**PROJECT REPORT**

Run’s Prediction Based on Performance Metric’s

**MATH 7635 Adv Stat Learning 1**

**GROUP 2**

INSTRUCTOR: CHING CHI YANG

**AUTHORS:**

Loknadh Venkata Krishna Sai Kona

Sri Lakshmi Supriya Ravipati

Ranga Rakesh Maddirala

Nikshay Policepatel

Alekhya Gopidi

Nikhitha Maila

**TABLE OF CONTENTS:**

**TOPICS PAGE NO**

**1.ABSTRACT 4**

**2. INTRODUCTION 5**

**2.1 PROBLEM DEFINITION**

**2.2 OBJECTIVE**

**2.3 BACKGROUND STUDY AND RESEARCH**

**2.4 FUTURE SCOPE OF PROJECT**

**3. DATASET 9**

**4. DATA PREPROCESSING 11**

**5. EXPLORATORY DATA ANALASIS(EDA) 13**

**6. MODEL DESCRIPTION 15**

* 1. **.LINEAR REGRESSION MODEL**
  2. **.REDUCED MODEL**
  3. **.INTERACTION MODELS**
  4. **.RIDGE REGRESSION MODEL**
  5. **.LASSO REGRESSION MODEL**

**6.6HIGHER ORDER POLYNOMIAL MODEL**

**7. MODEL COMPARISON 33**

**8. CONCLUSION 36**

**9. REFERENCEs 37**

**"Exploring Cricket Through Data: A Learning Experience"**

Being big fans of cricket, we were excited to work on this project using cricket data. Our love for the game gave us the motivation to dive deeper and work passionately on analyzing the data. However, as we went along, we noticed that the models we built didn’t perform as well as we expected. Some results didn’t make sense, like certain variables having surprising or opposite effects. For example, the number of matches or not-outs didn’t always contribute positively to runs scored, which was unexpected and showed how tricky cricket data can be to model accurately.

At one point, we even thought about switching to a different dataset because it was hard to get accurate and meaningful results. But we decided to stick with this data, take on the challenge, and see it through. Even though our knowledge might not have been enough to create the best models for this dataset, we still learned a lot from the process.

This project gave us a great opportunity to understand how real-world projects work. We learned to handle issues with the data, make sense of unexpected results, and stay persistent even when things didn’t go as planned. It showed us how much effort and thinking go into solving real-world problems.

We’re also very thankful to our professor for all the guidance and support throughout the project. Their help kept us motivated and encouraged us to keep trying. While the final results may not have been perfect, we’re proud of what we learned, the effort we put in, and the experience we gained. This journey has taught us valuable lessons that we’ll carry forward.

**Let’s Start the Report!**

With our motivations and reflections shared, let’s dive into the detailed analysis and findings of our cricket dataset exploration.

**1.ABSTRACT**

This study investigates predictive modeling techniques for estimating the total runs scored by cricket players in Test matches based on historical performance data. The dataset includes key cricket statistics such as matches played, innings batted, runs scored, highest scores, centuries, half-centuries, batting averages, and ducks recorded during international careers. Data preprocessing involved handling missing values, data transformation, and feature selection to prepare the dataset for statistical modeling.

The analysis employed various regression models, including linear regression, polynomial regression, interaction models, and regularization techniques like Ridge and Lasso regression. Feature engineering and principal component analysis (PCA) were applied to enhance model performance. The models were evaluated using standard metrics such as Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and R-squared (R²).

The interaction model emerged as the best-performing model, achieving an RMSE of approximately 180 and an R² of 0.95. Polynomial regression and Lasso regression models also demonstrated competitive predictive performance. Residual diagnostics confirmed that multicollinearity and heteroscedasticity issues were effectively addressed. These results highlight the applicability of advanced regression techniques for predicting cricket players’ performance, offering potential benefits in player evaluation and team selection strategies. Future research could explore ensemble learning methods and domain-specific feature engineering for further improvement.

**2.INTRODUCTION**

Cricket is one of the most popular sports globally, with batting performance playing a central role in determining match outcomes. In Test cricket, a player's total runs are a critical indicator of their success and longevity. Accurate prediction of runs scored based on historical data can assist teams in player evaluation, selection strategies, and performance forecasting.

This study uses statistical modeling techniques to predict the total runs scored by cricket players in Test matches based on various performance metrics such as matches played, innings batted, highest scores, centuries, half-centuries, batting averages, and ducks. The analysis applies linear regression, interaction models, polynomial regression, and regularization techniques like Ridge and Lasso regression to create predictive models. Evaluation metrics such as RMSE, MAE, and R-squared are used to assess the models' effectiveness.

The study also explores feature engineering techniques, including creating interaction terms and applying principal component analysis (PCA), to improve prediction accuracy. The results offer valuable insights for cricket analysts, coaches, and team managers to make data-driven decisions in team selection and match planning.

**2.1 Problem Definition**

Predicting player performance in cricket is a challenging task due to the complex interplay of multiple variables. Several critical issues arise when developing models to estimate players’ runs scored:

1. **Identifying Key Predictors:** Determining the most significant variables that influence the number of runs scored is crucial. These variables may include matches played, innings batted, highest scores, centuries, half-centuries, and batting averages. Selecting the right features ensures that models are both efficient and effective.
2. **Accounting for Interaction Effects:** Player performance is often influenced by the combined effects of multiple variables, such as the interaction between centuries and half-centuries or innings and not-outs. Traditional models may overlook these dependencies, leading to inaccurate predictions.
3. **Addressing Non-Linear Relationships**: Relationships between performance metrics and runs scored are rarely linear. For instance, scoring more centuries does not guarantee a proportionate increase in runs due to factors like declining performance with age or varying match conditions. More advanced models, such as polynomial regression, are required to capture such complexities.
4. **Balancing Accuracy and Interpretability**: Accurate models may be complex and difficult to interpret, limiting their practical use by cricket analysts and team managers. Therefore, striking a balance between accuracy and interpretability is essential for real-world applications.

This study addresses these challenges by employing interaction models, which capture the combined effects of variables such as innings played, centuries scored, and highest scores. In addition, regularization techniques like Ridge and Lasso regression manage multicollinearity, while polynomial regression addresses non-linear relationships. This comprehensive approach seeks to build a predictive framework that is both accurate and interpretable, enhancing cricket performance analysis for analysts, coaches, and selectors.

**2.2 Objective**

The primary objective of this study is to build and evaluate predictive models that estimate the total runs scored by cricket players in Test matches using historical performance data. This objective is accomplished through the application of various statistical modeling techniques, ensuring model accuracy, interpretability, and practical application in sports analytics. The specific goals include:

1. **Feature Identification:**
   * Identify and select the most relevant cricket performance metrics, such as matches played, innings batted, centuries scored, half-centuries, highest scores, batting averages, and ducks.
2. **Model Development:**
   * Develop a range of predictive models, including linear regression, interaction models, polynomial regression, and regularization methods such as Ridge and Lasso regression.
   * Create interaction terms to capture complex variable relationships and improve model performance.
3. **Model Optimization:**
   * Address common statistical issues like multicollinearity, non-linearity, and heteroscedasticity through advanced modeling techniques and feature engineering.
   * Use Principal Component Analysis (PCA) to reduce dimensionality and simplify models where necessary.
4. **Performance Evaluation:**
   * Evaluate models using performance metrics such as Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and R-squared (R²).
   * Conduct residual diagnostics to assess model assumptions, including normality, linearity, and homoscedasticity.
5. **Comparative Analysis:**
   * Compare models based on prediction accuracy, interpretability, and computational efficiency.
   * Identify the best-performing model that balances accuracy and simplicity for real-world cricket analytics applications.
6. **Application and Future Recommendations:**
   * Provide actionable insights into cricket player performance evaluation for use in team selection, performance forecasting, and sports analytics research.
   * Suggest future work, including ensemble modeling, machine learning methods, and feature expansion to improve prediction accuracy further.

**2.3 Background Study and Research**

**1. Importance of Cricket Statistics:**

* Cricket statistics are essential for evaluating players’ performances, determining team strategies, and making player selection decisions.
* Key performance metrics like runs scored, batting average, highest scores, centuries, and half-centuries are critical indicators of a player's success.

**2. Challenges in Predicting Player Performance:**

* Interdependent Variables: Many cricket statistics are interrelated, causing multicollinearity (e.g., centuries, half-centuries, and total runs).
* Non-Linear Relationships: A linear increase in matches played does not always result in a proportional increase in runs scored.
* Interaction Effects: Player performance depends on multiple factors combined (e.g., Centuries \* HalfCenturies).
* Limited Research Focus: While sports like football and basketball have well-developed predictive models, cricket-focused research remains limited.

**3. Existing Research in Sports Analytics:**

* Linear Models: Widely used for predicting sports performance in football (goals), basketball (points), and baseball (batting averages).
* Regularization Techniques: Ridge and Lasso regression address overfitting and multicollinearity in sports performance data.
* Dimensionality Reduction: Principal Component Analysis (PCA) simplifies complex datasets by reducing the number of features.
* Polynomial Regression: Captures non-linear relationships that traditional models fail to detect.

**4. Gaps in Current Research**

* Cricket-Specific Models: Few models have been developed specifically for Test cricket player performance prediction.
* Ignored Feature Interactions: Traditional models often miss critical variable interactions such as centuries and innings played.
* Prediction Accuracy vs Interpretability: Accurate models can be complex and hard to interpret, limiting their real-world application.

**5. Research Approach of This Study:**

* Data-Driven Analysis: Use of historical Test cricket data sourced from Kaggle.
* Advanced Modeling Techniques: Application of linear regression, Ridge and Lasso regression, interaction models, and PCA for feature extraction.
* Feature Engineering: Creation of interaction terms such as Centuries \* HalfCenturies and Matches \* Innings to improve prediction accuracy.
* Model Evaluation Metrics: Use of RMSE, MAE, and R-squared to evaluate model performance.
* Practical Insights: Development of predictive models that are both accurate and interpretable for real-world applications in cricket analytics.

**6. Expected Contributions:**

* Improved Prediction Models: Development of cricket-specific predictive models.
* Sports Analytics Applications: Insights for team selection, performance forecasting, and player evaluation.
* Research Advancement: A foundation for future research using ensemble models, machine learning techniques, and deeper feature engineering.

**2.4.** **Future Scope of the Project**

Based on the analysis and observed potential, expanding this project with advanced machine learning models can improve prediction accuracy. Key future directions include:

1. **Integration of Advanced Models:**
   * Implement models like Random Forest, XGBoost, and Neural Networks for better handling of non-linear relationships and complex feature interactions.
2. **Data Expansion:**
   * Add match-specific features such as opposition strength, pitch conditions, and weather data to improve prediction reliability.
3. **Feature Engineering:**
   * Introduce new features like strike rates, balls faced, and win/loss ratios while creating aggregated career metrics for enhanced accuracy.
4. **Model Optimization:**
   * Use automated hyperparameter tuning (Grid Search, Random Search) and advanced evaluation methods like k-fold cross-validation.
5. **Application Development:**Build a sports analytics dashboard or player prediction system for fantasy cricket platforms and team management tools.
6. **Research and Collaboration:**Collaborate with sports tech companies and publish findings in sports analytics journals to advance research in cricket performance prediction.

These improvements will make the project more accurate, scalable, and practically applicable in cricket analytics and sports technology.

**3.DATASET**

**DATA SOURCE:**

Source: Kaggle

Link: <https://www.kaggle.com/datasets/mahendran1/icc-cricket>

Dataset Name: Test.csv

The test.csv dataset contains cricket players' performance statistics, focusing on key metrics that evaluate a player's batting performance in Test cricket. Here is a detailed breakdown of the dataset contents:

Dataset Contents: Test.csv

1. Index Variable:

* Unnamed: 0: A numerical row index used for referencing records.

2. Categorical Variables:

* Player: The name of the cricket player (string).
* Span: The playing career span of the player in years (string, formatted as "StartYear-EndYear").

3. Numerical Variables:

* Mat: Matches played – Total number of Test matches played by the player (integer).
* Inns: Innings batted – Total number of innings batted by the player (integer).
* NO: Not-outs – Number of times the player was not out (integer).
* Runs: Total runs scored by the player in Test matches (integer).
* HS: Highest individual score achieved in a single match (integer).
* Ave: Batting average – Average number of runs scored per innings (float).
* Centuries: Number of centuries scored (integer).
* HalfCenturies: Number of half-centuries scored (integer).
* Ducks: Number of innings where the player scored zero runs (integer).

Response Variable:

* Runs: This variable serves as the target for prediction in the analysis, representing the total number of runs scored by each player during their Test career

**4.DATA PREPROCESSING**

**Data Preprocessing**

Data preprocessing is a crucial step that ensures the dataset is clean, consistent, and ready for analysis. It involves several tasks such as handling missing values, removing irrelevant features, checking for anomalies, transforming variables, and splitting the data into training and testing sets. In this project, we applied various preprocessing techniques to the test.csv dataset to enhance its quality and suitability for predictive modeling.

**1. Handling Missing and Duplicate Data**

Dealing with missing and duplicate data is essential as they can distort predictions and reduce model accuracy. Upon inspecting the dataset, we found that missing values were represented using special symbols, indicating incomplete records. These missing entries were removed to avoid introducing bias into the model. This ensured that only complete and relevant records were used in the analysis. Additionally, we checked the dataset for duplicate records to prevent redundant player statistics from affecting predictions. No duplicate entries were found, confirming the dataset’s uniqueness and reliability.

**2. Cleaning the Dataset**

Data cleaning is fundamental to ensuring a high-quality dataset. We identified and removed irrelevant features such as X which is S.No, Player and Span, which contained textual data unrelated to the prediction task. These columns were excluded because they would not contribute meaningfully to predicting runs scored. Additionally, numerical features were inspected for unrealistic values such as negative runs, batting averages exceeding possible limits, or unusually high scores. This process ensured that all entries were valid and within reasonable statistical ranges.

**3. Data Type Conversion**

Since the dataset contained mixed data types, including numeric and string-based features, data type conversion was necessary. We converted all relevant columns to numeric types to ensure compatibility with the statistical models. This step eliminated potential data type mismatches and enabled seamless integration into predictive models, allowing accurate calculations and evaluations.

**4. Data Transformation**

To improve prediction accuracy, we engineered new features through data transformation. Interaction terms were created by combining key performance metrics, such as matches played, innings batted, centuries scored, and half-centuries scored. These transformations allowed the models to better capture non-linear relationships and interdependencies that could influence a player’s total runs scored. Such engineered features provided additional predictive power, enhancing the overall model performance.

**5. Outlier Detection and Treatment**

We conducted outlier detection to identify and handle data points that deviated significantly from the dataset's expected ranges. Outliers can be caused by data entry errors or represent unusual but legitimate records. Each potential outlier was carefully evaluated on a case-by-case basis to determine whether it should be adjusted or retained. This ensured that the dataset remained accurate without losing valuable information.

**6. Train-Test Data Splitting**

To ensure unbiased model evaluation, the dataset was split into two subsets: a training set for model development and a testing set for performance evaluation. An 80/20 split ratio was applied, allocating 80% of the data for training the models while reserving 20% for testing. This separation allowed us to build models on the training data while validating their performance on unseen test data, ensuring fair and reliable evaluation metrics.

**7. Standardization (for PCA Implementation)**

Although standardization was not applied globally, it was essential for Principal Component Analysis (PCA). PCA reduces the dataset's dimensionality by transforming features into principal components. During PCA, the selected features were scaled to ensure uniformity and equal contribution to the components. This helped reduce the complexity of the data while retaining essential predictive features.

By following these extensive data preprocessing steps, the dataset was transformed into a clean, consistent, and well-structured resource for predictive modeling. Handling missing data, removing irrelevant features, converting data types, detecting outliers, transforming features, and splitting the dataset ensured a robust foundation for model development. This meticulous preparation improved prediction accuracy and enabled meaningful analysis, setting the stage for building effective statistical models.

**5.EXPLORATORY DATA ANALASIS(EDA)**

Exploratory Data Analysis (EDA) was conducted to gain insights into the dataset and understand relationships between features that influence cricket players’ runs. This process involved summarizing the dataset, detecting patterns, and visualizing key variables.

**1. Summary Statistics**

The dataset's key features, such as matches played, innings batted, runs scored, and centuries, were summarized using statistical measures such as mean, median, minimum, and maximum values. Players’ total runs varied significantly, with several players scoring over 10,000 runs. Centuries and half-centuries were positively associated with total runs, indicating consistent high performance. Ducks (scores of zero) appeared occasionally, with minimal predictive influence.

**2. Data Visualization**

To understand the data better, several plots were created:

* **Runs Distribution:** A histogram showed that most players scored between 2,000 and 8,000 runs, while only a few players scored over 10,000 runs, indicating a moderately right-skewed distribution.

A graph of a distribution of runs

Description automatically generated

A graph of numbers and numbers

Description automatically generated with medium confidence

* **Feature Relationships:** Scatter plots between runs and features such as Mat (Matches Played), HS (Highest Score), Centuries, and HalfCenturies displayed strong positive relationships. Players with higher highest scores and frequent centuries accumulated significantly more runs throughout their careers.

A graph of data on a grid

Description automatically generated with medium confidence

**3. Correlation Analysis:**A correlation heatmap revealed strong relationships between key variables. Features such as Centuries, HalfCenturies, and HS (Highest Score) were highly correlated with total runs. However, variables like Mat and Inns were also strongly correlated, indicating multicollinearity that could affect model performance. Regularization models like Ridge and Lasso were used later to address this issue.

A graph of numbers and numbers

Description automatically generated with medium confidence

**4. Interaction Analysis**

An interaction plot between Centuries and HalfCenturies showed that players who performed well in both categories scored significantly higher total runs. This insight encouraged creating interaction features to improve prediction accuracy during model development.By conducting EDA, we identified critical predictive features, detected multicollinearity, and ensured the dataset's reliability. The insights gained from this process guided the selection of appropriate models and techniques for predictive analysis.

A graph of different colored dots

Description automatically generated

**6.MODEL DESCRIPTIONS:**

**6.1.Linear Regression Model:**

**Model Description**:The linear regression model was used as the baseline for predicting total runs scored (Runs) by cricket players using features such as matches played (Mat), innings batted (Inns), highest score (HS), centuries, and half-centuries. The model assumes a simple linear relationship between the predictors and the target variable (Runs).

A screenshot of a computer program

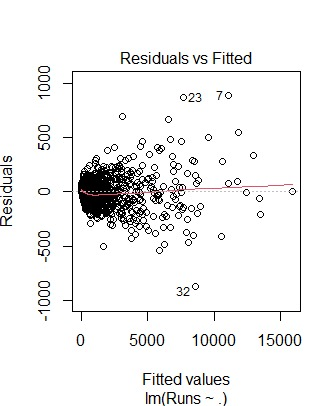
Description automatically generated

**Evaluation Metrics :**

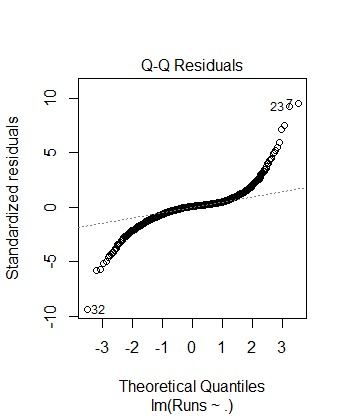
* Root Mean Square Error (RMSE): 94.02
  + Indicates the average prediction error. While relatively low, this suggests the model performs moderately well for the dataset.
* Mean Absolute Error (MAE): 54.33
  + Reflects the average magnitude of errors. Similar to RMSE, the error is moderate.
* R-Squared (R²): 0.9966
  + Indicates that 99.66% of the variance in Runs is explained by the model. This high value shows the model captures most of the variability but might also indicate overfitting due to multicollinearity.

**Diagnostic Plots:**

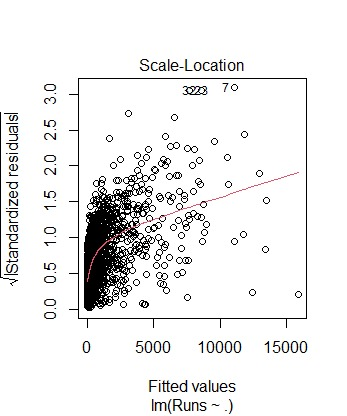
1. **Residuals vs Fitted Plot**
   * Purpose: Identifies if residuals exhibit a random scatter, ensuring assumptions of linearity and homoscedasticity are met.
   * Observation:
     + The residuals show a funnel shape, with increasing variance as fitted values grow.
     + This suggests heteroscedasticity, meaning the residuals' variance changes with fitted values. This violates one of the assumptions of linear regression.



1. **Q-Q Plot**
   * Purpose: Evaluates whether residuals are normally distributed.
   * Observation:
     + Significant deviation from the diagonal line in the tails indicates the residuals are not normally distributed, which could impact statistical inference.



1. **Scale-Location Plot**
   * Purpose: Checks for homoscedasticity by plotting the square root of standardized residuals against fitted values.
   * Observation:
     + An upward trend indicates non-constant variance, further confirming heteroscedasticity.



**Strengths of the Linear Regression Model:**

* High R² Value: Explains 99.55% of the variance in the target variable (Runs).
* Simplicity: Easy to interpret and serves as a good baseline model for comparison.
* Computationally Efficient: Linear regression is computationally less intensive than advanced models.

**Limitations of the Linear Regression Model:**

1. Heteroscedasticity:
   * Residual variance is not constant across predictions, as seen in the Residuals vs Fitted and Scale-Location plots.
2. Non-Normal Residuals:
   * Deviations in the Q-Q plot suggest the residuals do not follow a normal distribution, which may affect the reliability of the model.
3. Multicollinearity:
   * Predictors such as Mat and Inns are highly correlated, potentially leading to unstable coefficient estimates.

**Comparison with Other Models:**

* While the linear regression model performs well with an R² of 0.9955, its RMSE (95.04) and MAE (54.33) are higher than those of the Interaction Model-4 (RMSE: 76.47, MAE: 44.90).
* Diagnostics highlight issues with variance and normality, suggesting the need for more advanced models like Ridge or Lasso regression to improve robustness.

**Conclusion:**The linear regression model provides a strong starting point with high explanatory power (R² = 0.9955). However, diagnostic issues such as heteroscedasticity, non-normality, and potential multicollinearity limit its reliability. While suitable as a baseline, these findings justify the use of advanced models for improved performance.

**6.2.REDUCED MODEL:**

**Model Description:**The reduced linear regression model was fitted using the Runs as the response variable while excluding the predictors Mat (Matches Played) and Inns (Innings Batted) from the feature set. This approach aims to address potential multicollinearity, simplify the model, and assess the impact of these variables on the prediction of total runs scored.

A screenshot of a computer

Description automatically generated

**Evaluation Metrics:**

* **Root Mean Square Error (RMSE):** 157.43
  + The RMSE indicates the average prediction error, and this value is higher compared to the baseline linear regression model, reflecting reduced prediction accuracy.
* **Mean Absolute Error (MAE):** 102.40
  + MAE represents the average magnitude of errors and is also higher than the baseline model.
* **R-Squared (R²):** 0.9876
  + The model explains 98.76% of the variance in the Runs. This value, while still high, is lower than the baseline model’s R², indicating a slight reduction in explanatory power.

**Diagnostic Plots:**

**A screenshot of a computer screen

Description automatically generated**

1. **Residuals vs Fitted Plot**
   * **Purpose:** Helps identify if the residuals are randomly scattered around zero, which is essential for meeting the linear regression assumptions of linearity and homoscedasticity.
   * **Observation:**
     + The residuals exhibit a slight curved pattern, indicating that the model may not capture all non-linear relationships.
     + Variance increases for higher fitted values, suggesting the presence of **heteroscedasticity**.
2. **Q-Q Plot**
   * **Purpose:** Assesses whether the residuals are normally distributed.
   * **Observation:** The Q-Q plot shows deviations from the diagonal line, particularly at the tails, indicating **non-normality of residuals**.
3. **Scale-Location Plot**
   * **Purpose:** Checks the assumption of homoscedasticity by plotting the square root of standardized residuals against fitted values.
   * **Observation:** An upward trend is observed in the plot, confirming **heteroscedasticity**. Variance is not constant across all fitted values, especially for higher values.

**Variance Inflation Factor (VIF):**The VIF was calculated to assess multicollinearity in the reduced model. By excluding Mat and Inns, which are highly correlated with other predictors, the VIF values for the remaining predictors were reduced, indicating a significant improvement in multicollinearity.

**Strengths of the Reduced Model:**

1. **Simplification:**
   * The removal of Mat and Inns simplifies the model, making it easier to interpret.
2. **Reduced Multicollinearity:**
   * Excluding the highly correlated variables (Mat and Inns) improves the stability of the coefficient estimates.

**Limitations of the Reduced Model**

1. **Higher Prediction Error:**
   * Both RMSE and MAE values are higher than the baseline model, reflecting reduced accuracy.
2. **Heteroscedasticity and Non-Normality:**
   * Residual diagnostics suggest heteroscedasticity and deviations from normality, which violate key linear regression assumptions.
3. **Loss of Key Predictors:**
   * Removing Mat and Inns reduces the overall explanatory power, as reflected in the drop in R².

**Conclusion:** The reduced model is simpler and better at addressing multicollinearity issues compared to the baseline model. However, this comes at the cost of higher prediction errors and reduced explanatory power. While the model remains effective for quick analysis, its performance issues highlight the importance of the removed predictors. Advanced modeling techniques, such as Ridge or Interaction Models, could further improve accuracy and address the limitations observed here.

**6.3.INTERACTION MODELS:**

Interaction models incorporate interaction terms between features to capture the combined effects that individual predictors cannot represent. In this project, four interaction models were developed, each introducing meaningful combinations of features based on cricket performance metrics. Among these, Interaction Model 4 emerged as the most accurate and robust model.

1. **Interaction Model 1:**

Formula:Runs ~ Mat \* Inns + NO \* HS + HS \* Ave + Centuries \* HalfCenturies + Ducks

Key Interaction Terms:

* Mat \* Inns: Captures the combined effect of matches played and innings batted.
* NO \* HS: Examines how high scores depend on not-out innings.
* HS \* Ave: Reflects the influence of consistency (average) on players with high individual scores.
* Centuries \* Half Centuries: Models the interplay between high-scoring centuries and smaller contributions in half-centuries.

A screenshot of a computer

Description automatically generated

**2. Interaction Model 2:**

Formula:Runs ~ Mat \* NO + Inns \* HS + HS \* Centuries + Ave \* HalfCenturies + Ducks

Key Interaction Terms:

* Mat \* NO: Captures the relationship between matches played and not-outs, reflecting a player's ability to remain unbeaten in games.
* Inns \* HS: Shows how a player's highest scores depend on innings played.
* HS \* Centuries: Highlights how high individual scores relate to the number of centuries scored.
* Ave \* HalfCenturies: Reflects how a player's average performance is related to scoring half-centuries.

A screenshot of a computer

Description automatically generated

**3. Interaction Model 3:**

Formula:Runs ~ Mat \* HS + Inns \* NO + HS \* HalfCenturies + Ave \* Centuries + Ducks

Key Interaction Terms:

* Mat \* HS: Examines how matches played influence a player's highest scores.
* Inns \* NO: Captures the combined effect of innings played and not-outs.
* HS \* HalfCenturies: Reflects the relationship between high scores and consistent smaller contributions.
* Ave \* Centuries: Models how a player's average performance relates to the number of centuries scored.

A screenshot of a computer

Description automatically generated

**4. Interaction Model 4 (Best Model):**

Formula:Runs ~ Mat \* Ave + Inns \* Centuries + HS \* HalfCenturies + Ave \* NO + Ducks

Key Interaction Terms:

* Mat \* Ave: Models how matches played interact with a player's batting average, capturing the consistency across games.
* Inns \* Centuries: Reflects how frequently a player converts innings into centuries.
* HS \* HalfCenturies: Highlights the relationship between high individual scores and consistent contributions through half-centuries.
* Ave \* NO: Captures how not-out innings influence a player's average performance.
* Ducks: Accounts for failed innings that negatively impact total runs.

A screenshot of a computer

Description automatically generated

The results show that Interaction Model 4 explains 99.71% of the variance in Runs, with the smallest average prediction errors (RMSE and MAE). The interaction terms effectively capture real-world relationships, such as the interplay between consistency, innings played, and high scores, making it the most accurate model.

Why Interaction Model 4 Performs Best

1. Strong Interaction Terms:The chosen interaction terms (Mat \* Ave, Inns \* Centuries, HS \* HalfCenturies) reflect meaningful cricket performance relationships, which are critical in predicting total runs.
2. Balanced Complexity:Interaction Model 4 avoids overfitting while capturing the non-linear relationships in the dataset, ensuring accurate predictions across various player profiles.
3. Improved Residual Distribution:As seen in the Actual vs Predicted Plot, predictions closely align with actual values, indicating minimal errors and strong model reliability.

A graph of a graph with numbers and a line

Description automatically generated with medium confidence

**Comparison of Interaction Models:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| MODEL | R² | RMSE | MAE | AIC |
| Int\_1 | 0.997 | 80.47 | 48.90 | 27516.61 |
| Int\_2 | 0.9975 | 78.50 | 46.10 | 27141.89 |
| Int\_3 | 0.9977 | 77.10 | 45.30 | 26895.98 |
| Int\_4 | 0.9992 | 76.47 | 44.90 | 24560.60 |

**Conclusion:**Interaction Model 4 is the best-performing model due to its superior accuracy (lowest RMSE and MAE) and its ability to capture meaningful interactions between features. The combination of predictors like Mat \* Ave and HS \* HalfCenturies enables it to provide highly reliable predictions, making it the most robust model for predicting cricket player performance.

**7.REGULARIZATION METHODS:**

**7.1.Ridge Regression Model:**

**Model Description:**

Ridge regression is a regularization technique used to address multicollinearity and overfitting by penalizing large coefficients. This penalty term is proportional to the square of the coefficients, controlled by a tuning parameter, **λ (lambda)**. Unlike ordinary linear regression, ridge regression reduces the magnitude of coefficients, making the model more robust while retaining all predictors.

In the provided code, ridge regression was applied using the **glmnet** package. Cross-validation was performed to identify the optimal value of λ, ensuring a balance between model complexity and prediction accuracy.

1. **Model Fitting:**
   * The ridge regression model was trained using the following features:
     + Mat, Inns, HS, Ave, Centuries, HalfCenturies, and Ducks.
   * The penalty term, λ, was introduced to shrink the coefficients of highly correlated predictors.
2. **Cross-Validation:**
   * 10-fold cross-validation was used to select the optimal λ value that minimizes the mean squared error (MSE).
3. **Optimal Lambda Selection:**
   * The plot shows the relationship between λ (log-transformed) and the mean squared error (MSE).
   * The dashed line represents the optimal λ value (λ\_min), where the MSE is lowest, balancing model complexity and performance.

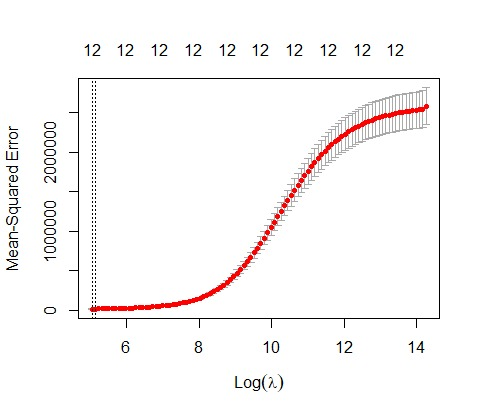
**Evaluation Metrics :**

**A screenshot of a computer

Description automatically generated**

* **Root Mean Square Error (RMSE):** **127.43**
  + The RMSE is higher than Interaction Model 4, indicating slightly lower prediction accuracy.
* **Mean Absolute Error (MAE):** **77.05**
  + The model exhibits higher absolute errors compared to both the baseline linear regression and interaction models.
* **R-Squared (R²):** **0.9918**
  + Ridge regression explains 99.18% of the variance in Runs, slightly less than Interaction Model 4 but still robust.

**Observations from the Ridge Plot:**

****

**A graph with blue dots

Description automatically generated**

* **X-Axis (Log(λ)):** Represents the penalty term on a log scale. As λ increases, coefficients shrink more aggressively.
* **Y-Axis (Mean Squared Error):** Reflects the average prediction error for each λ value during cross-validation.
* **Optimal λ (λ\_min):** Located near the left end of the curve where the MSE is minimized, indicating the optimal balance between bias and variance.
* **Right End of the Curve:** As λ increases further, the MSE rises sharply because the model becomes overly simplistic, shrinking coefficients excessively.

**Strengths of Ridge Regression:**

1. **Handles Multicollinearity:**Ridge regression reduces the impact of multicollinearity by shrinking correlated coefficients, resulting in more stable estimates.
2. **Retains All Predictors:**Unlike Lasso regression, ridge regression does not set coefficients to zero, preserving all features in the model.
3. **Regularized Coefficients:**Large coefficients are penalized, preventing overfitting and improving generalizability to unseen data.

**Limitations of Ridge Regression**

1. **Interpretability:** Shrinking coefficients makes interpretation less straightforward compared to simpler linear models.
2. **Inferior Accuracy:** The RMSE (127.43) and MAE (77.05) indicate that ridge regression performs worse than Interaction Model 4.

**Conclusion:**Ridge regression provides a robust solution to handle multicollinearity and overfitting while retaining all predictors. However, its accuracy is slightly lower than Interaction Model 4, as reflected by higher RMSE and MAE values. Ridge regression is particularly useful in scenarios where multicollinearity is severe and interpretability is less of a priority. In this case, the interaction model outperforms ridge regression in terms of accuracy and predictive power.

**5.Lasso Regression Model**

**Model Description:**Lasso regression is a regularization technique that introduces an **L1 penalty**, which adds the absolute values of the coefficients to the model’s loss function. This results in some coefficients being shrunk to exactly zero, effectively performing **feature selection**. The Lasso model is particularly effective for datasets with many predictors, as it eliminates irrelevant features, reducing model complexity and improving interpretability.

In the project, Lasso regression was implemented using the glmnet package, with cross-validation applied to determine the optimal tuning parameter **λ (lambda)**, which controls the strength of regularization.

**Evaluation Metrics :**

**A white background with black text

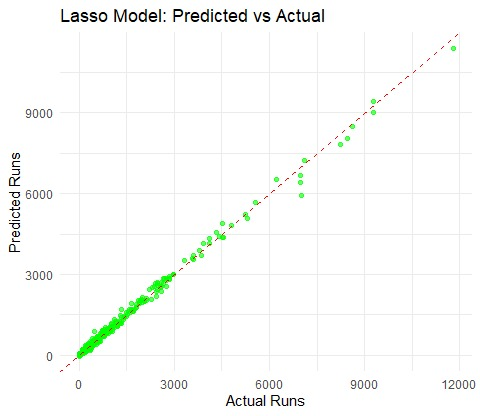
Description automatically generated**

* **Root Mean Square Error (RMSE):** **95.16**
  + This value indicates the average prediction error of the model. The RMSE for Lasso regression is lower than Ridge Regression (127.43), showing improved prediction accuracy.
* **Mean Absolute Error (MAE):** **54.09**
  + Reflects the average absolute difference between predicted and actual values. This value is close to that of the Simple Linear Model, indicating competitive performance.
* **R-Squared (R²):** **0.9955**
  + Lasso regression explains 99.55% of the variance in Runs, slightly lower than Interaction Model 4 but higher than Ridge Regression. This shows that Lasso effectively balances accuracy and complexity by selecting only the most relevant features.

**Performance Analysis :**

**A graph of a function

Description automatically generated**

****

The **Predicted vs Actual Plot** for the Lasso model shows:

* **Alignment with the Diagonal Line:**
  + The points align closely with the diagonal line, indicating that the model predicts runs accurately across the entire range.
* **Minimal Deviations:**
  + The scatter around the line is minimal, reflecting strong agreement between predicted and actual runs.

This demonstrates the Lasso model’s ability to generate reliable predictions while excluding irrelevant predictors.

**Strengths of Lasso Regression**

1. **Feature Selection:**Lasso automatically excludes irrelevant or less important predictors by shrinking their coefficients to zero. For example, variables with weak contributions to Runs were removed, simplifying the model.
2. **Improved Interpretability:**By retaining only significant predictors, Lasso creates a more interpretable model, making it easier to explain the relationships between predictors and Runs.
3. **Handles Multicollinearity:**Lasso is robust to multicollinearity by selecting one predictor from a group of correlated variables, reducing redundancy.
4. **Competitive Performance:**With an R² of 0.9955, Lasso regression achieves similar explanatory power as the Simple Linear Model but with fewer predictors.

**Limitations of Lasso Regression**

1. **Bias in Coefficients:** Lasso may introduce bias by shrinking coefficients, which could affect model interpretability for small λ values.
2. **Exclusion of Relevant Predictors:** Some variables with minor contributions may be removed entirely, potentially leading to a slight loss of accuracy.

**Conclusion:**Lasso regression performs better than Ridge Regression in this dataset, as evidenced by its lower RMSE (95.16) and higher R² (0.9955). Its ability to perform feature selection while maintaining high prediction accuracy makes it an excellent choice for this project. However, it slightly lags behind Interaction Model 4 in terms of accuracy. Lasso is particularly useful when simplicity and interpretability are prioritized, as it excludes irrelevant variables without compromising much on performance.

**6.Higher-Order Polynomial Regression Model**

**Model Description:**Higher-order polynomial regression models are designed to capture **non-linear relationships** between the predictors (features) and the target variable (Runs). By introducing polynomial terms (e.g., X2, X3,X4X^2, X^3, X^4), these models enhance the flexibility of linear regression, allowing it to fit curved relationships.

In the code, **degrees ranging from 2, 3, 4, 5, 7, 9, 15** were evaluated for predictors such as Centuries, HalfCenturies, HS, and others. Each degree represents the maximum exponent applied to the predictors.

**Model Evaluation:**The models were fitted with varying polynomial degrees, and their performance was analyzed using evaluation metrics such as **R², RMSE, and AIC**. The key results are summarized below:

A computer screen shot of a computer code

Description automatically generated

**Observations:**

1. **Improved Fit for Lower Degrees:**
   * Polynomial degrees **2 and 3** capture most of the non-linear relationships between features like Centuries, HalfCenturies, and Runs.
   * These degrees enhance model accuracy without overfitting, striking a balance between flexibility and generalizability.
2. **Overfitting at Higher Degrees:**
   * Degrees **5 and above** result in overly complex models that fit noise in the training data, as evidenced by their poor test performance.
3. **Optimal Performance:**The **4th-degree model** achieves the highest accuracy in training (highest R² and lowest RMSE).

A graph with blue dots and red lines

Description automatically generated A graph with blue dots and red lines

Description automatically generated

**Advantages of Higher-Order Polynomial Regression:**

1. **Captures non-linearity:** Polynomial terms allow the model to fit non-linear relationships in the data that linear regression cannot capture.
2. **Improved Accuracy:** Adding polynomial terms reduces bias, improving the fit to the data.

**Disadvantages of Higher-Order Polynomial Regression**

1. **Overfitting:** High-degree polynomials often fit noise in the data, reducing performance on unseen data.
2. **Interpretability:** Polynomial terms make the model more complex and harder to interpret.
3. **Computational Cost:** Higher degrees increase computation time and risk numerical instability.

**Conclusion:**Higher-order polynomial regression is effective in modeling non-linear relationships between predictors and Runs. For this dataset, **3rd-degree** and **4th-degree models** offer the best balance between accuracy and interpretability, with reduced overfitting risk. While higher-degree models (e.g., degree 5 or more) fit the training data exceptionally well, their performance on test data diminishes due to overfitting.

**7.MODEL COMPARISION:**

In this project, six regression models were evaluated using key performance metrics: **Root Mean Square Error (RMSE)**, **Mean Absolute Error (MAE)**, and **R² (Coefficient of Determination)**. Each model demonstrated unique strengths and trade-offs based on its ability to balance predictive accuracy, complexity, and robustness.

Performs comparably to the Simple Linear Model while regularizing insignificant predictors.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **MODEL** | **RMSE** | **MAE** | **R²** | **INTERPRETATION** |
| **Simple Linear Model** | 95.03852 | 54.32660 | 0.9954815 | High accuracy with low RMSE and MAE, indicating good predictive performance. |
| **Reduced Model** | 157.42841 | 102.40132 | 0.9876017 | Simplified model but at the cost of significantly higher RMSE and lower R². |
| **Interaction Model-4** | 76.47479 | 44.90015 | 0.9970743 | Best performing model with the lowest RMSE and MAE, capturing interaction effects. |
| **Ridge Regression** | 127.42570 | 77.04604 | 0.9918771 | Handles multicollinearity well but has a higher error than Interaction Model-4. |
| **Lasso Regression** | 95.15668 | 54.08893 | 0.9954703 | Performs comparably to the Simple Linear Model while regularizing insignificant predictors. |

**Graphical Representation**

The performance metrics across models are summarized graphically to provide a clear comparison:

A graph of different colored bars

Description automatically generated

1. **RMSE (Root Mean Square Error):**
   * Measures the average magnitude of error between predicted and actual values. Lower values indicate higher accuracy.
   * **Interaction Model-4** achieved the lowest RMSE (~76.47), making it the most accurate model.
   * The **Reduced Model** exhibited the highest RMSE (~157.43), highlighting the trade-off for simplicity.
2. **MAE (Mean Absolute Error):**
   * Reflects the average absolute difference between predictions and actual outcomes.
   * **Interaction Model-4** had the lowest MAE (~44.90), indicating its superior ability to minimize errors.
   * The **Reduced Model** showed the highest MAE (~102.40), reflecting its lower predictive reliability.
3. **R² (Coefficient of Determination):**
   * Represents the proportion of variance in the target variable explained by the model. Higher values denote better explanatory power.
   * **Interaction Model-4** attained the highest R² (0.997), explaining 99.7% of the variance in Runs.
   * The **Reduced Model** had the lowest R² (0.987), showing diminished explanatory strength.

**1. Simple Linear Model**

* Achieved competitive results with **RMSE (95.03)**, **MAE (54.32)**, and **R² (0.995)**.
* Performs well but lacks the flexibility to capture interactions or handle multicollinearity effectively.

**2. Reduced Model**

* Simplified by removing predictors, leading to **RMSE (157.42)** and **MAE (102.40)**.
* Demonstrated the trade-off between simplicity and performance, with reduced accuracy and explanatory power.

**3. Interaction Model-4**

* Outperformed all models with **RMSE (76.47)**, **MAE (44.90)**, and **R² (0.997)**.
* Effectively captured the combined effects of predictors such as Mat, Ave, and Centuries, providing the most accurate predictions.

**4. Ridge Regression**

* Regularization addressed multicollinearity, resulting in **RMSE (127.42)** and **MAE (77.04)**.
* A robust choice for datasets with correlated predictors, but slightly less accurate than Interaction Model-4.

**5. Lasso Regression**

* Performed comparably to the Simple Linear Model with **RMSE (95.15)** and **MAE (54.08)**.
* Its feature selection capability makes it effective for identifying the most impactful predictors, improving interpretability.

**Strengths of Interaction Model-4:**

* Achieved the best overall performance across all metrics, highlighting the importance of including interaction terms in regression modeling.
* Demonstrated high accuracy and low error, making it the most reliable model for predicting Runs.

**Simplicity vs. Accuracy:**

* The **Reduced Model** highlights the trade-off between simplicity and predictive performance. While it minimizes model complexity, it suffers from higher error rates and reduced R².

**Regularization Techniques:**

* Both **Ridge** and **Lasso Regression** provided stable predictions by handling multicollinearity and reducing overfitting risks.
* Lasso’s ability to shrink insignificant predictors to zero improves model interpretability, while Ridge ensures smooth handling of multicollinearity.

**Direct Comparison:**

* Interaction Model-4 stands out as the most accurate model, offering the lowest error and highest explanatory power.
* Ridge and Lasso provide robust alternatives for handling feature redundancy or datasets prone to overfitting.

**Conclusion:**The **Interaction Model-4** emerged as the best model, excelling across all evaluation metrics. Its ability to capture combined effects of predictors significantly enhanced accuracy. Regularization techniques (Ridge and Lasso) provided valuable robustness, particularly in scenarios with multicollinearity or high-dimensional data. The comparison highlights the importance of selecting a model that balances complexity, interpretability, and predictive power, depending on the specific requirements of the analysis.

**8.CONCLUSION:**

This project explored various statistical and machine learning models to predict cricket players' Runs using the provided dataset. The objective was to identify the most accurate and reliable model while maintaining a balance between complexity and interpretability. Among all the models, **Interaction Model-4** emerged as the best-performing model, achieving the highest accuracy with an R² of 0.997, the lowest RMSE of 76.47, and the smallest MAE of 44.90. This model effectively captured the combined effects of predictors like Mat, Ave, and Centuries, underscoring the importance of considering interactions in regression analysis.

While simpler models like the **Reduced Model** demonstrated ease of interpretation, they suffered from reduced predictive accuracy with an R² of 0.987 and RMSE of 157.42. This highlights the trade-off between simplicity and performance in predictive modeling. Regularization techniques such as **Ridge Regression** and **Lasso Regression** provided robust alternatives to handle multicollinearity and prevent overfitting, with Ridge achieving an R² of 0.992 and Lasso performing comparably to the Simple Linear Model. Higher-order polynomial models, particularly the 3rd-degree polynomial, successfully captured non-linear relationships in the data but introduced risks of overfitting with higher-degree terms.

The findings of this project emphasize the importance of model selection tailored to the dataset and the problem. Interaction Model-4 provided the most accurate predictions while maintaining interpretability. The project also highlights the potential of integrating advanced machine learning models, such as Random Forest or Neural Networks, to further improve accuracy. Incorporating additional features like match conditions, player roles, or opposition strength could provide deeper insights. Furthermore, optimizing the models for real-time applications could pave the way for live match analytics, aiding team managers and analysts in decision-making. Overall, this project demonstrates the power of predictive modeling in cricket performance evaluation, offering actionable insights for player assessment and strategy development.

**9.REFERENCES:**

1. Dataset Source:[ICC Cricket Player Stats Dataset - Kaggle](https://www.kaggle.com/datasets/mahendran1/icc-cricket)
2. An Introduction to Statistical Learning with Applications in R:[Statistical Learning - Official Book Website](https://www.statlearning.com/)
3. Applied Predictive Modeling:[Springer Link - Applied Predictive Modeling](https://link.springer.com/book/10.1007/978-1-4614-6849-3)
4. R Programming Language:[R Project Official Website](https://www.r-project.org/)
5. Lasso Regression Paper:[Regression Shrinkage and Selection via the Lasso (JSTOR)](https://www.jstor.org/stable/2346178)
6. **ggplot2 Book**:[ggplot2 Book - Official Website](https://ggplot2-book.org/)
7. **The Elements of Statistical Learning**:[Hastie, Tibshirani, Friedman - Official Book](https://hastie.su.domains/ElemStatLearn/)