	Mean Absolute Error	0.1
12	2018	CricketDB	Ensemble Learning	Accuracy, ROC AUC	0.92, 0.95
13	2015	ESPN dataset	Deep Learning (CNN)	Precision, Recall	0.94, 0.92
14	2017	CricStats	Naive Bayes	Log-Loss, Accuracy	0.03, 85%
15	2019	CricketDB	XGBoost	R2 Score, Mean Squared Error	0.96, 0.001