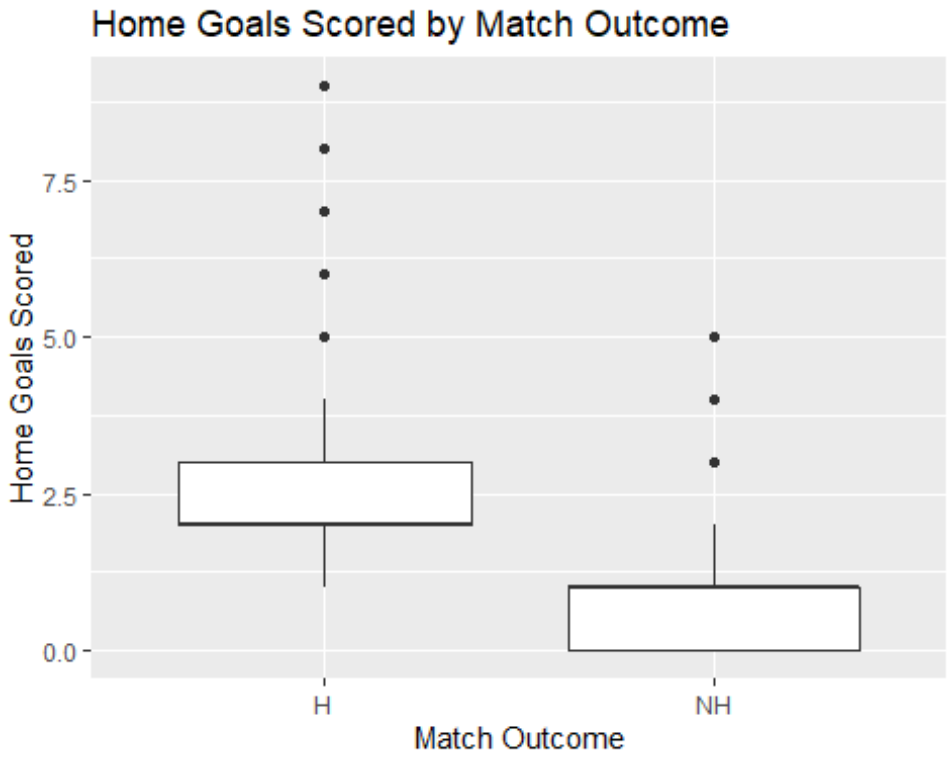
title: “Comprehensive Football Match Analysis”
output:
word_document: default
html_document: default

Descriptive Analysis

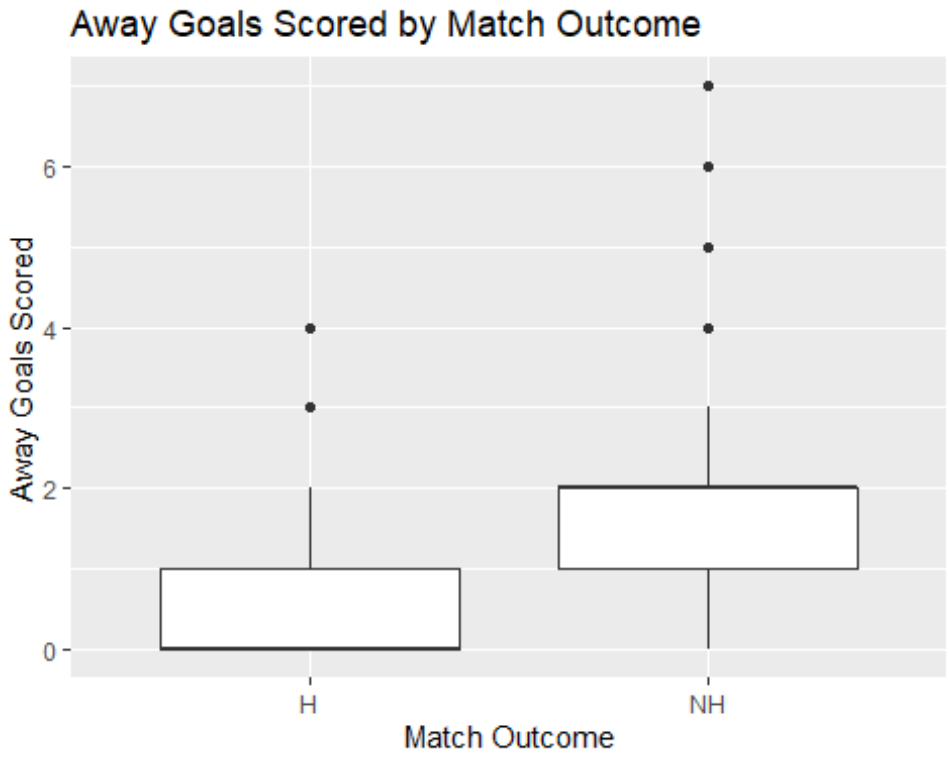
```
library(readr)
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

library(ggplot2)
library(caret)

## Loading required package: lattice

library(randomForest)

## randomForest 4.7-1.1

## Type rfNews() to see new features/changes/bug fixes.

##
## Attaching package: 'randomForest'

## The following object is masked from 'package:ggplot2':
##
##     margin

## The following object is masked from 'package:dplyr':
##
##     combine

library(tidyr)

# Load the dataset
final_dataset <- read_csv("C:/Users/Salman_Imtiaz1/Desktop/EPL.csv")

## New names:
## • `` -> `...1`
```

```
## Rows: 6840 Columns: 40
## ── Column specification
──────────────────────────────────────────────────────────
## Delimiter: ","
## chr (16): Date, HomeTeam, AwayTeam, FTR, HM1, HM2, HM3, HM4, HM5, AM1,
AM2, ...
## dbl (24): ...1, FTHG, FTAG, HTGS, ATGS, HTGC, ATGC, HTP, ATP, MW,
HTFormPts,...
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this
message.

# Basic overview
total_matches <- nrow(final_dataset)
total_goals_scored <- sum(final_dataset$FTHG) + sum(final_dataset$FTAG)
average_home_goals <- mean(final_dataset$FTHG)
average_away_goals <- mean(final_dataset$FTAG)
match_outcome_distribution <- table(final_dataset$FTR) / total_matches * 100

# Print overview
cat("Total Matches:", total_matches, "\n")

## Total Matches: 6840

cat("Total Goals Scored:", total_goals_scored, "\n")

## Total Goals Scored: 18179

cat("Average Home Goals:", average_home_goals, "\n")

## Average Home Goals: 1.527485

cat("Average Away Goals:", average_away_goals, "\n")

## Average Away Goals: 1.130263

print(match_outcome_distribution)

##
##        H       NH
## 46.43275 53.56725

# Correlation between Home/Away Goals Scored and Match Outcome
final_dataset$FTR_binary <- ifelse(final_dataset$FTR == "H", 1, 0)
correlation_fthg_ftr <- cor(final_dataset$FTHG, final_dataset$FTR_binary)
correlation_ftag_ftr <- cor(final_dataset$FTAG, final_dataset$FTR_binary)

# Plotting
ggplot(final_dataset, aes(x = FTR, y = FTHG)) +
  geom_boxplot() +
```

```
  labs(title = "Home Goals Scored by Match Outcome", x = "Match Outcome", y =
"Home Goals Scored")
```




```
ggplot(final_dataset, aes(x = FTR, y = FTAG)) +
  geom_boxplot() +
  labs(title = "Away Goals Scored by Match Outcome", x = "Match Outcome", y =
"Away Goals Scored")
```




```r
# Prepare the data
features <- c("HTGS", "ATGS", "HTGC", "ATGC", "HTGD", "ATGD", "DiffPts",
"DiffFormPts")
target <- "FTR_binary"

# Ensure there are no missing values in the dataset
final_dataset <- final_dataset %>% drop_na()

# Create training and testing sets
set.seed(123)
training_samples <- createDataPartition(final_dataset[[target]], p = 0.8,
list = FALSE)
train_data <- final_dataset[training_samples, ]
test_data <- final_dataset[-training_samples, ]

# Train the Random Forest model using the formula interface
model <- randomForest(as.formula(paste(target, "~", paste(features, collapse
= "+"))), data = train_data)

## Warning in randomForest.default(m, y, ...): The response has five or fewer
## unique values.  Are you sure you want to do regression?

# Predict on the test set
predictions <- predict(model, test_data)

# Calculate accuracy
```

```r
accuracy <- sum(predictions == test_data[[target]]) / nrow(test_data)
print(paste("Accuracy:", accuracy))
```

```
## [1] "Accuracy: 0"
```

```r
model <- randomForest(as.formula(paste(target, "~", paste(features, collapse
= "+"))), data = train_data)
```

```
## Warning in randomForest.default(m, y, ...): The response has five or fewer
## unique values.  Are you sure you want to do regression?
```

```r
final_dataset <- final_dataset %>% drop_na()
```