Demand of Football in European Leagues

DATA 450 Capstone

Matthew Wilcox

1 Introduction

Football (Soccer) is the most populat sport in the world. With an estimated 3.5 billion fans world wide it is the most viewed sport in the word [https://sportforbusiness.com/the-worlds-most-watched-sports/]. Being such a popular sport, there is a large business and economy surrounding the sport. Football teams need to be profitable to succeed. These teams have 5 primary revenue sources. They are television money, prize money, player transfers, sponsorships, and matchday revenues[https://www.football-stadiums.co.uk/articles/how-do-football-clubs-make-money/]. Out of all of these one of the most universial is matchday revenues. Television, prize money, player transfers, and sponsorships can all vary based on what level the team is at. Matchday revenues are much more applicable at all levels of the game. Match day revenues are the profits a team makes from people attending the game. This is from ticket, consessions, and merchandise sales from attending a game. For many teams this is the lifeblood of the club and what allows for the club to survie.

The most important factor within match day revenues is the attendnace. The amount of people that will attend a given match will greatly affect the match day revenues. So understanding factors and being able to predict the attendance for a given match is so important. If a club was able to predict the amount of people that will attend a given match, they could be better prepared for an individual match. Additionally if a predicted match was predicted to have lower attendnace then desired from the club, the club could market it differently or have special promotions in an attempt to increase the attendance for that match.

What has been done here is an evaluation of factors that may impact the attendance for an individual match. These factors are the day/time of the match, betting odds for a match, and whom the away team is for any given match. Additionally, in the end a random forest model was produced to predict attendnace of matches based on these factors.

```
import seaborn as sns
from sklearn.ensemble import RandomForestRegressor
import numpy as np
```

```
import matplotlib.pyplot as plt
import pandas as pd
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
from sklearn import metrics
from sklearn.linear_model import LinearRegression
import july
from datetime import datetime as dt
from jupyter_dash import JupyterDash
from dash import html, dcc, Input, Output
import plotly.graph_objects as go
import plotly.express as px
```

2 Data Collection

The data used in this project was collected from two sources: worldfootball.net and Football-data.co.uk. The data that was collected from worldfootball.net was information on the match such as the names of the teams, the time and date of the match, and most importantly the attendnace for an individual match. This data was not already tabulated so it was scraped from the website. This scrape occured on January 31st 2023. The data that was collected from Football-data.co.uk was primarily betting information for each game. This data was already tabulated into csv files however they were divided based upon the year and league. All the files were downloaded on Febuary 3rd 2023.

The data that was collected was between 2010 and 2023. It consisted of leagues from 11 countries being England, Scotland, Germany, Italy, Spain, France, Netherlands, Belgium, Portugal, Turkey, and Greece. From these Countries 21 leagues of data was collected.

3 Data Processing

All of the csv file's from Football-data.co.uk were combined togther resulting in two datasets. One with all of the betting data and the other with the attendance. These two datasets were to be combined on the home team name, away team name, and the date/time of the game. However, the two data sets had differnet naming structures for team names. For example, the team Manchester City in one dataset would be identified as "man_city" and in the other as "manchester_city". With this differenting structures and spellings of team names, a list of team names was created for both datasets. These lists were than put through a script that took a team from one list and compared the characters to values in the other list. This was

starting with one entire team and slowly decreasing its size and observign that through the other team list.

	Itteration	Results
1	man_city	
2	man_cit, an_city	
3	man_ci, an_cit, n_city	
4	man_c, an_ci, n_cit, _city	

This is an example of how it would divide the name of one list up. It would do this until the team list was broken up into single characters. Then It would start with the largest length of a team name and look through the other list for any results, It would then proceed through the remaining potential names. What resulted is a list of potential matching teams with the team with the most similar name at the top. Then I would determine from the suggestion what was the matching team name. Now that the two data sets had a matching naming of home and away teams, the data sets were able to combine. The resulting data set had 79673 rows and 172 columns where each row was an individual match.

However more processing was needed. Although the intial dataset collected data all the way to 2023, the range of the data was filterd to only 2010 to 2019. This was to attribute to the COVID-19 pandemic. During the pandemic attendance basically ceased to occur for matches. Additionally some leagues cancelled the remaining matches for the season. For those reasons the dataset is focused up untill that time.

Certain leagues were removed from the dataset for analysis . The Scotish Divsiion 2 and Division 3 leagues as well as the Ethniki Katigoria, which is the Greek top league, were removed. This is due to them having several missing values for many variables. Some matches from a variety of leagues had missing values for only betting variables. These matches within leagues were used during the analysis of day/time and the impact of the away team, however, were dropped from the dataset for analysis betting data and in the modeling.

Lastly a few new variables were added. The first variable added was the Season the match occured. Although leagues end on different dates on different years the date selected for the season to switch was July 14th. Most leagues conclude in the begining of June and start back in the begining fo August. July is predominantly used for international games. Although there were a couple of matches that occured in July from 2010-2019, July 14th was the only date with zero matches played. So it was used as the cutoff point.

The other variables were the mean and standard deviaiton of the home team for that season and the z-score of that individual match. The mean and standard deviation were just used to create to create the z-score variable. The z-score is the standardization of the mathces attendance in relation to the home team's average attendance for that particular season.

3.1 Final Dataset

The resulting dataset that was used consisted fo 53,224 rowss and 34 columns. Here is a list of the variables used as well as their description:

Variable	Description
home_team	Name of Home team for na individual match
away_team	Name of Away team for an individual match
raw_attendance	Total number of people who attended an individual match
division	The league the match took place in
B365H	Bet365 home team win odds
B365D	Bet 365 draw odds
B365A	Bet365 away team win odds
BWH	Bet&Win home team win odds
BWD	Bet&Win draw odds
BWA	Bet&Win away team odds
WHH	William Hilll Home win Odds
WHD	William Hill Draw odds
WHA	William Hill Away win odds
VCH	VC Bet Home team win odds
VCD	VC bet draw odds
VCA	VC Bet away win odds
BbAv>2.5	Bet Brain Average over 2.5 goals
BbAV<2.5	Bet Brain Average under 2.5 goals
date_time	Date and time of when a match occured
season	The season a match occured
mean_attend	Average home team attednace for that season
std_attend	Standard deviation of the home team attendance that season
Standard_attendance	The z-score of attendance for a match in relation to the home team attednacne that season

```
total_data = pd.read_pickle('../data/final_datasets/data_standardized.pkl')
total_data.head()
```

```
C:\Users\Matthew Wilcox\AppData\Roaming\Python\Python310\site-packages\IPython\core\formatte

In future versions `DataFrame.to_latex` is expected to utilise the base implementation of `St
```

	home_team	away_team	home_score	away_score	date	time	day_
0	tottenham_hotspur	manchester_city	0	0	2010-08-14	12:45	Satu
1	tottenham_hotspur	wigan_athletic	0	1	2010-08-28	15:00	Satu
2	tottenham_hotspur	wolverhampton_wanderers	3	1	2010-09-18	15:00	Satu
3	tottenham_hotspur	everton_fc	1	1	2010-10-23	12:45	Satu
4	tottenham_hotspur	sunderland_afc	1	1	2010-09-11	20:00	Tues

4 Date & Time

The first factor that will be evaluated is the date and time of individual matches. There are multiple attributes to this that will be viewed, from the day of the week, calendar date, and time of the match.

```
time_df = total_data[[
    'date', 'time', 'day_of_week', 'date_time', 'raw_attendance', 'capacity_filled', 'stan
]]


df_grouped_mean = time_df.groupby('day_of_week')['raw_attendance', 'capacity_filled', 'sta
df_grouped_median = time_df.groupby('day_of_week')['raw_attendance', 'capacity_filled', 's

day_categories = ['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday', 'Sun
df_grouped_median['day_of_week'] = pd.Categorical(df_grouped_median['day_of_week'], catego
df_grouped_median.sort_values(by = 'day_of_week', inplace = True)
```

4.1 Day of Week

First in Figure **??**, the average attendance for matches are being viewed by the day of the week. What is significant here is that it appears that wednesday and sunday has the highest average attendance while Tuesday has the lowest. Wednesday being having the highest average attendance is striking. Most would expect games on the weekday games may have struggle to have high attendance

```
sns.barplot(data=df_grouped_median, x = 'day_of_week', y = 'raw_attendance').set(title ='M
plt.xticks(rotation=90)
plt.xlabel('Day of the Week')
plt.ylabel('Median Attendance')
plt.show()
```

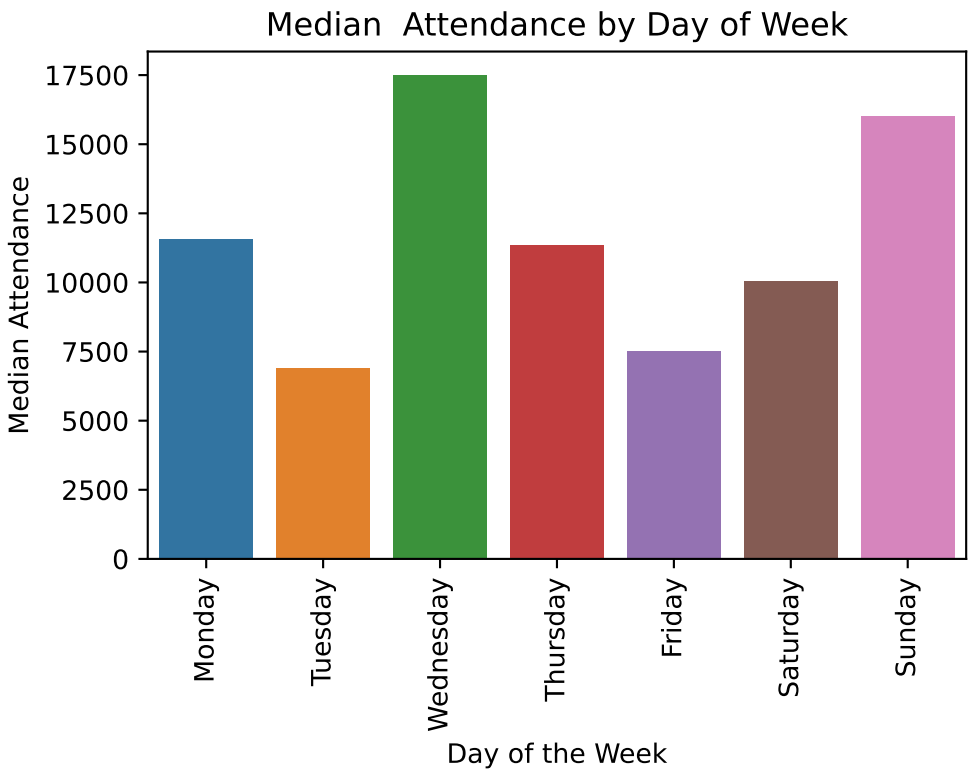

Figure 1: Median Attendnace by day of the week

Figure **??** graph had saturday lower than expected average attendnace with wednesday havign the most games attended on average due to the distribution of number of games seen in Figure **??**. Saturday has by far the most amount of games in comparjision to any day of the week. This results in Saturday having more lower attend games from lower leagues. This is confirmed by Figure **??**. Looking ath the amount of games by leagues on saturday, it had most of the lower leagues hosting matches on Saturday then other days of the week. Wednesday has the second fewest games played however looking again at Figure **??**, it was predominantly composed of the top leagues in England, France, and Italy. The top leagues on average have greater attended games hence Wednesday on average has the greatest attendance on average.

```
grouped_week_count = time_df.groupby('day_of_week').count().reset_index()

grouped_week_count['day_of_week'] = pd.Categorical(grouped_week_count['day_of_week'], cate
grouped_week_count.sort_values(by = 'day_of_week', inplace = True)


sns.barplot(data = grouped_week_count, x = 'day_of_week', y = 'date')
```