Analysis of La Liga Football League Data

Ca’ Foscari University of Venice

Department of Environmental Sciences, Informatics and Statistics

Course CM0471 : STATISTICAL INFERENCE AND LEARNING

Academic Year 2024 - 2025

Student Ajay Prakash Nair 898141

Contents

1 Introduction **1**

2 Steps Involved in the Process **2**

2.1 Data Acquisition 2

2.2 Data Preparation and Cleaning 3

2.3 Exploratory Data Analysis (EDA) 3

2.4 Remove Multicollinear Variables 5

2.5 Model Building 5

2.5.1 Logistic Regression 5

2.6 Model Interpretation 8

2.7 Model Comparison 9

2.7.1 Stepwise Model 10

2.7.2 Lasso Regression 11

2.7.3 Ridge logistic regression 12

2.7.4 Linear Discriminant Analysis 14

2.7.5 Naive Bayes 15

3 Conclusion **16**

List of Figures

2.1 Match Results Visualization 4

2.2 Goals Scored Distribution 4

2.3 Home Team Goals vs. Match Outcome 4

2.4 Home Team Shots vs. Match Outcome 4

2.5 Correlation Matrix of Match Statistics 5

2.6 Correlation Matrix with Multicollinear Variables 6

2.7 Removed Multicollinear Variables 6

2.8 HST Effect Plot 8

2.9 AST Effect Plot 8

2.10 ROC Curves for Cleaned Multinomial Model 9

2.11 ROC Curves for Multinomial Model 9

2.12 ROC Curves for Multinomial Model with Interaction Terms 9

2.13 ROC Curves for Cleaned Multinomial Model with Interaction Terms . 9

2.14 ROC Curves for Stepwise Model 10

2.15 LASSO Plots 11

2.16 LASSO Best Lambda 12

2.17 LASSO ROC Curves 12

2.18 Ridge Plots 13

2.19 Ridge Best Lambda 13

2.20 Ridge ROC Curves 14

2.21 LDA ROC Curves 14

2.22 Naive Bayes ROC Curves 15

List of Tables

1.1 Description of match statistics in the dataset 1

2.1 AIC and Degrees of Freedom for Various Multinomial Models 8

2.2 Full Model 10

2.3 Full Interaction Model 10

2.4 Cleaned Model Confusion Matrix 10

2.5 Cleaned Interaction Model Confusion Matrix 10

2.6 Stepwise Model Confusion Matrix 11

2.7 Lasso Model Confusion Matrix 11

2.8 Ridge Confusion Matrix 14

2.9 LDA Confusion Matrix 14

2.10 Naive Bayes Confusion Matrix 15

3.1 Model Performance Metrics for Different Models 16

Chapter 1

Introduction

This project aims to delve into the realm of La Liga through a statistical lens, exploring various aspects such as team performance metrics and predictive modeling of match outcomes. The dataset comprises detailed statistical records of matches played in the Spanish La Liga over the past decade.

Label	**Description**
Date	Date of the match
HomeTeam	Home Team of the match
AwayTeam	Away Team of the match
FTHG	Full Time Home Team Goals
FTAG	Full Time Away Team Goals
FTR	Full Time Result (H = Home Win, D = Draw, A = Away Win)
HTHG	Half Time Home Team Goals
HTAG	Half Time Away Team Goals
HTR	Half Time Result (H = Home Win, D = Draw, A = Away Win)
HS	Home Team Shots
AS	Away Team Shots
HST	Home Team Shots on Target
AST	Away Team Shots on Target
HF	Home Team Fouls Committed
AF	Away Team Fouls Committed
HC	Home Team Corners
AC	Away Team Corners
HY	Home Team Yellow Cards
AY	Away Team Yellow Cards
HR	Home Team Red Cards
AR	Away Team Red Cards

Table 1.1: Description of match statistics in the dataset

Chapter 2

Steps Involved in the Process

The steps involved in the analysis process are as follows:

- **Data Acquisition:** Collect data from reliable sources such as La Liga's official records.
- **Data Preparation and Cleaning:** Handle missing values, correct errors, and transform variables for analysis.
- **Exploratory Data Analysis (EDA):** Visualize trends, identify patterns, and summarize key statistics.
- **Model Building:** Develop statistical models to analyze and predict outcomes.
- **Model Evaluation:** Evaluate model performance using metrics like accuracy, specificity, and sensitivity.
- **Results Interpretation:** Interpret insights and summarize findings to address the project's objectives.

2.1 Data Acquisition

The dataset is sourced from `http://www.football-data.co.uk/`, which provides comprehensive match information, including results, corner kicks, and disciplinary actions.

2.2 Data Preparation and Cleaning

The CSV file downloaded from the website contains data for each season of the Spanish La Liga, starting from the 2009/2010 season and spanning up to the 21st of October of the 2024/2025 season. The data was filtered to retain only essential match statistics for subsequent analysis.

Data Cleaning Steps

- **Missing Values:** Checked for and handled missing values appropriately using the following command:

  ```
  missing_values <- colSums(is.na(football_data))
  ```

- **Data Types:** Converted necessary columns to appropriate data types, particularly the response variable `FTR`, which was converted to a factor.
- **Duplicates:** Checked for and removed any duplicate entries to ensure data integrity.

2.3 Exploratory Data Analysis (EDA)

Exploratory Data Analysis is crucial for understanding the dataset. It includes visualizing trends, identifying patterns, and summarizing statistics.

Univariate Analysis

- **Distribution of Match Outcomes:** A bar plot was created to visualize the distribution of match outcomes (Home Win, Draw, Away Win).
- **Goals Scored Distribution:** Histograms were plotted to show the distribution of goals scored by home and away teams.

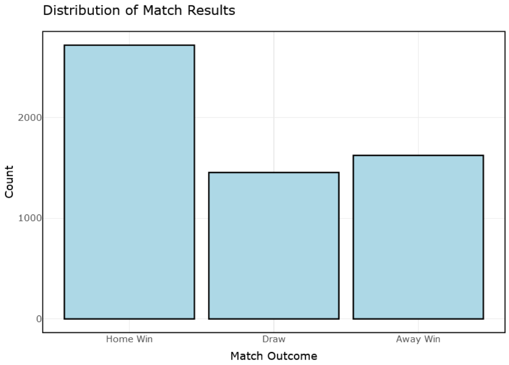

Figure 2.1: Match Results Visualization

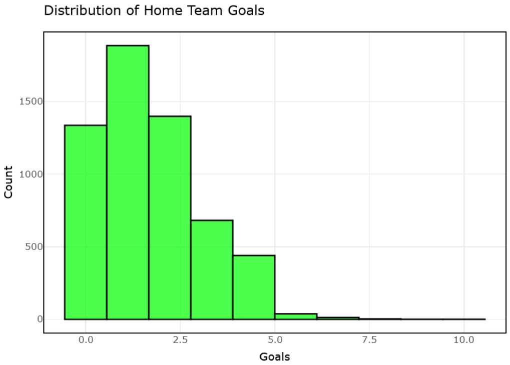

Figure 2.2: Goals Scored Distribution

Bivariate Analysis

- **Goals vs. Match Outcome:** Boxplot was created to analyze the relationship between goals scored by home teams and match outcomes.
- **Shots vs. Match Outcome:** Boxplot was created to analyze the relationship between shots taken by home teams and match outcomes.

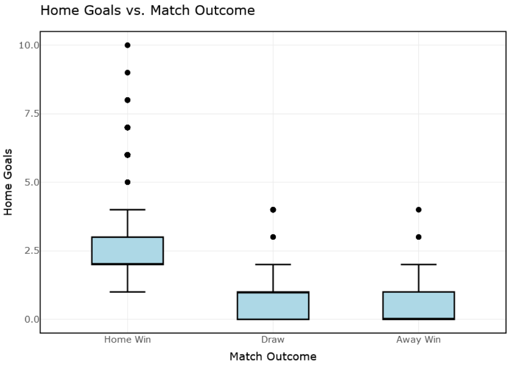

Figure 2.3: Home Team Goals vs. Match Outcome

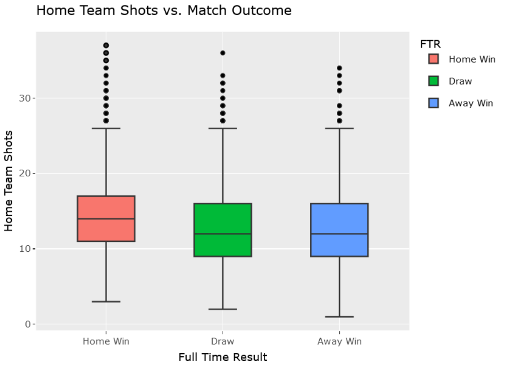

Figure 2.4: Home Team Shots vs. Match Outcome

Correlation Matrix

A correlation matrix was generated to analyze the relationships between different variables such as goals scored, shots taken, fouls committed, and other match statistics.

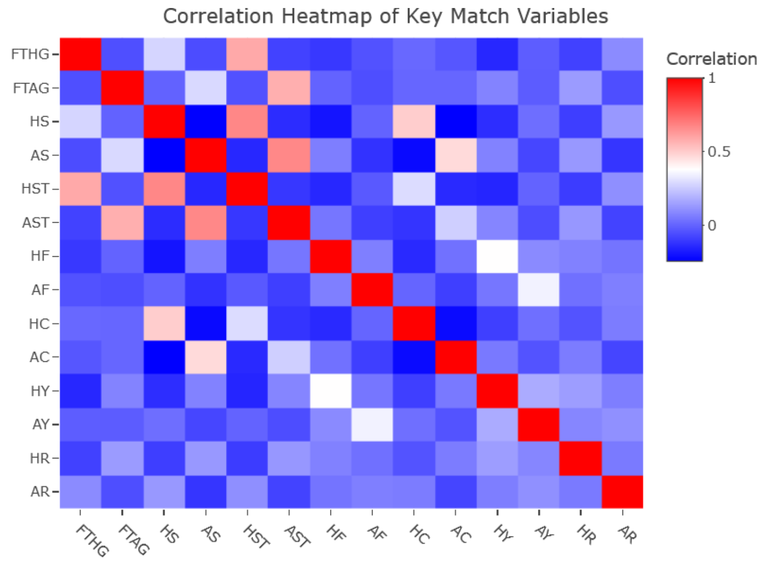

Figure 2.5: Correlation Matrix of Match Statistics

2.4 Remove Multicollinear Variables

A heatmap was plotted to visualize the strength and direction of these correlations, where green indicates a positive correlation, red represents a negative correlation, and white indicates little to no correlation. To address multicollinearity, highly correlated variables were identified and removed, and the correlation matrix was recalculated. This process helps reduce multicollinearity, improving the reliability of regression models by eliminating variables that could introduce instability or inflate standard errors.

2.5 Model Building

In this phase, statistical models are developed to analyze and predict outcomes.

2.5.1 Logistic Regression

Logistic regression was performed to predict match outcomes based on various predictors, including home team statistics, away team statistics, and match-related features. The model was evaluated using the Akaike Information Criterion (AIC) to assess the goodness of fit, with a lower AIC indicating a better model. Additionally,

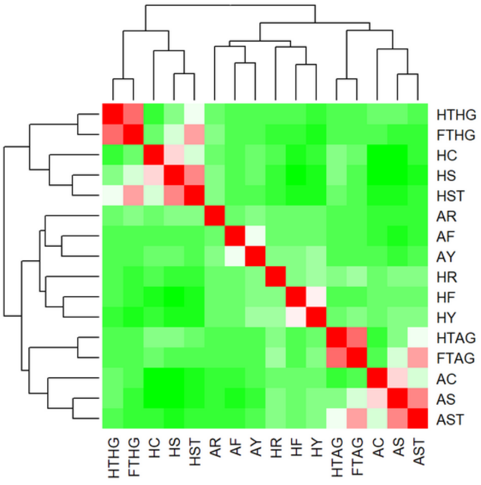

Figure 2.6: Correlation Matrix with Multicollinear Variables

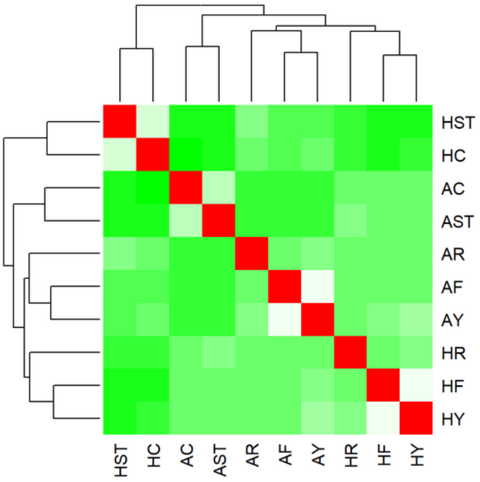

Figure 2.7: Removed Multicollinear Variables

the Variance Inflation Factor (VIF) was calculated to check for multicollinearity between the predictors. A high VIF value suggests a potential issue of multicollinearity, which can distort the estimates of regression coefficients and impact model interpretation.

Releveling the Reference Level

We begin by re-leveling the reference level of the outcome variable *FTR* (Full-Time Result) to “Draw”:

```
train_data$FTR <− relevel(factor(train_data$FTR), ref = ”
    Draw”)
```

Fitting the Initial Model

The initial multinomial logistic regression model is fit using various predictor variables:

```
multinom_model <− multinom(FTR ~ HTHG + HTAG + HTR + HS + AS
    + HST + AST + HF + AF + HC + AC + HY + AY + HR + AR,
    data = train_data)
```

Model Output and Interpretation

We generated the model summary, highlighting key findings: Half-time goals (HTHG, HTAG) and results (HTR) significantly influence match outcomes, with shots on target (HST, AST) also playing a crucial role. Red cards (HR, AR) have varying impacts, while other statistics, such as fouls, yellow cards, and corners, show minimal effect.

Checking for Multicollinearity

We calculate the Variance Inflation Factor (VIF) to check for multicollinearity. Some variables, such as HTR (Half-Time Result), have a high VIF of 49.97. Additionally, HS and AS have VIFs greater than 16, indicating significant multicollinearity with other predictors in the model.

Cleaning the Model

We update the model by removing highly correlated predictors and recheck the VIF:

```
multinom_model.clean <- update(multinom_model, .~.-HTR -HS -
    AS)
```

Adding Interaction Terms

We are adding the following interaction terms : HTHG × HTAG: Home half-time goals × Away half-time goals HST × AST: Home shots on target × Away shots on target HR × AR: Home red cards × Away red cards HY × AY: Home yellow cards × Away yellow cards.

```
multinom_model.inter <- update(multinom_model, . ~ . + HTHG:
    HTAG + HST:AST + HR:AR + HY:AY)
```

The refitted model with interaction terms is analyzed. HST:AST, HY:AY appear to be not so much significant. So, we removed it.

We calculated the AIC for all the 3 models and By looking at the AIC the best model remain the initial full model. Now let's try adding interaction terms to the clean model. Now let's add the following interactions:

```
multinom_model.clean.inter <- update(multinom_model.clean,
    . ~ . + HTHG:HTAG + HST:AST + HR:AR + HY:AY)
```

Model	df	AIC
multinom_model	34	5860.762
multinom_model.inter	38	5863.758
multinom_model.clean.inter	30	5899.191
multinom_model.clean	22	5899.844

Table 2.1: AIC and Degrees of Freedom for Various Multinomial Models

The full model is just slightly better with respect to the full model with interaction terms.

2.6 Model Interpretation

We have decided to interpret the model multinom_model based on its lower AIC value of 5860.762, which suggests that it strikes the best balance between model fit and complexity.

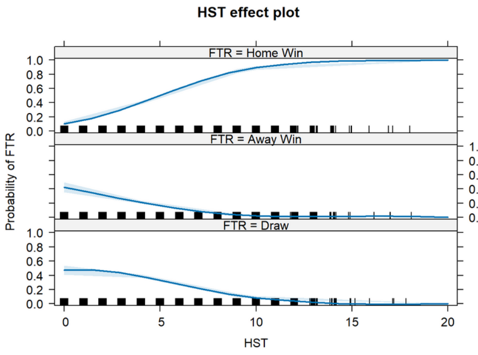

Figure 2.8: HST Effect Plot

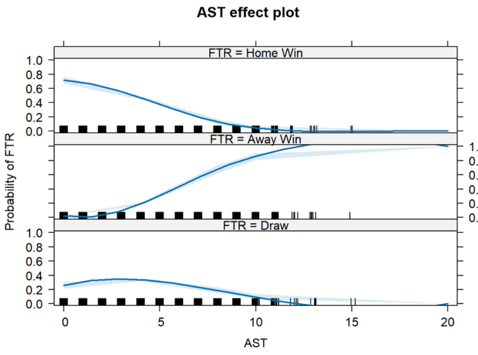

Figure 2.9: AST Effect Plot

2.7 Model Comparison

Let's compare all the models done so far in terms of prediction power to be able to compare them with other types of models:

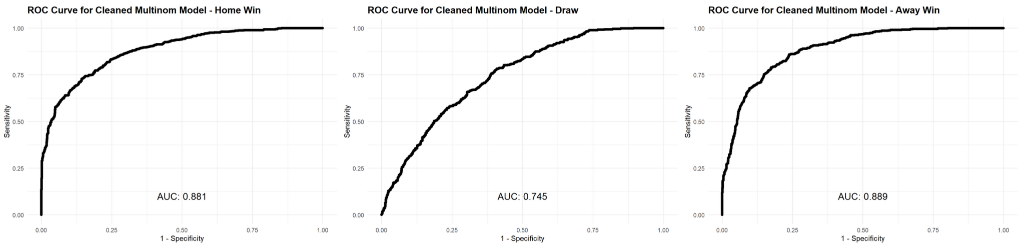

Figure 2.10: ROC Curves for Cleaned Multinomial Model

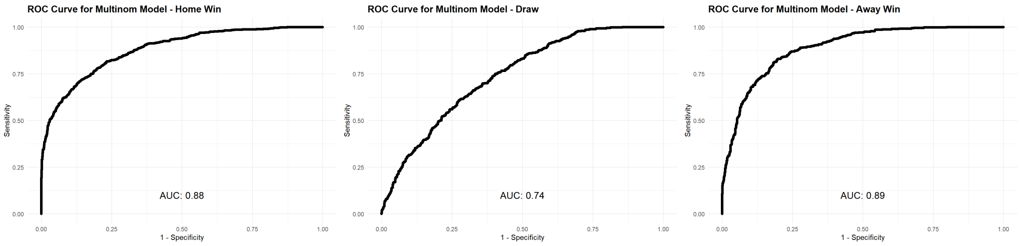

Figure 2.11: ROC Curves for Multinomial Model

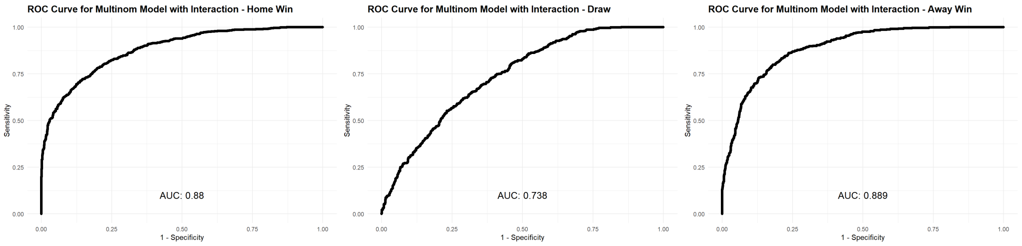

Figure 2.12: ROC Curves for Multinomial Model with Interaction Terms

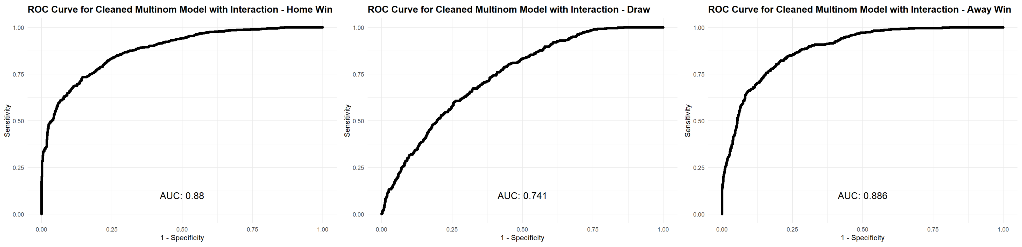

Figure 2.13: ROC Curves for Cleaned Multinomial Model with Interaction Terms

Now let's look at the confusion matrix for all the 4 models :

	Away Win	**Draw**	**Home Win**
Away Win	361	121	65
Draw	64	148	85
Home Win	62	167	666

Table 2.2: Full Model

	Away Win	**Draw**	**Home Win**
Away Win	358	122	67
Draw	69	147	81
Home Win	60	167	668

Table 2.3: Full Interaction Model

	Away Win	**Draw**	**Home Win**
Away Win	357	120	66
Draw	67	145	66
Home Win	63	171	684

Table 2.4: Cleaned Model Confusion Matrix

	Away Win	**Draw**	**Home Win**
Away Win	361	123	67
Draw	68	147	79
Home Win	58	166	670

Table 2.5: Cleaned Interaction Model Confusion Matrix

2.7.1 Stepwise Model

The stepwise model selection was performed using the 'stepAIC' function, which iteratively adds and removes predictors based on the Akaike Information Criterion (AIC) to optimize the model. The process was conducted in both forward and backward directions, with progress displayed at each step. The final model resulted in an AIC of 2476.563, which was better than all the other models.

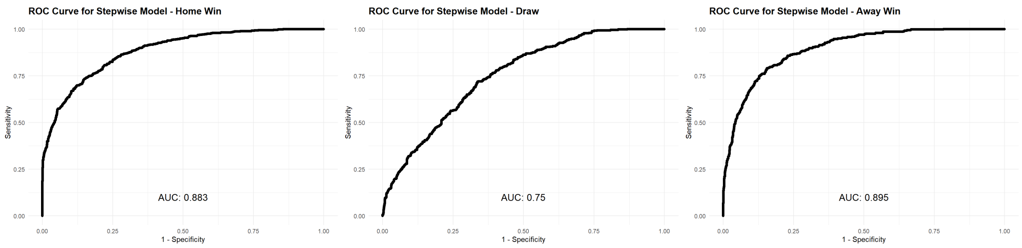

Figure 2.14: ROC Curves for Stepwise Model

	Away Win	Draw	Home Win
Away Win	351	102	46
Draw	74	159	91
Home Win	62	175	679

Table 2.6: Stepwise Model Confusion Matrix

2.7.2 Lasso Regression

The data was prepared by creating a design matrix from the training dataset, scaling the predictors, and fitting a multinomial logistic regression model using LASSO regularization with the glmnet package. The resulting model was visualized by plotting coefficient paths against the regularization parameter (λ).

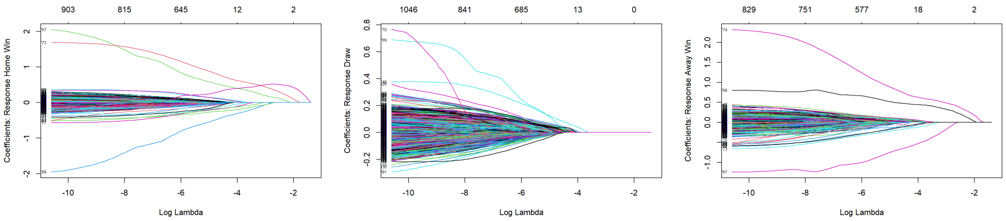

Figure 2.15: LASSO Plots

Cross-validation was performed to select the optimal λ for the LASSO model using the ‘cv.glmnet‘ function, which evaluates performance across a range of regularization values. The results were visualized with a plot showing the cross-validated error for different λ values.

The optimal λ minimizing the cross-validated mean square error (λ_{min}) was found to be 0.01153638. Additionally, a more regularized value, λ_{1se}, corresponding to one standard error from the minimum, was determined as 0.01673731.

Now we apply prediction to the test set, resulting in the following confusion matrix.

Prediction	Away Win	Draw	Home Win
Away Win	361	117	65
Draw	65	148	78
Home Win	61	171	673

Table 2.7: Lasso Model Confusion Matrix

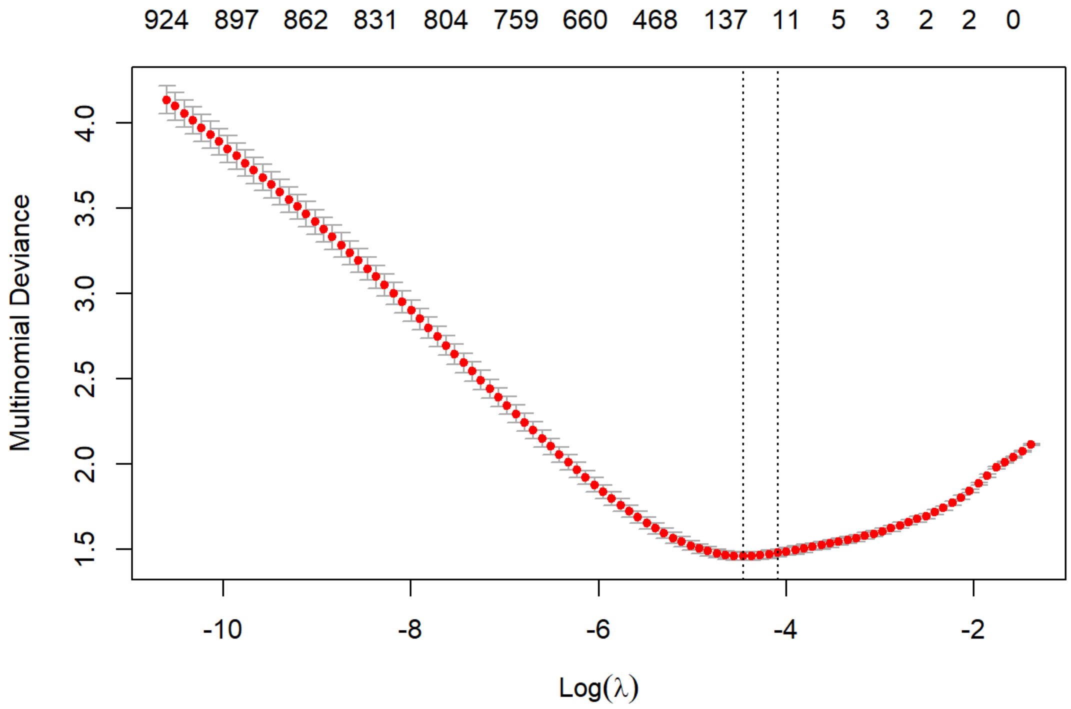

Figure 2.16: LASSO Best Lambda

Then we apply the roc function to obtain the best threshold value.

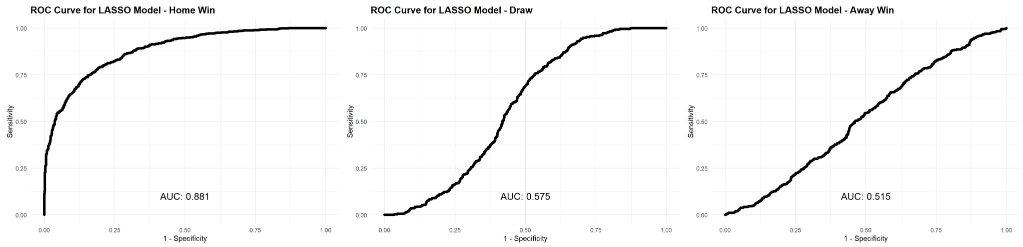

Figure 2.17: LASSO ROC Curves

2.7.3 Ridge logistic regression

The ridge regression model is now applied, starting with the formula containing all predictors and interaction terms. A design matrix is created from the training data, and a multinomial ridge regression model is fitted using the ‘glmnet‘ function with ‘alpha = 0‘. The plot of model coefficients against the regularization parameter (λ) is generated to visualize the effect of regularization.

The optimal value of λ for the ridge regression model is determined using cross-

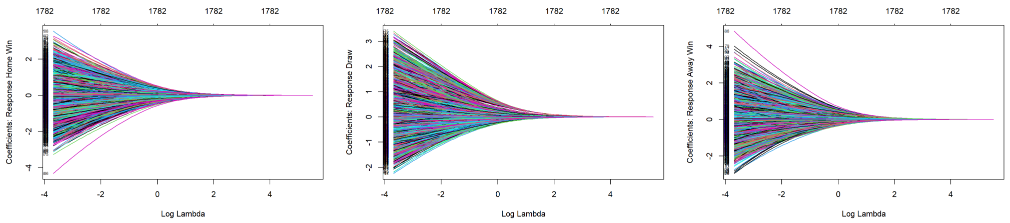

Figure 2.18: Ridge Plots

validation via the ‘cv.glmnet‘ function. The plot generated illustrates the cross-validated mean squared error against different values of λ, helping identify the value that minimizes the error.

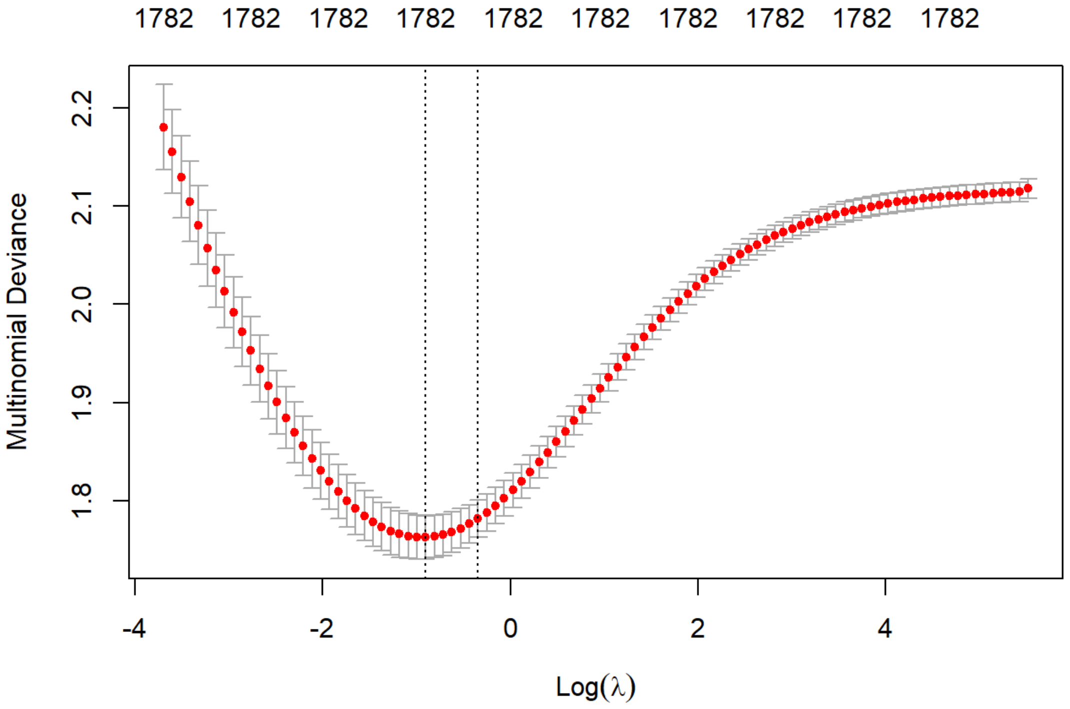

Figure 2.19: Ridge Best Lambda

The value of λ that minimizes the ridge regression cross-validated mean squared error is $\lambda_{\text{min}} = 0.4050644$. However, the selected λ is $\lambda_{1\text{se}} = 0.7078615$, which represents the simplest model within one standard error of the minimum error.

Now we apply prediction to the test set, resulting in the following confusion matrix.

Then we apply the roc function to obtain the best threshold value.

	Away Win	**Draw**	**Home Win**
Away Win	288	114	72
Draw	81	81	83
Home Win	118	241	661

Table 2.8: Ridge Confusion Matrix

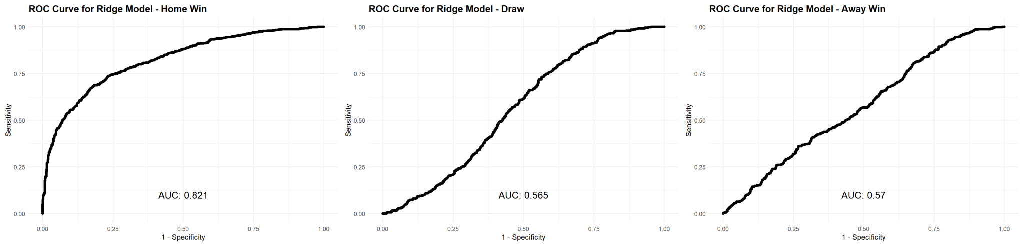

Figure 2.20: Ridge ROC Curves

2.7.4 Linear Discriminant Analysis

This section involves applying Linear Discriminant Analysis (LDA), a generative model, using the full set of predictors. The LDA model is subsequently trained on the scaled training data to predict the target variable ‘FTR‘. The process begins with predicting the classes of the test data using the trained Linear Discriminant Analysis (LDA) model. The confusion matrix is then generated to evaluate the performance of the LDA model by comparing the predicted classes with the actual FTR values from the test data.

	Away Win	**Draw**	**Home Win**
Away Win	308	124	96
Draw	114	145	154
Home Win	65	167	566

Table 2.9: LDA Confusion Matrix

Now, let’s plot ther ROC Curve :

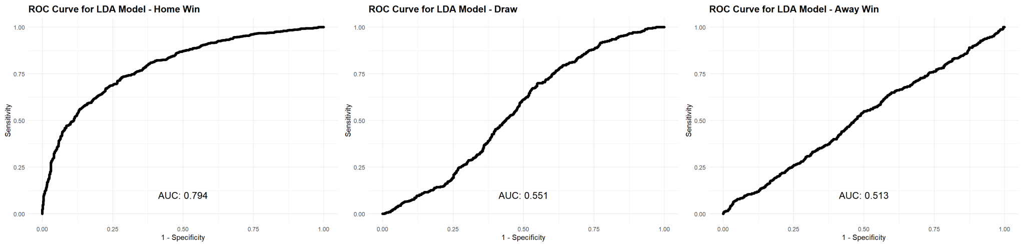

Figure 2.21: LDA ROC Curves

2.7.5 Naive Bayes

The Naive Bayes algorithm is used to create a model for predicting the FTR (Full-Time Result) based on the training data. This model assumes that the features are conditionally independent given the class label.

The Naive Bayes model was applied to the test dataset to predict the class labels for the target variable FTR (Full Time Result). The confusion matrix for the model was generated, showing the distribution of predicted versus actual classes.

	Away Win	**Draw**	**Home Win**
Away Win	336	114	59
Draw	113	211	171
Home Win	38	111	586

Table 2.10: Naive Bayes Confusion Matrix

This was followed by the ROC curve, which visualized the model's performance.

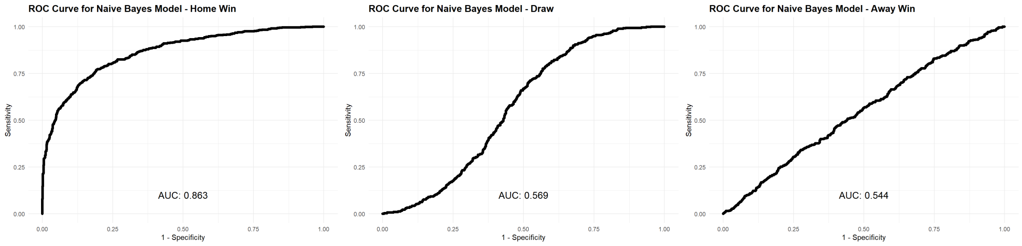

Figure 2.22: Naive Bayes ROC Curves

Chapter 3

Conclusion

Here is the comparison of accuracy, specificity and sensitivity for different models.

Model	Class	Accuracy	Sensitivity	Specificity
Full Model	Away Win	0.7452559	0.7978973	0.7226593
Full Model	Draw	0.7297297	0.8141779	0.7023899
Full Model	Home Win	0.7414223	0.7943578	0.7323039
Full Interaction Model	Away Win	0.7379720	0.8089458	0.7084885
Full Interaction Model	Draw	0.7452559	0.7978973	0.7226593
Full Interaction Model	Home Win	0.7297297	0.8141779	0.7023899
Cleaned Model	Away Win	0.7414223	0.7943578	0.7323039
Cleaned Model	Draw	0.7379720	0.8089458	0.7084885
Cleaned Model	Home Win	0.7452559	0.7978973	0.7226593
Cleaned Interaction Model	Away Win	0.7297297	0.8141779	0.7023899
Cleaned Interaction Model	Draw	0.7414223	0.7943578	0.7323039
Cleaned Interaction Model	Home Win	0.7379720	0.8089458	0.7084885
Stepwise Model	Away Win	0.6837263	0.7207392	0.8817891
Stepwise Model	Draw	0.6837263	0.3646789	0.8733691
Stepwise Model	Home Win	0.6837263	0.8321078	0.7432286
LASSO Model	Away Win	0.6797010	0.7412731	0.8546326
LASSO Model	Draw	0.6797010	0.3394495	0.8902533
LASSO Model	Home Win	0.6797010	0.8247549	0.7486457
Ridge Model	Away Win	0.5922944	0.5913758	0.8514377
Ridge Model	Draw	0.5922944	0.1857798	0.8741366
Ridge Model	Home Win	0.5922944	0.8100490	0.6110509
LDA Model	Away Win	0.5859689	0.6324435	0.8242812
LDA Model	Draw	0.5859689	0.3325688	0.7943208
LDA Model	Home Win	0.5859689	0.6936275	0.7486457
Naive Bayes	Away Win	0.6515239	0.6899384	0.8618211
Naive Bayes	Draw	0.6515239	0.4839450	0.7820414
Naive Bayes	Home Win	0.6515239	0.7181373	0.8385699

Table 3.1: Model Performance Metrics for Different Models

The best model in terms of accuracy is the **Full Model** for the class "Away Win," the **Full Interaction Model** for the class "Draw," and the **Cleaned Model** for the class "Home Win." The best model in terms of sensitivity is the **Cleaned Interaction Model** for the class "Away Win," the **Full Model** for the class "Draw," and the **Stepwise Model** for the class "Home Win." The best model in terms of specificity is the **Stepwise Model** for the class "Away Win," the **LASSO Model** for the class "Draw," and the **Naive Bayes** for the class "Home Win."

However, recalling the unbalanced problem, our choice of the top three overall models is as follows:

- **Full Model:** This model demonstrates strong performance across all classes, particularly in accuracy and sensitivity for the Away Win and Home Win classes.

- **Full Interaction Model:** This model also performs well, especially for the Draw class, indicating its effectiveness in capturing interactions between predictors.

- **Cleaned Model:** This model shows competitive performance, particularly for the Home Win class, suggesting that it effectively balances complexity and interpretability.