Demand of Football in European Leagues

DATA 450 Capstone

Matthew Wilcox

1 Introduction

Football (Soccer) is the most popular sport in the world. With an estimated 3.5 billion fans worldwide, it is the most viewed sport in the world (n.d.a). Football being such a popular sport, there is a large business and economy surrounding the sport. Football teams need to be profitable to succeed. But how do these teams make money? Well mainly from 5 different sources being: television money, prize money, player transfers, sponsorships, and matchday revenues(n.d.b). Out of all of these, one of the most universal is matchday revenues. Television, prize money, player transfers, and sponsorships can vary in significance for a club based on the team's level. Take a team in the England's League two such as AFC Wimbledon. The money they recieve for television drawfs what a team such as Manchester City would recive for television revenue in the English Premier League. Matchday revenues are much more applicable at all levels of the game. Matchday revenues are the profits a team makes from people attending the game. This is from ticket, concessions, and merchandise sales from attending a game. For many teams, this is the lifeblood of the club and what allows the club to survive.

The attendnace of a match significantly affect the match day revenues. So understanding factors and predicting the attendance for a given match is increasingly important. If a club could predict the number of people attending a match, they could be better prepared for an individual match such as if it is expected to have lower attendance than desired by the club, the club could market it differently or have special promotions to increase the attendance for that match.

There are many factors that could impact the attendance of a match. However, the factors used and evaluated here are the day/time of the match, betting odds for a match, and who the away team is for any given match. Additionally, in the end, a random forest model was produced to predict the attendance of matches based on these factors.

2 Python Packages Used

```
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
import pickle
```

3 Data Collection

This project used datat that 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 attendance for an individual match. This data was scraped from the website. This scrape occurred 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 on the year and league. All the files were downloaded on February 3rd, 2023.

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

4 Data Processing

All of the CSV files from Football-data.co.uk were combined together, resulting in two datasets. One with all of the betting data and the other with the attendance. To create one final dataset, the two datasets were to be combined on the home team name, away team name, and the date/time of the game. However, issues arose as they had different 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 these differentiating structures and spellings of team names, a list of team names was created for both datasets. These lists were then put through a python 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 name and slowly decreasing its size and observing 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 split up one team name into a list of smaller strings. 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, looking through the other list for any matching character combinations. 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 initial dataset collected data all the way to 2023, the range of the data was filtered to only 2010 to 2019. This was attributed to the COVID-19 pandemic. During the pandemic, attendance basically ceased to occur for matches. Additionally, some leagues canceled the remaining matches for the season. For those reasons, the dataset is focused up until that time.

Certain leagues were removed from the dataset for analysis. The Scottish Division 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, they were dropped from the dataset for analysis of betting data and in the modeling.

Lastly, a few new variables were added. The first variable added was the season the match occurred. Although leagues end on different dates in different years, the date selected for the

season to switch was July 14th. Most leagues conclude at the beginning of June and start back at the beginning of August. July is predominantly used for international games. Although there were a couple of matches that occurred 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 deviation 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 the z-score variable. The z-score is the standardization of the match's attendance in relation to the home team's average attendance for that particular season.

4.1 Final Dataset

The resulting dataset that was used consisted of 53,224 rows 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

Variable	Description
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

5 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
]]
div_dict = {'D1':'Bundesliga', 'D2': '2. Bundesliga', 'E0':'Premier League', 'E1':'Champio
            'E2':'League 1', 'E3':'Leauge 2','SP1':'La Liga Primera', 'SP2':'La Liga Segun
              'B1':'Jupiler League', 'F1':'Ligue 1','F2':'Ligue 2','I1':'Serie A','I2':'Se
              'SC0':'Scotish Premier League', 'SC1':'Scotish Division 1', 'T1':'Fubol Ligi
divisions_list =['D1', 'D2', 'E0', 'E1', 'E2', 'E3', 'SP1' ,'SP2', 'B1', 'F1', 'F2', 'I1',

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)
```

5.1 Day of Week

First, in Figure **??**, the average attendance for matches is viewed by the day of the week. What is significant here is that it appears that Wednesday and Sunday have the highest average attendance while Tuesday has the lowest. Wednesday having the highest average attendance is striking. Most would expect matches on the weekdaywould 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()
```

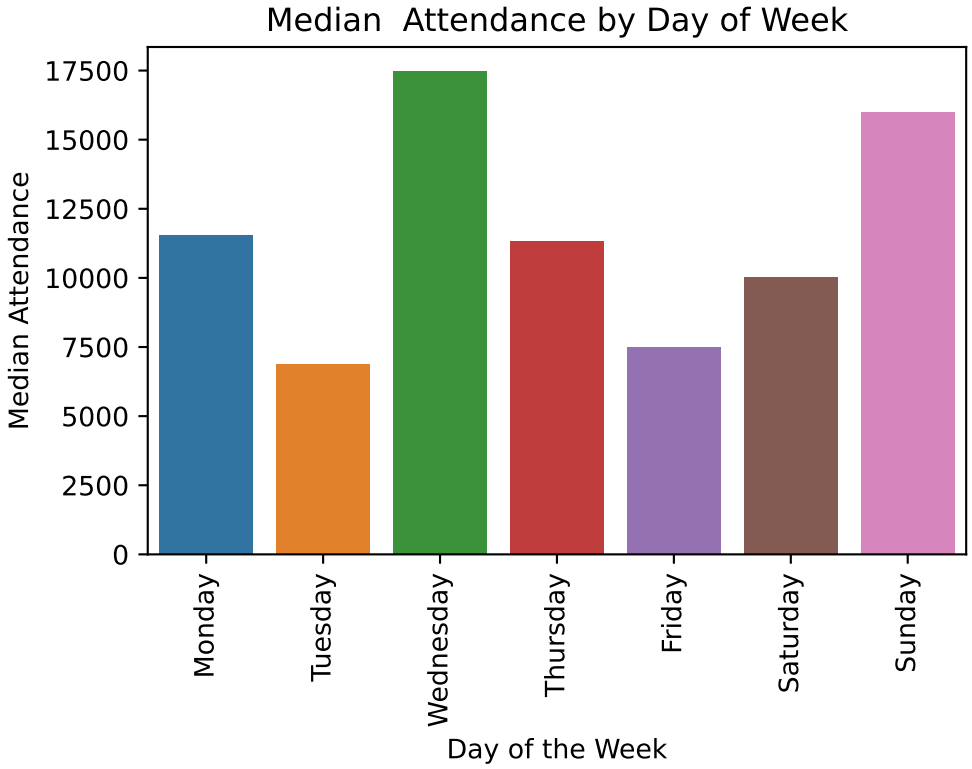

Figure 1: Median Attendnace by day of the week