Soccer Quality by Country

Edmond Leahy

Dataset(s)

For this project, the European Soccer Database dataset was used.

https://www.kaggle.com/hugomathien/soccer

Motivation

In the soccer world, the English Premier League is often said to be the best soccer league to watch in terms of quality. I have heard this all my life, while wondering if this was true. I have watched many soccer games from many countries, and have found that other leagues are just as exciting to watch, if not more exciting. I wanted to find out what the data showed.

Research Question(s)

Is England the most exciting soccer country in the world, in terms of goals per game and goals per country?

Methodology

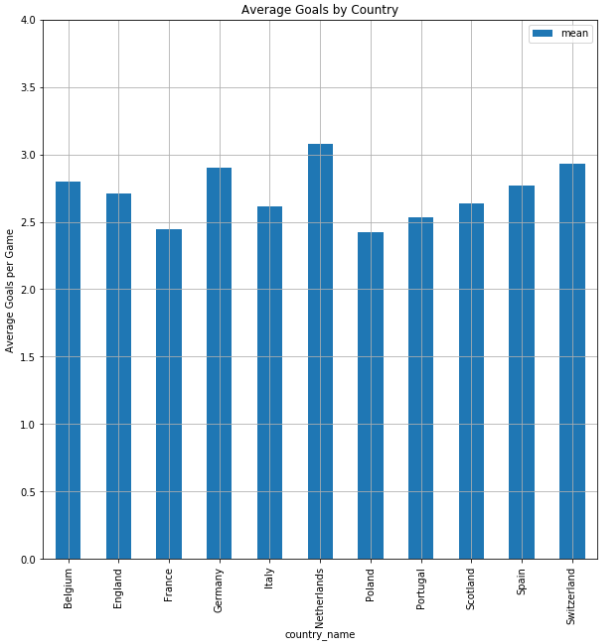
By using a combination of techniques taught in the Python for Data Science course, I was able to manipulate the dataset to show the relevant statistics to the research question.

The following are some of the techniques used:

- SQL querying
- DataFrame Filtering
- DataFrame merging
- Matplotlib graphing

Findings –Goals Per Game(1)

The results of grouping the dataset by country gave the following results for the average number of goas per game :



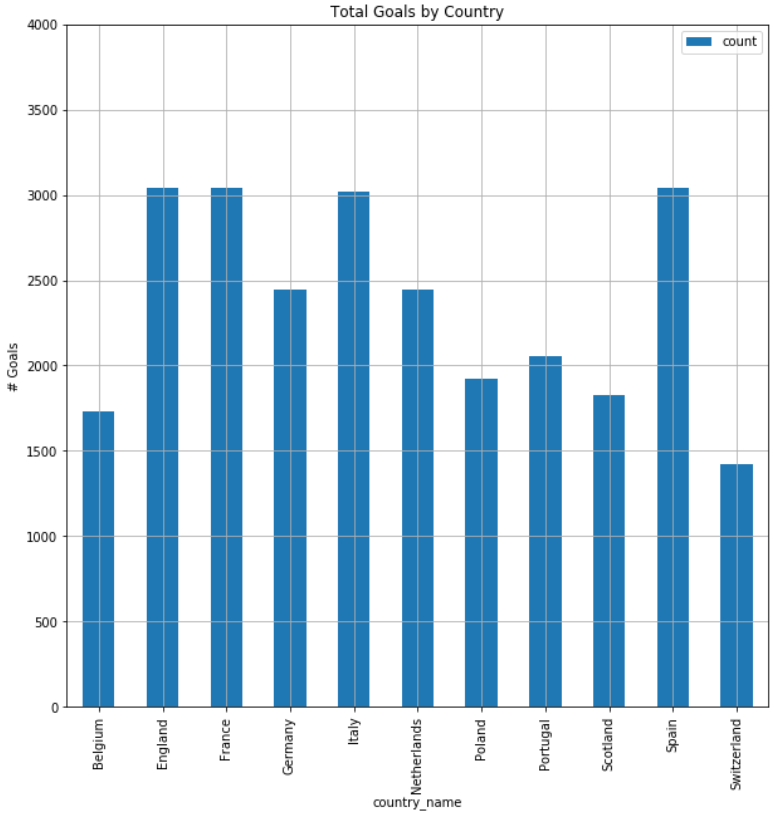
country_name	count	mean	std	min	50%	75%	95%	max
Netherlands	2448.0	3.080882	1.740640	0.0	3.0	4.0	6.0	10.0
Switzerland	1422.0	2.929677	1.717765	0.0	3.0	4.0	6.0	9.0
Germany	2448.0	2.901552	1.704974	0.0	3.0	4.0	6.0	11.0
Belgium	1728.0	2.801505	1.656507	0.0	3.0	4.0	6.0	9.0
Spain	3040.0	2.767105	1.731111	0.0	3.0	4.0	6.0	12.0
England	3040.0	2.710526	1.691127	0.0	3.0	4.0	6.0	10.0
Scotland	1824.0	2.633772	1.644379	0.0	2.0	4.0	6.0	12.0
Italy	3017.0	2.616838	1.640327	0.0	2.0	4.0	6.0	9.0
Portugal	2052.0	2.534600	1.637348	0.0	2.0	4.0	5.0	9.0
France	3040.0	2.443092	1.551799	0.0	2.0	3.0	5.0	10.0
Poland	1920.0	2.425000	1.540355	0.0	2.0	3.0	5.0	8.0

Findings – Goals Per Game(2)

As the plot in the previous slide shows, the top three countries in terms of goal average per game are: Netherlands, Switzerland, Germany. England, however, is listed as sixth.

But looking at the table shown in the previous slide, I notice that the sum of goals scored per country might show a different story. This is explored in the following slide

Findings –Total Goals(1)



country_name	count	mean	std	min	50%	75%	95%	max
England	3040.0	2.710526	1.691127	0.0	3.0	4.0	6.0	10.0
France	3040.0	2.443092	1.551799	0.0	2.0	3.0	5.0	10.0
Spain	3040.0	2.767105	1.731111	0.0	3.0	4.0	6.0	12.0
Italy	3017.0	2.616838	1.640327	0.0	2.0	4.0	6.0	9.0
Germany	2448.0	2.901552	1.704974	0.0	3.0	4.0	6.0	11.0
Netherlands	2448.0	3.080882	1.740640	0.0	3.0	4.0	6.0	10.0
Portugal	2052.0	2.534600	1.637348	0.0	2.0	4.0	5.0	9.0
Poland	1920.0	2.425000	1.540355	0.0	2.0	3.0	5.0	8.0
Scotland	1824.0	2.633772	1.644379	0.0	2.0	4.0	6.0	12.0
Belgium	1728.0	2.801505	1.656507	0.0	3.0	4.0	6.0	9.0
Switzerland	1422.0	2.929677	1.717765	0.0	3.0	4.0	6.0	9.0

Findings –Total Goals(2)

As shown in the previous slide, the top three countries in term of total goals are: England, France, and Spain. All of these countries have the same number of goals.

As a note, Spain had the highest scoring game recorded out of the three, so this might raise their level of excitement too!

Findings - Conclusions

To conclude the results of the findings, it is clear that England is not a stand-alone country in terms of goals per game and overall goals.

The statement that England has the best soccer games would depend heavily on how you define a good soccer game. In terms of average goals per game, England does not have the best soccer games. In terms of total goals, it is tied with France and Spain.

Acknowledgements

This project was limited by the dataset used. Ranking soccer by countries in terms of Goals per game and Total goals is not a complete story of the quality of soccer. Many other factors influence the quality of the game, and many of these factors are subjective. This project was chosen to have a small scope for simplicity.

References

I used the bleacher report article to gain some knowledge on how to rank games:

https://bleacherreport.com/articles/1922780-statistically-ranking-the-worlds-top-10-football-leagues#slide0

I extensively used Edx's Python for Data Science course reference material:

https://courses.edx.org/courses/course-v1:UCSanDiegoX+DSE200x+3T2019/course/

Soccer Quality of Countries

Measured by average goals per game and goals per country

Import the useful packages, and make sure that matplot lib can show in the notebook

In [18]:
```
import sqlite3
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

Create connections to the database, and use SQL to retrieve the desired data.

In [19]:
```
cnx = sqlite3.connect('database.sqlite')
#player_at = pd.read_sql_query("SELECT * FROM Player_Attributes", cnx)
#player = pd.read_sql_query("SELECT * FROM Player", cnx)
#league = pd.read_sql_query("SELECT * FROM League", cnx)
team = pd.read_sql_query("SELECT * FROM Team", cnx)
country= pd.read_sql_query("SELECT * FROM Country", cnx)
match= pd.read_sql_query("SELECT * FROM Match", cnx)
```

rename the columns to be the merge columns, for simplicity

In [20]:
```
country['country_id'] = country['id']
country['country_name'] = country['name']
country = country.drop(columns=['name','id'])
```

Filter the datasets to only the desired data

In [21]:
```
match_filtered = match[['country_id', 'league_id','home_team_api_id', 'away_te
am_api_id', 'home_team_goal', 'away_team_goal']]
team_filtered = team[['team_api_id', 'team_long_name']]
```

Show the table

In [22]:
```
match_filtered.head()
```

Out[22]:

	country_id	league_id	home_team_api_id	away_team_api_id	home_team_goal	away_team_goa
0	1	1	9987	9993	1	
1	1	1	10000	9994	0	
2	1	1	9984	8635	0	
3	1	1	9991	9998	5	
4	1	1	7947	9985	1	

Get country and league information, place them into the desired dataframe (soccer_data)

In [23]:
```
soccer_data = match_filtered.merge(country, on='country_id', how='inner')
soccer_data = soccer_data.merge(league, left_on='league_id', right_on='id', ho
w='inner')
soccer_data = soccer_data.merge(team_filtered, left_on='home_team_api_id', rig
ht_on='team_api_id', how='inner')
soccer_data = soccer_data.rename(columns={"team_long_name": "home_team"})
soccer_data = soccer_data.merge(team_filtered, left_on='away_team_api_id', rig
ht_on='team_api_id', how='inner')
soccer_data = soccer_data.rename(columns={"team_long_name": "away_team", "nam
e":"league_name"})
soccer_data = soccer_data[['country_name', 'league_name', 'home_team', 'home_t
eam_goal', 'away_team', 'away_team_goal']]
```

Fill in the 'Total Goals' per game by adding the home and away goals

In [24]:
```
soccer_data['total_goals'] = soccer_data['home_team_goal'] + soccer_data['away
_team_goal']
soccer_data.head()
```

Out[24]:

	country_name	league_name	home_team	home_team_goal	away_team	away_team_goal	tota
0	Belgium	Belgium Jupiler League	KRC Genk	1	Beerschot AC	1	
1	Belgium	Belgium Jupiler League	KRC Genk	1	Beerschot AC	1	
2	Belgium	Belgium Jupiler League	KRC Genk	2	Beerschot AC	1	
3	Belgium	Belgium Jupiler League	KRC Genk	3	Beerschot AC	1	
4	Belgium	Belgium Jupiler League	KRC Genk	3	Beerschot AC	0	

Time to plot!

First, group the data by country_name

Next, show the description of the data, sorted by highest mean first

In [25]:
```
country_data = soccer_data.groupby('country_name')
country_total_goals = country_data['total_goals'].describe(percentiles=[.95,.75])
country_total_goals.sort_values(by=['mean'], ascending=False)
```

Out[25]:

country_name	count	mean	std	min	50%	75%	95%	max
Netherlands	2448.0	3.080882	1.740640	0.0	3.0	4.0	6.0	10.0
Switzerland	1422.0	2.929677	1.717765	0.0	3.0	4.0	6.0	9.0
Germany	2448.0	2.901552	1.704974	0.0	3.0	4.0	6.0	11.0
Belgium	1728.0	2.801505	1.656507	0.0	3.0	4.0	6.0	9.0
Spain	3040.0	2.767105	1.731111	0.0	3.0	4.0	6.0	12.0
England	3040.0	2.710526	1.691127	0.0	3.0	4.0	6.0	10.0
Scotland	1824.0	2.633772	1.644379	0.0	2.0	4.0	6.0	12.0
Italy	3017.0	2.616838	1.640327	0.0	2.0	4.0	6.0	9.0
Portugal	2052.0	2.534600	1.637348	0.0	2.0	4.0	5.0	9.0
France	3040.0	2.443092	1.551799	0.0	2.0	3.0	5.0	10.0
Poland	1920.0	2.425000	1.540355	0.0	2.0	3.0	5.0	8.0

Show the table, this time being sorted by highest count first

In [26]:
```
country_total_goals.sort_values(by=['count'], ascending=False)
```

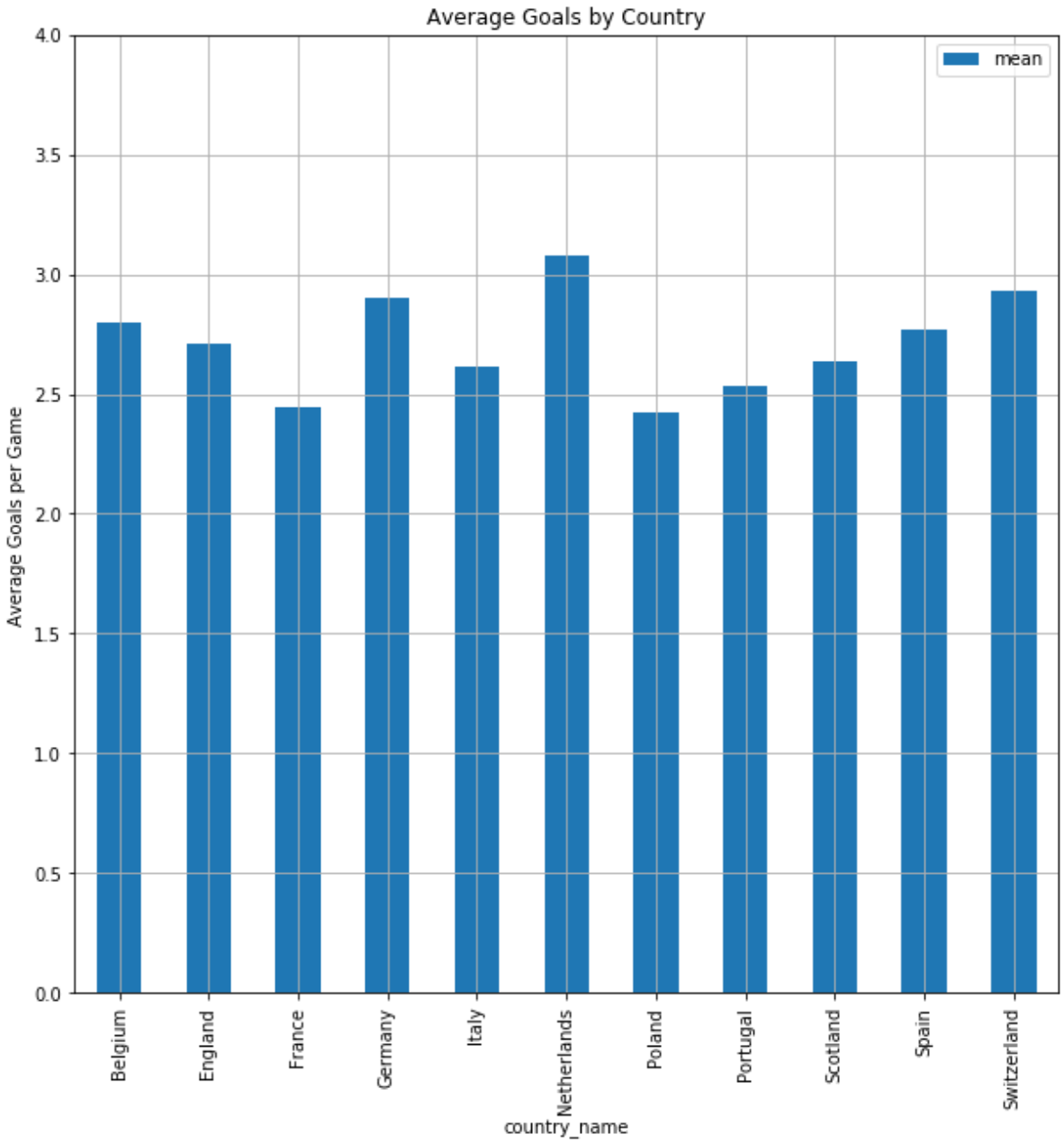
Out[26]:

country_name	count	mean	std	min	50%	75%	95%	max
England	3040.0	2.710526	1.691127	0.0	3.0	4.0	6.0	10.0
France	3040.0	2.443092	1.551799	0.0	2.0	3.0	5.0	10.0
Spain	3040.0	2.767105	1.731111	0.0	3.0	4.0	6.0	12.0
Italy	3017.0	2.616838	1.640327	0.0	2.0	4.0	6.0	9.0
Germany	2448.0	2.901552	1.704974	0.0	3.0	4.0	6.0	11.0
Netherlands	2448.0	3.080882	1.740640	0.0	3.0	4.0	6.0	10.0
Portugal	2052.0	2.534600	1.637348	0.0	2.0	4.0	5.0	9.0
Poland	1920.0	2.425000	1.540355	0.0	2.0	3.0	5.0	8.0
Scotland	1824.0	2.633772	1.644379	0.0	2.0	4.0	6.0	12.0
Belgium	1728.0	2.801505	1.656507	0.0	3.0	4.0	6.0	9.0
Switzerland	1422.0	2.929677	1.717765	0.0	3.0	4.0	6.0	9.0

Plot the Average Goals by Country

In [27]:
```
ax = country_total_goals.plot.bar(y='mean', figsize=(10,10))
ax.grid()
ax.set_ylabel('Average Goals per Game')
ax.set_title('Average Goals by Country')
ax.set_ylim(0,4)
```

Out[27]: (0, 4)



Plot the Total Goals by Country

In [28]:
```
ax = country_total_goals.plot.bar(y='count', figsize=(10,10))
ax.grid()
ax.set_ylabel('# Goals')
ax.set_title('Total Goals by Country')
ax.set_ylim(0,4000)
```

Out[28]: (0, 4000)