Analysis of Soccer Players’ Score

Tianfeng Liao
12212661

Haohan Cui
12210503

Yan Shao
12212004

***Abstract*—This study explores soccer player performance using FIFA datasets to understand how the integrated ability is evaluated and predict their future score. Data preprocessing and visualization are conducted to ensure quality and identify key patterns. Cluster analysis, utilizing PCA and K-means, groups players based on similar characteristics. Regression analysis focuses on variable selection, assumption testing, and building a final predictive model while Bootstrap is used for estimation. Classification techniques, including decision trees, conditional inference trees, and random forests, are employed to determine the most effective predictive approach. The study offers a comprehensive evaluation framework for player performance and potential applications in team optimization.**

I. About Data

Originally here are two comprehensive datasets, FIFA and FIFA21, which contain performance metrics and attributes of male football players participating in any leagues. These datasets provide valuable information for analyzing and predicting player abilities.

- The first dataset comprises 3,961 records spanning the 2016–2020 seasons. Each record includes detailed player attributes such as basic information (e.g., name, nationality, age, height, weight), club position, positional data, technical skills (e.g., crossing, dribbling, passing accuracy), physical characteristics (e.g., speed, agility, stamina, strength), and defensive and offensive capabilities (e.g., interceptions, positioning, finishing). Notably, this dataset includes a composite ability score corresponding to the player’s performance in the subsequent season (2017–2021), enabling longitudinal performance tracking and predictive analysis.
- The second dataset features 1,079 records of active male players from the 2021 any league season. Similar to FIFA.xlsx, it contains a rich set of variables capturing players’ characters. While this dataset does not include a future ability score, it serves as a complementary snapshot of contemporary player performance for cross-season comparisons and benchmarking.

A. Data Checking and Revision

In the process of data analysis, the quality and integrity of the data affect the reliability and accuracy of the analysis of the analysis results. Therefore, we need to first clean the data and take appropriate measures to organize it: using R language to check for missing values and fill in missing values functions to conduct a preliminary organization of the original data. Finally, we found that there were no missing values in the all datasets, but the data from 2021 lacked a overall ability indicator compared to the previous years, which suggests our research goal!

In order to facilitate the extraction of the dataset, we attempted to fill in the missing values of comprehensive ability in the 2021 dataset by assigning NA values, and vertically merged the two datasets together to classify players based on their age.

B. Data Visualization

Before the formal research begins, we should conduct a macro analysis of the overall datasets and visualize the data to make it more understandable.

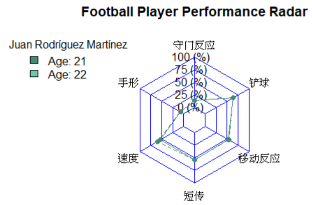


Fig. 1: Radar Chart

As a digital football coach, understanding how player abilities evolve over time is essential for both performance analysis and long-term team strategies. To achieve this, we devised a structured approach to analyze the relationship between player age and their observed abilities. The figure above presents a comparison of player age with individual performance indicators, achieved by selecting appropriate metrics to assess player ability. The following figure displays a line chart illustrating the relationship between comprehensive ability and age, which provides a clearer view of the data and facilitates a more straightforward analysis of player performance trends over time.

During the research, we explored whether over ability might be connected with other factors such us weight,height and so on. To test this hypothesis, we used two bubble charts for the more intuitive analysis. The results revealed a strong positive connection between international reputation and comprehensive ability, as shown in the left figure. Combining the insights from both the left and right figures, it became apparent that there is no significant linear correlation between height, weight, age, and overall ability. Additionally, using the F-test, we found that the P-value was greater than 0.05, further

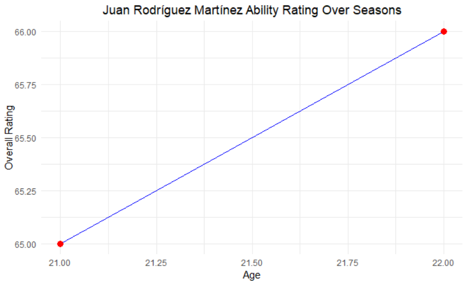


Fig. 2: Line Chart

supporting the conclusion that the null hypothesis cannot be rejected.

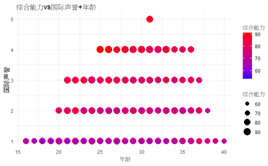


Fig. 3: Overall vs F&A

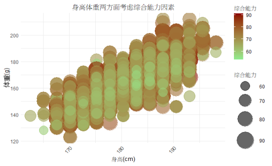


Fig. 4: Overall vs H&W

Next, we compared the differences in abilities of players in the same position and verified the hypothesis through radar charts and T-tests. The results showed that except for 'reverse foot ability', there was no significant difference in other abilities, indicating that the differences in abilities among players in different positions were small that can be ignore!

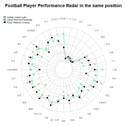
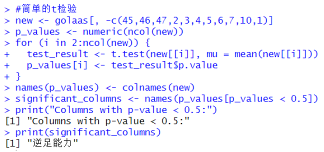


Fig. 5: Radar Chart



```
> #简单的t检验
> new <- golaas[, -c(45,46,47,2,3,4,5,6,7,10,1)]
> p_values <- numeric(ncol(new))
> for (i in 2:ncol(new)) {
+   test_result <- t.test(new[[i]], mu = mean(new[[i]]))
+   p_values[i] <- test_result$p.value
+ }
> names(p_values) <- colnames(new)
> significant_columns <- names(p_values[p_values < 0.5])
> print("Columns with p-value < 0.5:")
[1] "Columns with p-value < 0.5:"
> print(significant_columns)
[1] "逆足能力"
```

Fig. 6: T-test

Finally, we analyzed the ability of top players in different positions and compared them by advanced radar charts, visually presenting the differences in performance of players in various abilities in different positions.

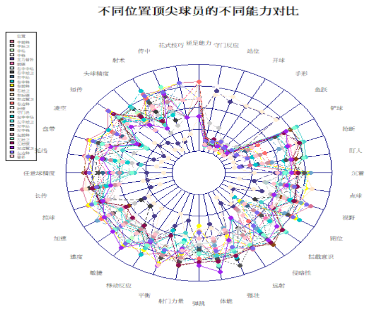


Fig. 7: Comparison Radar Chart

II. Clutser Algorithm

A. Data Pre-processing

During the visualization process, we discovered that the originally integrated dataset was not suitable for our subsequent analysis, and the merging of variables did not yield and meaningful results. Therefore, we decided to remove the international reputation indicator and combine height and weight into a single indicator: BMI. What's more, we created a new dataset with only two columns—international reputation and comprehensive ability—specifically for validation purposes.

B. Alghrithm Selection

1) PCA: In our study, in order to extract key factors from complex data and effectively reduce dimensionality, we first attempted PCA principal component analysis.

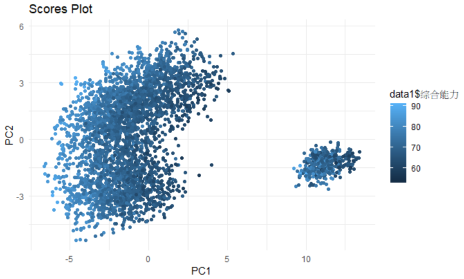


Fig. 8: PCA Score Plot

In the score graph after dimensionality reduction, two clusters can be clearly observed: the left cluster has a wider distribution of points, covering a larger range of PC1 and PC2; On the right side, the cluster is more concentrated around high PC1 values and narrower PC2 ranges. This indicates that PC1 is particularly important in explaining group differences.

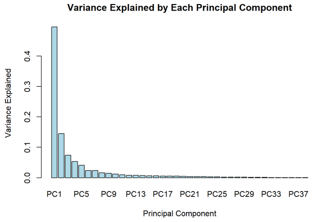


Fig. 9: Variance

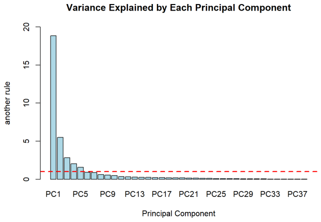


Fig. 10: Character

Regarding the selection of PCA components, based on the Kaiser criterion, we selected the first five principal components that satisfy the eigenvalues being greater than 1 and the cumulative variance explanation rate exceeding 80%. These reduced dimensional data are used for regression analysis and also serve as inputs for cluster analysis.

This is the first preliminary attempt to apply PCA for linear regression after dimensionality reduction, with the goal of predicting the overall performance of players in 2021. The figure above shows the results of the initial linear regression using the first five principal components and their interactions.

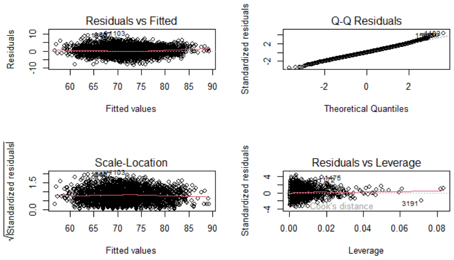


Fig. 11: Preliminary Attempt At Linear Regression

The residual plot is close to the red line, and all the data points in the Q-Q plot lie near the reference line, indicating that the linear regression fit is generally good, although some deviations remain. To address this, we removed irrelevant factors, and the resulting improved linear model, shown in the figure below, demonstrates significant improvements. These enhancements not only refine the model but also boost its predictive ability to some extent.

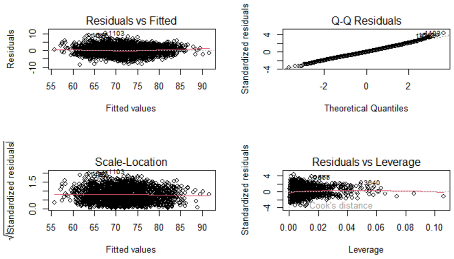
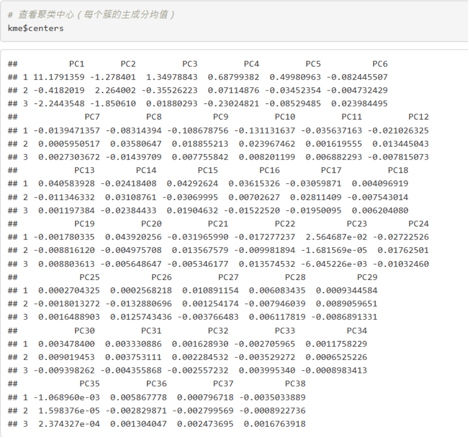


Fig. 12: Final Attempt At Linear Regression

2) K-means Algorithm: The K-mean method can help us classify players into attacking, defensive, and comprehensive types based on their features



```
# 查看聚类中心（每个簇的主成分均值）
kme$centers
```

```
##          PC1       PC2         PC3         PC4         PC5          PC6
## 1 11.1791359 -1.278401  1.34978843  0.68799382  0.49980963 -0.082445507
## 2 -0.4182019  2.264002 -0.35526223  0.07114876 -0.03452354 -0.004732429
## 3 -2.2443548 -1.850610  0.01880293 -0.23024821 -0.08529485  0.023984495
##             PC7         PC8          PC9         PC10         PC11         PC12
## 1 -0.0139471357 -0.08314394 -0.108678756 -0.131131637 -0.035637163 -0.021026325
## 2  0.0005950517  0.03580647  0.018855213  0.023967462  0.001619555  0.013445043
## 3  0.0027303672 -0.01439709  0.007755842  0.008201199  0.006882293 -0.007815073
##            PC13        PC14        PC15        PC16        PC17         PC18
## 1  0.040583928 -0.02418408  0.04292624  0.03615326 -0.03059871  0.004096919
## 2 -0.011346332  0.03108761 -0.03069995  0.00702627  0.02811409 -0.007543014
## 3  0.001197384 -0.02384433  0.01904632 -0.01522520 -0.01950095  0.006204080
##            PC19         PC20         PC21         PC22          PC23        PC24
## 1 -0.001780335  0.043920256 -0.031965990 -0.017277237  2.564687e-02 -0.02722526
## 2 -0.008816120 -0.004975708  0.013567579 -0.009981894 -1.681569e-05  0.01762501
## 3  0.008803613 -0.005648647 -0.005346177  0.013574532 -6.045226e-03 -0.01032460
##            PC25          PC26         PC27         PC28          PC29
## 1  0.0002704325  0.0002568218  0.010891154  0.006083435  0.0009344584
## 2 -0.0018013272 -0.0132880696  0.001254174 -0.007946039  0.0089059651
## 3  0.0016488903  0.0125743436 -0.003766483  0.006117819 -0.0086891331
##            PC30         PC31         PC32         PC33          PC34
## 1  0.003478400  0.003330886  0.001628930 -0.002705965  0.0011758229
## 2  0.009019453  0.003753111  0.002284532 -0.003529272  0.0006525226
## 3 -0.009398262 -0.004355868 -0.002557232  0.003995340 -0.0008983413
##            PC35         PC36         PC37          PC38
## 1 -1.068960e-03  0.005867778  0.000796718 -0.0035033889
## 2  1.598376e-05 -0.002829871 -0.002799569 -0.0008922736
## 3  2.374327e-04  0.001304047  0.002473695  0.0016763918
```

Fig. 13: Cluster Center

In the clustering results, we classified the players based on the clustering centers:

- Category 1: Defensive players (PC1 leads defense with high contribution).
- Category 2: Offensive players (PC1 defense weaker than Category 1, offense more prominent).
- Category 3: Comprehensive players (balanced defense and offense, with no obvious outstanding points).

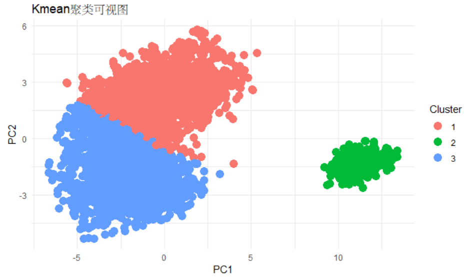
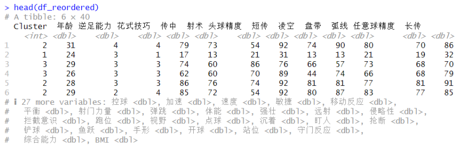


Fig. 14: Visual Result of Cluster

We are added the classification results to the original data, allowing players to be intuitively classified into three types: defensive, offensive, and comprehensive, laying the foundation for subsequent analysis.



```
> head(df_reordered)
# A tibble: 6 × 40
  Cluster  年龄 逆足能力 花式技巧  传中  射术 头球精度  短传  凌空  盘带  弧线 任意球精度  长传
    <int> <dbl>    <dbl>    <dbl> <dbl> <dbl>    <dbl> <dbl> <dbl> <dbl> <dbl>      <dbl> <dbl>
1       2    31        4        4    79    73       54    92    74    90    80         70    86
2       1    24        3        1    17    13       21    31    13    13    21         19    32
3       3    29        3        3    74    60       86    76    66    57    73         68    70
4       3    26        3        3    62    60       70    89    44    74    66         68    79
5       2    28        3        3    86    76       74    92    81    81    77         81    91
6       2    29        2        4    85    72       54    92    80    87    83         77    85
# ℹ 27 more variables: 控球 <dbl>, 加速 <dbl>, 速度 <dbl>, 敏捷 <dbl>, 移动反应 <dbl>,
#   平衡 <dbl>, 射门力量 <dbl>, 弹跳 <dbl>, 体能 <dbl>, 强壮 <dbl>, 远射 <dbl>, 侵略性 <dbl>,
#   拦截意识 <dbl>, 跑位 <dbl>, 视野 <dbl>, 点球 <dbl>, 沉着 <dbl>, 盯人 <dbl>, 抢断 <dbl>,
#   铲球 <dbl>, 鱼跃 <dbl>, 手形 <dbl>, 开球 <dbl>, 站位 <dbl>, 守门反应 <dbl>,
#   综合能力 <dbl>, BMI <dbl>
```

Fig. 15: Supplementary Data

C. Short conclusion

We initially used PCA and k-mean methods to analyze the data, but there was a major issue that we did not notice, which is that we only roughly viewed the data from a macro perspective and still did not classify and analyze it separately according to the actual situation. This is also the priority factor that we need to consider in the next step.

III. Regression Analysis

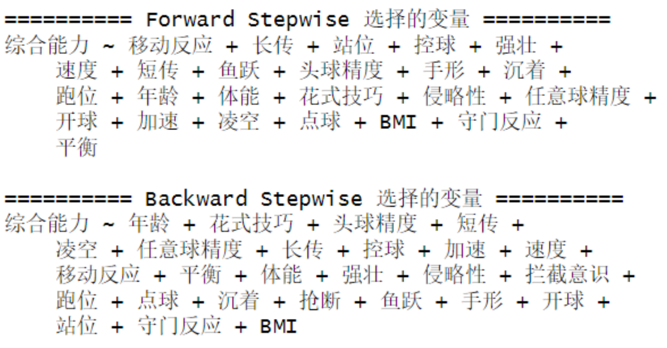
After data pre-processing, we have totally 38 features (BMI added with height, weight and international reputation deleted) to quantify a soccer player's detailed ability in each aspect. To find the integrated ability evaluation standard and better predict the data in FIFA21, we first apply multiple regression. basic assumptions:

A. Model Selection

1) Variable Selection: It is difficult to deal with 38-dimension independent variables, we naturally first try some methods to do a selection so that the model is not too complicated.

a) Stepwise Method: The stepwise selection method is employed to identify and retain the most significant independent variables. This approach iteratively adds or removes predictors based on criteria such as the Akaike Information Criterion (AIC), ensuring an optimal balance between model complexity and explanatory power.

Through the Forward and Backward Stepwise method, 24 and 26 independent variables are selected relatively.



```
========== Forward Stepwise 选择的变量 ==========
综合能力 ~ 移动反应 + 长传 + 站位 + 控球 + 强壮 +
      速度 + 短传 + 鱼跃 + 头球精度 + 手形 + 沉着 +
      跑位 + 年龄 + 体能 + 花式技巧 + 侵略性 + 任意球精度 +
      开球 + 加速 + 凌空 + 点球 + BMI + 守门反应 +
      平衡

========== Backward Stepwise 选择的变量 ==========
综合能力 ~ 年龄 + 花式技巧 + 头球精度 + 短传 +
      凌空 + 任意球精度 + 长传 + 控球 + 加速 + 速度 +
      移动反应 + 平衡 + 体能 + 强壮 + 侵略性 + 拦截意识 +
      跑位 + 点球 + 沉着 + 抢断 + 鱼跃 + 手形 + 开球 +
      站位 + 守门反应 + BMI
```

Fig. 16: Stepwise Results

b) LASSO: LASSO (Least Absolute Shrinkage and Selection Operator) is another widely used technique for variable selection. By applying a penalty to the absolute size of coefficients, LASSO encourages sparsity in the solution, effectively eliminating less relevant variables and retaining only the most important predictors. However, all coefficients are above 0, which corresponds to our intuition that all aspects of ability influence the integrated score.

2) Comparing Models: Based on above variable selection results, we trained 3 models with the same selected data. Since stepwise results are 24 and 26 variables, we use LASSO top 25 variables to train a LASSO model.

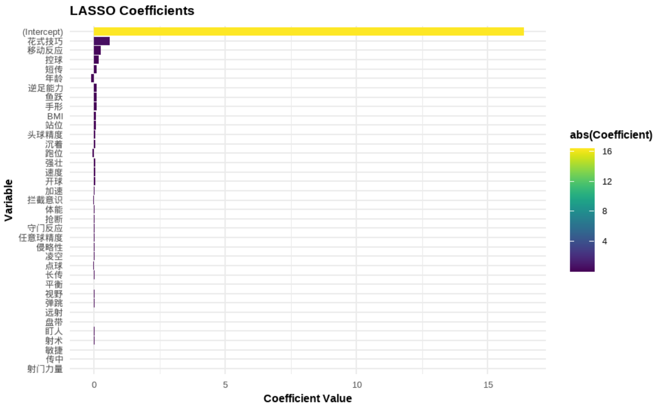
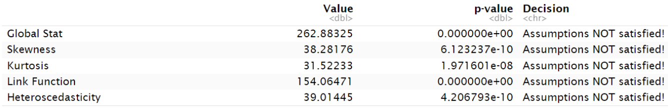


Fig. 17: LASSO Coefficients

Model	Training RMSE	CV RMSE	AIC	R²	R² CV
Forward	2.96	2.96	15926.95	0.7361	0.7312
Backward	2.95	2.95	15914.89	0.7374	0.7323
LASSO	2.95	8.92	15917.33	0.7370	0.7320

TABLE I: Model comparison based on RMSE, AIC, and R-squared metrics.



	Value <dbl>	p-value <dbl>	Decision <chr>
Global Stat	262.88325	0.000000e+00	Assumptions NOT satisfied!
Skewness	38.28176	6.123237e-10	Assumptions NOT satisfied!
Kurtosis	31.52233	1.971601e-08	Assumptions NOT satisfied!
Link Function	154.06471	0.000000e+00	Assumptions NOT satisfied!
Heteroscedasticity	39.01445	4.206793e-10	Assumptions NOT satisfied!

Fig. 18: `gvlma()`

To evaluate model performance, we utilize several metrics, with root mean square error (RMSE) being the most commonly used, Akaike Information Criterion (AIC) and R-squared values are included to provide a comprehensive assessment. Model validation is performed using k-fold cross-validation, implemented with the `crossval()` function from the `bootstrap` package. AIC is used to measure the quality of a model in terms of both fit and complexity, with lower values indicating a better balance between model simplicity and fit. Here, the 3 models have simlar AIC. But for the true LASSO model which adds a regularized term, the AIC is only about 6929.

R² indicates the proportion of variance explained by the model, with values closer to 1 suggesting better model fit and backward model has the highest one.

Finally, the backward model is chosen with least RMSE and highest R².

B. Evaluate the Assumptions

First we use `gvlma()` to test the model globally. The result is pretty poor. All assumptions are not satisfied. We test the above assumptions of an OLS model with backward variables step by step:

1) Normality: First we apply Q-Q plot to compare the theoretical quantiles of a normal distribution with the sample quantiles of the studentized residuals. It looks almost matches the assumption except the tails.

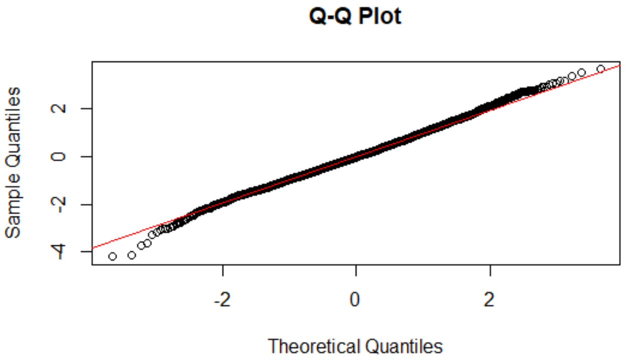
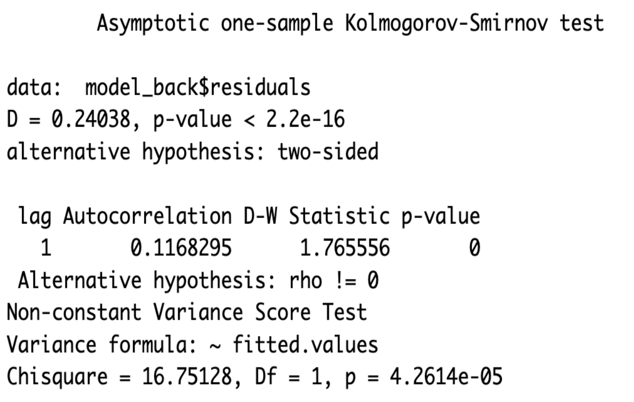


Fig. 19: Q-Q Plot

However, the ks test shows that this model does not pass the normality test.

2) Independence: By `Dulbin-Waston` test, these variables show high correlation and the p-value is directly 0. This model also violates our independence assumption.

3) Linearity: By examining the component + residual (cr) plots below, it can be observed that most independent variables exhibit a linear relationship with the dependent variable. However, certain variables demonstrate severse deviations



```
        Asymptotic one-sample Kolmogorov-Smirnov test

data:  model_back$residuals
D = 0.24038, p-value < 2.2e-16
alternative hypothesis: two-sided

 lag Autocorrelation D-W Statistic p-value
   1       0.1168295      1.765556       0
 Alternative hypothesis: rho != 0
Non-constant Variance Score Test
Variance formula: ~ fitted.values
Chisquare = 16.75128, Df = 1, p = 4.2614e-05
```

Fig. 20: Test

from linearity, suggesting the need for further investigation or potential transformation.

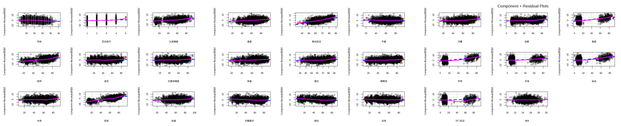


Fig. 21: Component + Residual plots for selected independent variables.

4) Homoscedasticity: By `ncvTest()` result showed in above figure, the backward model also does not pass the homoscedaasticity assumption.

5) Test Multicollinearity: Through VIF metric, if we delete variables with value larger than 2, only 2 variables(Age and BMI) will reserve. So we just delete those with value larger than 10, but R^2 of this model drops to 0.63, which is unacceptable when trying to build a model. And a ridge model is constructed to solve this problem if we want to do prediction(see fig. [**?**]).

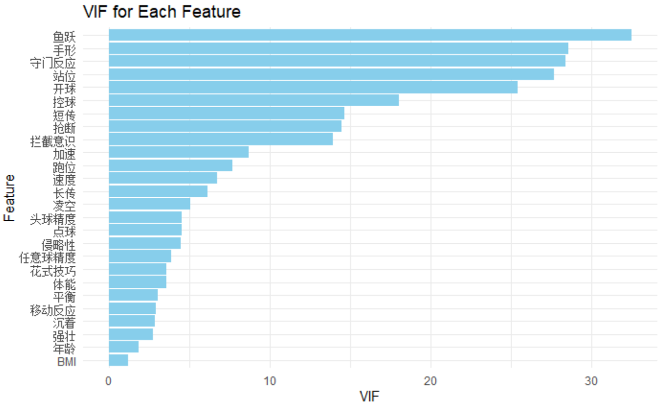


Fig. 22: VIF Values and the Process of Ridge Regression

C. Corrective Methods

In this section, since the previous model could not pass the parametric test, some more corrected models are built:

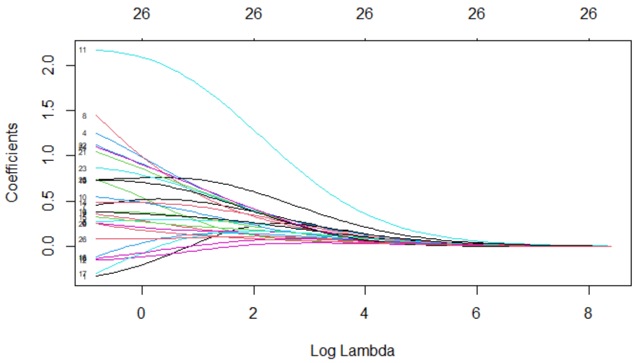


Fig. 23: Process of Ridge Regression

1) model2: By `boxTidwell()`, all variables that show non-linear relationship are transformed (we selected them manully). And we draw the crplots below to check the linearity, finding that some variables do not need to be transformed since the transformed one like ”Goalkeeper Reaction” does not perform better.

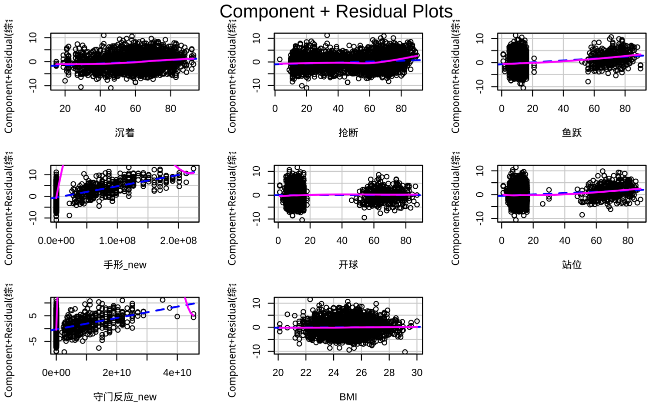


Fig. 24: CrPlot

2) model3: Based on model2’s crplots, only 2 variables(Short Passes and Moving Reaction) are transformed from the backward model.

3) model4: Based on model3, to achieve the normality assumption, use `powerTransform()` and do power transformation with suggested $\lambda = 0.5$to the dependent variable. The new Q-Q plot is below:
Though the plot seems to be better, it does not pass the normality test again.

4) model5: Based on model3, `boxcox()` is used to achieve homoscedasticity. The suggested $\lambda = 0.3434$. Again, the model does not pass our `ncvTest()`

- Non-constant Variance Score Test
  Variance formula: `fitted.values`
  Chisquare = **72.35854**, Df = **1**, p = **2.22e-16**

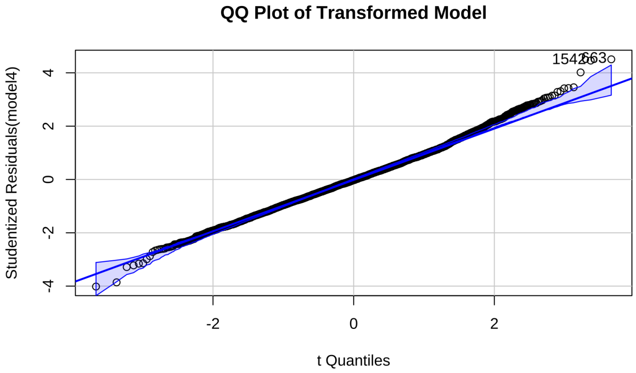


Fig. 25: Q-Q Plot for Model4

D. Final Model

Since this is a model aiming at quantify and predict players' ability, the prediction performance is the most important. And cross-validation results show that the error almost does not change among models, we may choose the original backward model for prediction as it is the simplest. Compared to the original model, the number of features are less while R^2 only drops about 0.002.

Model	CV_RMSE	R2
model_0	2.988708	0.7331283
model_back	2.987292	0.7319983
model2	2.985776	0.7822480
model3	2.989324	0.7434827
model4	2.988438	0.7426850
model5	2.985389	0.7422501

TABLE II: Model Comparison

1) Relative importance: The relative importance of top 10 influential variables are shown below:

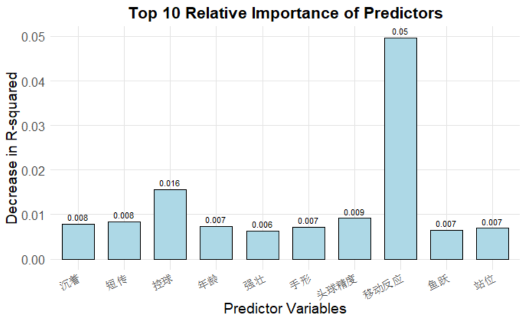


Fig. 26: Top 10 Variables

2) Influential Observations: And the plots of influential observation show that there are few outliers. Since the feature "International Reputation" actually influences the score, the results are acceptable because we simplified our features.

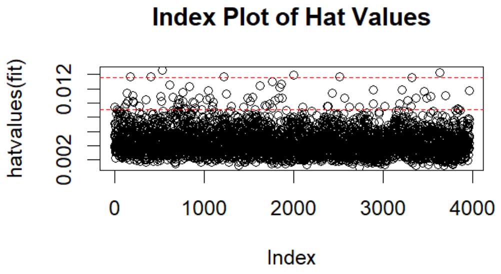


Fig. 27: Plot of Hat Values

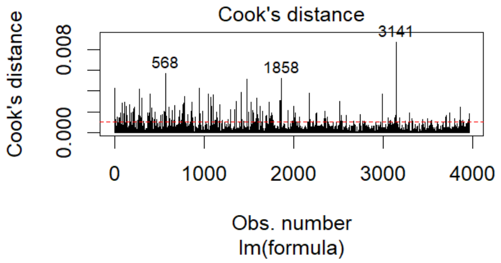
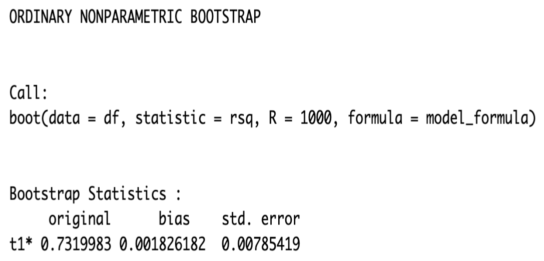


Fig. 28: Plot of Cook's Distance

IV. Bootstrap

To address the violation of assumptions in our data, we utilize bootstrap resampling to compute statistics and construct confidence intervals. Specifically, we perform a bootstrap simulation with 1000 repetitions to obtain the R^2 statistic and the linear regression coefficients.

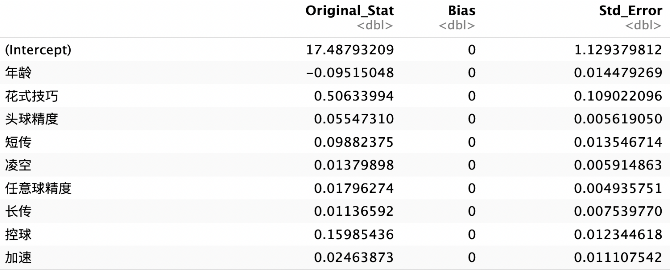


```
ORDINARY NONPARAMETRIC BOOTSTRAP


Call:
boot(data = df, statistic = rsq, R = 1000, formula = model_formula)


Bootstrap Statistics :
     original       bias    std. error
t1* 0.7319983 0.001826182  0.00785419
```

Fig. 29: Bootstrap Results for R^2

The R^2 statistic obtained from our linear regression model serves as the original estimate in the `boot()` method, as the resampling procedure does not involve repeated observations. The bootstrap results show low bias and standard error for R^2. Despite the violation of the normality assumption, the R^2 derived from the linear regression model is reliable based on the bootstrap findings.

Similarly, the biases in the linear regression coefficients are negligible, as indicated by the bootstrap analysis. The consistency of the results further supports the validity of the estimates obtained from the linear regression model. This



	Original_Stat <dbl>	Bias <dbl>	Std_Error <dbl>
(Intercept)	17.48793209	0	1.129379812
年龄	-0.09515048	0	0.014479269
花式技巧	0.50633994	0	0.109022096
头球精度	0.05547310	0	0.005619050
短传	0.09882375	0	0.013546714
凌空	0.01379898	0	0.005914863
任意球精度	0.01796274	0	0.004935751
长传	0.01136592	0	0.007539770
控球	0.15985436	0	0.012344618
加速	0.02463873	0	0.011107542

Fig. 30: Bootstrap Results for Linear Coefficients

demonstrates that the violation of the assumptions does not significantly compromise the credibility of the regression outcomes.

V. Classification

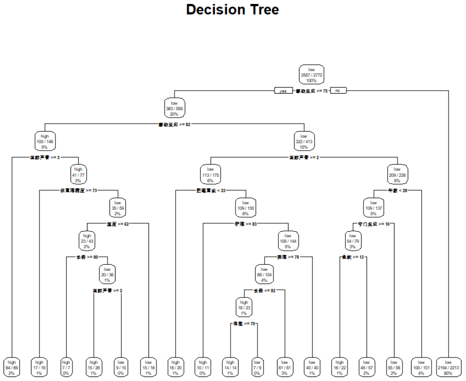
A. Data Processing

The FIFA dataset was preprocessed by:

- Removing irrelevant columns such as player names and club affiliations.
- Creating a binary classification target to distinguish high-performing players (scores bigger than 76).
- The threshold of 76 is selected from the upper quartile of the overall ability scores in the original dataset. During the classification process, cross-validation is required, so 70 percents of the original dataset is chosen as the training set, with the remaining 30 percents serving as the testing set.
- Specifically, to account for different positions, the original dataset is filtered to exclude substitutes, and the above procedure is repeated separately for forwards, midfielders, defenders, and goalkeepers.

B. Decision Tree Analysis

1) Classical Decision Trees: Figure 31 shows the decision tree structure and initial variable relationships.This figure illustrates the classical decision tree without position differentiation. It highlights the importance of variables such as Reactions, International Reputation, Age (normal), Free Kick Accuracy, Slide Tackle, and Goalkeeper Reflexes (unusual). However, it is unlikely for a player to excel in all these variables simultaneously.



(a) Decision Tree

Fig. 31: Decision Tree Analysis

Therefore, we need to analyze the data by categorizing players into Forwards, Midfielders, Defenders, and Goalkeepers.

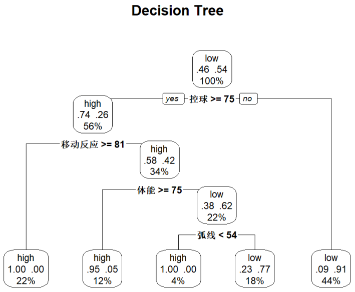


Fig. 32: Classical Decision Trees for Forwards

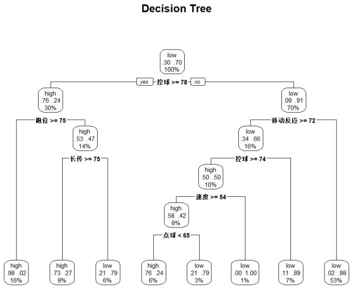


Fig. 33: Classical Decision Trees for Midfielders

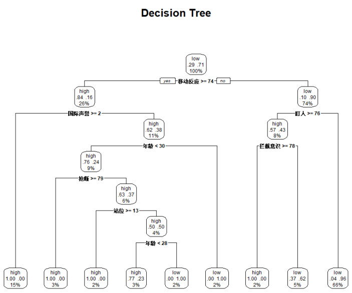


Fig. 34: Classical Decision Trees for Defenders

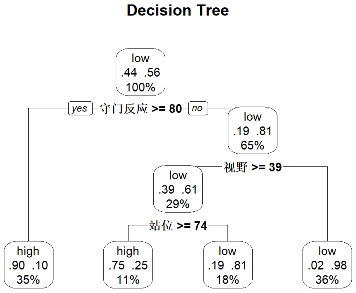


Fig. 35: Classical Decision Trees for Goalkeepers

Forwards: Ball Control, Reactions, Stamina, and Curve. Midfielders: Ball Control, Reactions, Positioning, and Long Passing. Defenders: Reactions, Marking, Interceptions, and Stand Tackle. Goalkeepers: Goalkeeper Reflexes, Vision, and Positioning, which are well-aligned with their role-specific requirements.

2) Conditional Inference Trees: Figure 32 shows the conditional inference tree without position differentiation. It highlights variables such as Handling, Diving, Ball Control, and Acceleration as important, which are still unreasonable.

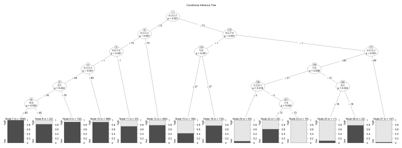


Fig. 36: Comprehensive Analysis of Decision Trees

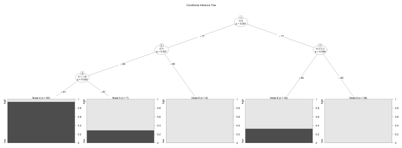


Fig. 37: Conditional Inference Tree for Forwards

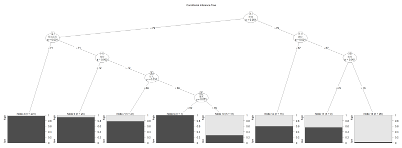


Fig. 38: Conditional Inference Tree for Midfielders

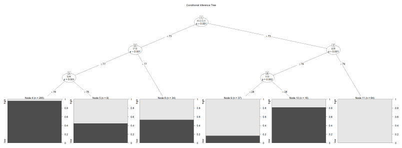


Fig. 39: Conditional Inference Tree for Defenders

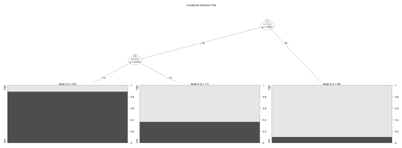
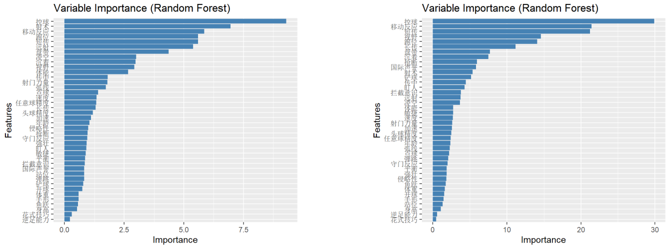


Fig. 40: Conditional Inference Tree for Goalkeepers

After differentiating by position, the key variables align better with role-specific requirements Forwards: Finishing and Shot Power. Midfielders: Short Passing, Marking, and Positioning. Defenders: Slide Tackle and Stand Tackle. Goalkeepers: Goalkeeper Reflexes. These variables are much more consistent with the expectations for each position.

C. Random Forest Analysis

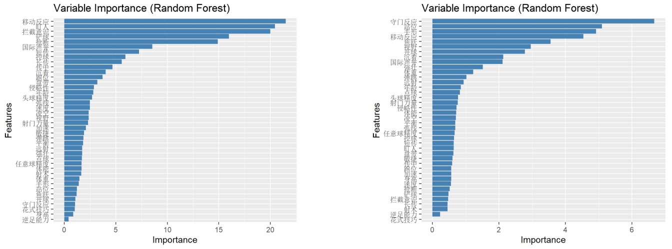
Figure **??** highlights the variable importance and detailed analysis.



(a) Importance variables for Forwards

(b) Importance variables for Midfielders

Fig. 41: Random Forest Analysis (Part 1)



(a) Importance variables for Defenders

(b) Importance variables for Goalkeepers

Fig. 42: Random Forest Analysis (Part 2)

The important variables identified are similar to the conclusions from the previous decision tree analysis. Additionally, the importance function of the Random Forest model highlights unimportant variables. For example, height and weight (BMI) have little influence on the overall ability for players in all positions. Skills such as skill moves and weak foot ability are also not considered key evaluation metrics.

Forwards generally do not require attributes like tackling, interceptions, or diving. Midfielders and defenders relatively do not need acceleration or sprint speed, while goalkeepers do not rely on long passing or finishing. Notably, the importance of skill moves for goalkeepers is zero.

D. Model Selection

1) Comparison: Figures 43 and 44 present the performance of different models. Model selection was performed using the performance function from Chapter 10:

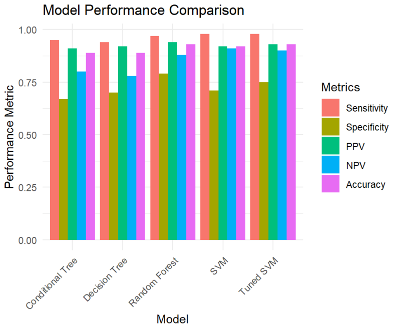
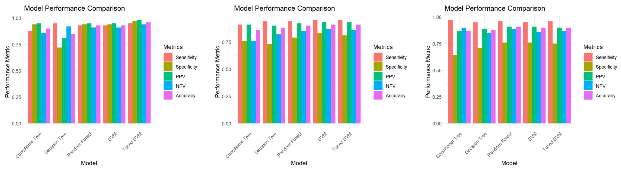


Fig. 43: Model Performance Comparison(overall)

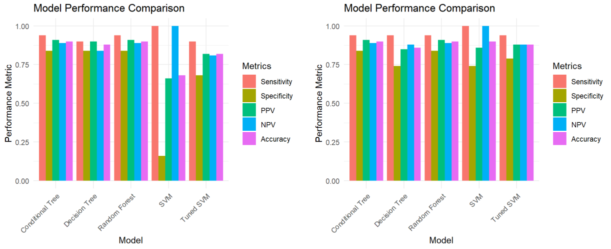


(a) Forwards

(b) Midfielders

(c) Defenders

Fig. 44: Detailed Performance Metrics



(a) Goalkeepers(before)

(b) Goalkeepers(after)

Fig. 45: Additional Metrics for Model Selection

2) Conclusion:

- Sensitivity: Ensures that key high-scoring players (e.g., potential star players) are not missed.
- Specificity: Ensures that ordinary or low-scoring players are not misclassified as high-scoring players, reducing selection errors.
- Positive Predictive Value (PPV): Ensures that predicted high-scoring players indeed have strong actual performance, increasing trust in the model.
- Negative Predictive Value (NPV): Ensures that predicted low-scoring players truly have weaker performance, helping avoid wasting resources on misclassified players.
- For goalkeepers. The accuracy of SVM for goalkeepers drops significantly, primarily because quantitative variables (e.g., skill moves) were not excluded. After removing these irrelevant variables, the performance returns to normal.
- Overall, Random Forest and the tuned SVM perform well across all metrics, which exhibited the highest accuracy and specificity, making them the most reliable predictors.