ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

KHOA TOÁN - TIN HỌC

…. ✍….

BÁO CÁO KỸ THUẬT

Đề tài: Dự đoán kết quả World Cup 2022 và so sánh nó với kết quả thực tế, rút ra nhận xét

Giảng viên hướng dẫn: Hà Văn Thảo

Lớp: 20KDL1

Thành viên nhóm:

19110139 – Nguyễn Song Nhật

TP. Hồ Chí Minh, tháng 1 năm 2023

MỤC LỤC

LỜI NÓI ĐẦU

Với sự phát triển nhanh công nghệ hiện nay, cùng với công cụ toán học, các dự báo về thời tiết, về thiên tai,... đang len lỏi trong cuộc sống hằng ngày của chúng ta. Dĩ nhiên chúng ta cũng không thể quên đi một trong những môn thể thao giải trí lớn nhất hành tinh - bóng đá, với lượng fan hâm mộ đông đảo trên khắp cả thế giới hứa hẹn là một bữa ăn thịnh soạn cho những nhà đầu tư, và dĩ nhiên là cả những công ty chuyên dự báo kết quả, đặt kèo, và cá cược bóng đá.

Ngày trước, vào thập niên những 40 - 50, người ta chỉ có thể coi tổng quan một trận đấu và rút ra kết luận một đội mạnh hay yếu dựa trên những kết quả rất chủ quan, hoặc với dữ liệu đầu vào rất ít. Máy tính thời đó cũng chưa phát triển đủ nhanh, đủ mạnh, và đủ dung lượng để lưu trữ một database lớn, nhằm đưa ra kết quả chính xác nhất.

Tuy nhiên bước vào thế kỉ 21 với sự phát triển chóng mặt của công nghệ bán dẫn, người ta đã tìm ra được cách để thu nhỏ kích thước vi xử lý, bộ nhớ, ram, vv nhưng lại có nhiều không gian hơn, chưa được số lượng khổng lồ hơn, và từ đó việc dự đoán kết quả, cá cược các trận đấu dần nở rộ, và trở thành nơi hái ra tiền không chỉ ở riêng mỗi bóng đá mà còn trải dài ở rất nhiều những môn thể thao khác.

Bóng đá đơn giản là ông vua kiếm tiền trong tất cả các môn thể thao, và với sự trợ giúp của công nghệ hiện nay, người ta bắt đầu có những dự đoán chính xác hơn về kết quả xảy ra trong tương lai với sai số thấp hơn, nhờ vào khả năng lưu trữ khổng lồ của máy tính hiện nay. Với hàng triệu các tệp dữ liệu trận đấu khác nhau.

Mặc dù World Cup 2022 đã kết thúc cách đây được 2 tuần, nhưng trong báo cáo kì này, tôi vẫn quyết định xây dựng một model đơn giản nhằm dự đoán kết quả. Với sự trợ giúp của dữ liệu từ wikipedia. Chúng ta hãy cùng xem thử, liệu sai sót giữa kết quả của kì World Cup 2022 đã diễn ra, và đang được dự đoán trong bài báo cáo này sẽ khác biệt ở đâu, điều gì làm nên sự khác biệt đó, và đâu là yếu tố khác nhau giữa đời thực và các con số dữ liệu trên máy tính.

Dự đoán kết quả World Cup 2022

Bước 1: Đọc hiểu dữ liệu

Bắt đầu bằng việc tải dữ liệu.

```
In [3]: import pandas as pd
```

Data Cleaning

```
In [4]: df_historical_data = pd.read_csv('data/fifa_worldcup_matches.csv')
        df_fixture = pd.read_csv('data/fifa_worldcup_fixture.csv')
        df_missing_data = pd.read_csv('data/fifa_worldcup_missing_data.csv')
```

```
In [5]: df_fixture
Out[5]:
```

	home	score	away	year
0	Qatar	Match 1	Ecuador	2022
1	Senegal	Match 2	Netherlands	2022
2	Qatar	Match 18	Senegal	2022
3	Netherlands	Match 19	Ecuador	2022
4	Ecuador	Match 35	Senegal	2022
...	...	...	...	...
59	Winners Match 51	Match 59	Winners Match 52	2022
60	Winners Match 57	Match 61	Winners Match 58	2022
61	Winners Match 59	Match 62	Winners Match 60	2022
62	Losers Match 61	Match 63	Losers Match 62	2022
63	Winners Match 61	Match 64	Winners Match 62	2022

Bước 2: Xử lí dữ liệu

+ df-fixture

```
In [7]: df_fixture['home'] = df_fixture['home'].str.strip()
        df_fixture['away'] = df_fixture['away'].str.strip()

Out[7]: 0               Qatar
        1             Senegal
        2               Qatar
        3         Netherlands
        4             Ecuador
                     ...
        59    Winners Match 51
        60    Winners Match 57
        61    Winners Match 59
        62     Losers Match 61
        63    Winners Match 61
        Name: home, Length: 64, dtype: object
```

Dễ thấy có khoảng trắng trong dữ liệu, ta dùng hàm strip() để xóa khoảng trắng

+ df_missing_data

```
In [10]: # null data
         df_missing_data[df_missing_data['home'].isnull()]

         # drop null data

         # concatena dfs and clean
```

Out[10]:

	home	score	away	year
396	NaN	NaN	NaN	2010
397	NaN	NaN	NaN	2010
398	NaN	NaN	NaN	2010
399	NaN	NaN	NaN	2010
400	NaN	NaN	NaN	2010
...	...	...	...	...
455	NaN	NaN	NaN	2010
456	NaN	NaN	NaN	2010
457	NaN	NaN	NaN	2010
458	NaN	NaN	NaN	2010

Nhìn vào bảng dữ liệu csv, dễ thấy missing data có rất nhiều null. Ta dùng lệnh isnull để thể hiện toàn bộ các dữ liệu có giá trị null. Trong đó có đến 64 hàng bị mất dữ liệu.

Qua đó chúng ta bắt buộc phải sử dụng đến phương pháp Drop Null data với cú pháp drop na(). với inplace = True để nó tự động thay thế và cập nhật dữ liệu.

```
In [18]: df_missing_data[df_missing_data['home'].isnull()]
         df_missing_data.dropna(inplace=True)
         df_historical_data = pd.concat([df_historical_data, df_missing_data], ignore_index=True)
         df_historical_data.drop_duplicates(inplace=True)
         df_historical_data.sort_values('year', inplace=True)
         df_historical_data
```

Out[18]:

	home	score	away	year
0	France	4–1	Mexico	1930
17	Romania	3–1	Peru	1930
16	Uruguay	4–2	Argentina	1930
15	Uruguay	6–1	Yugoslavia	1930
14	Argentina	6–1	United States	1930
...	...	...	...	...
863	Brazil	2–0	Costa Rica	2018
864	Serbia	1–2	Switzerland	2018
865	Sweden	1–0	South Korea	2018
867	France	4–2	Croatia	2018
900	Brazil	1–2	Belgium	2018

901 rows × 4 columns

Sau đó chúng ta hợp nhất hai bảng dữ liệu dfs và clean bằng concat. Cả hai bảng historical và missing đều bắt đầu từ index =1 với cú pháp ignore này, ta sẽ để bảng concat có index chạy từ 0. Trong bảng missing data sẽ có những dữ liệu bị trùng với matches, sử dụng drop duplicate, sau đó sắp xếp theo năm diễn ra.

```
In [14]: df_historical_data[df_historical_data['home'].str.contains('Sweden') &
                            df_historical_data['away'].str.contains('Austria')].index
```

Out[14]:

	home	score	away	year
37	Sweden	w/o[a]	Austria	1938

Trong bảng historical_data, nhận thấy có một dữ liệu bị trống, w/0 là viết tắt của walk over, tức là trận này đã bị xử thua 0 - 3 với đội thua, do 1 trong hai đội đã không tham gia trận đấu. Đội không tham gia bị xử thua.

```
# Xóa dữ liệu những trận bị xử thua 0-3 do đội không thi đấu
delete_index = df_historical_data[df_historical_data['home'].str.contains('Sweden') &
                                  df_historical_data['away'].str.contains('Austria')].index

df_historical_data.drop(index=delete_index, inplace=True)

# căn chỉnh lại khoảng trắng cho các cột score, home, way tương tự ở trên với hàm str strip()
df_historical_data['score'] = df_historical_data['score'].str.replace('[^\d-]', '', regex=True)
df_historical_data['home'] = df_historical_data['home'].str.strip()
df_historical_data['away'] = df_historical_data['away'].str.strip()

# tách cột tỷ số thành bàn thắng sân nhà và sân khách và bỏ cột tỷ số
df_historical_data[['HomeGoals', 'AwayGoals']] = df_historical_data['score'].str.split('-', expand=True)
df_historical_data.drop('score', axis=1, inplace=True)

# Đổi tên các cột, và thay đổi kiểu dữ liệu của chúng
df_historical_data.rename(columns={'home': 'HomeTeam', 'away': 'AwayTeam',
                                   'year':'Year'}, inplace=True)
df_historical_data = df_historical_data.astype({'HomeGoals': int, 'AwayGoals':int, 'Year': int})

# Thêm một cột totalgoals = home goals + away goals
df_historical_data['TotalGoals'] = df_historical_data['HomeGoals'] + df_historical_data['AwayGoals']
df_historical_data
```

	HomeTeam	AwayTeam	Year	HomeGoals	AwayGoals	TotalGoals
0	France	Mexico	1930	4	1	5
17	Uruguay	Argentina	1930	4	2	6
16	Uruguay	Yugoslavia	1930	6	1	7
15	Argentina	United States	1930	6	1	7
14	Paraguay	Belgium	1930	1	0	1
...	...	...	...	...	...	...
419	Brazil	Costa Rica	2018	2	0	2
420	Serbia	Switzerland	2018	1	2	3
421	Serbia	Brazil	2018	0	2	2
408	France	Peru	2018	1	0	1
450	Brazil	Belgium	2018	1	2	3

900 rows × 6 columns

Sau đó, ta sẽ xóa luôn những trận bị xử thua như vậy, đối với trường hợp này thì chỉ có một trận của Thụy Điển diễn ra vào 1938. Sau đó ta tiếp tục làm đẹp dữ liệu, dùng hàm str() như trên để xóa khoảng trắng, thay đổi tên của các cột trực quan hơn, tách cột tỷ số, thành hai cột và thêm cột Total Goals.

Đối với cột score sử dụng hàm replace để thay toàn bộ những kí tự lạ thành khoảng trắng, sau đó set regex = true để thay thế khoảng trắng vào trong dataframe khi sử dụng pandas.

	home	score	away	year
34	Italy	2–1 (a.e.t.)	Czechoslovakia	1934
27	Italy	1–1 (a.e.t.)	Spain	1934
24	Austria	3–2 (a.e.t.)	France	1934
48	Brazil	1–1 (a.e.t.)	Czechoslovakia	1938
42	Czechoslovakia	3–0 (a.e.t.)	Netherlands	1938
...	...	...	...	...
443	Spain	1–1 (a.e.t.)	Russia	2018
444	Croatia	1–1 (a.e.t.)	Denmark	2018
448	Colombia	1–1 (a.e.t.)	England	2018
452	Russia	2–2 (a.e.t.)	Croatia	2018
454	Croatia	2–1 (a.e.t.)	England	2018

```
In [8]: df_historical_data.to_csv('clean_fifa_worldcup_matches.csv',index=False)
        df_fixture.to_csv('clean_fifa_worldcup_fixture.csv',index=False)
```

Sau đó ta xuất các file đã được làm sạch thành một file khác có clean là tiền tố đầu tiên.

Ta kiểm tra thử xem file clean data đã thực sự sạch chưa, bằng cách chọn trường hợp xấu nhất đã xử lý là Sweden (Thụy điển) bị w/0 ở cột tỷ số.

```
: # verify data collected for a team
  print(df_historical_data[df_historical_data['HomeTeam'].str.contains('Sweden')])
  print(df_historical_data[df_historical_data['AwayTeam'].str.contains('Sweden')])
```

	HomeTeam	AwayTeam	Year	HomeGoals	AwayGoals	TotalGoals
23	Sweden	Argentina	1934	3	2	5
46	Sweden	Cuba	1938	8	0	8
66	Sweden	Italy	1950	3	2	5
67	Sweden	Paraguay	1950	2	2	4
74	Sweden	Spain	1950	3	1	4
134	Sweden	West Germany	1958	3	1	4
131	Sweden	Soviet Union	1958	2	0	2
119	Sweden	Wales	1958	0	0	0
118	Sweden	Hungary	1958	2	1	3
115	Sweden	Mexico	1958	3	0	3
466	Sweden	Israel	1970	1	1	2
498	Sweden	Uruguay	1974	3	0	3
494	Sweden	Bulgaria	1974	0	0	0
512	Sweden	Poland	1974	0	1	1
516	Sweden	Yugoslavia	1974	2	1	3
654	Sweden	Costa Rica	1990	1	2	3
652	Sweden	Scotland	1990	1	2	3
263	Sweden	Bulgaria	1994	4	0	4
262	Sweden	Brazil	1994	0	1	1
682	Sweden	Russia	1994	3	1	4
283	Sweden	Senegal	2002	1	2	3
789	Sweden	Nigeria	2002	2	1	3
791	Sweden	Argentina	2002	1	1	2
814	Sweden	Paraguay	2006	1	0	1
815	Sweden	England	2006	2	2	4
447	Sweden	Switzerland	2018	1	0	1
451	Sweden	England	2018	0	2	2
424	Sweden	South Korea	2018	1	0	1

	HomeTeam	AwayTeam	Year	HomeGoals	AwayGoals	TotalGoals
28	Germany	Sweden	1934	2	1	3
52	Brazil	Sweden	1938	4	2	6
50	Hungary	Sweden	1938	5	1	6
73	Uruguay	Sweden	1950	3	2	5
71	Brazil	Sweden	1950	7	1	8
136	Brazil	Sweden	1958	5	2	7
467	Uruguay	Sweden	1970	0	1	1
464	Italy	Sweden	1970	1	0	1
496	Netherlands	Sweden	1974	0	0	0
514	West Germany	Sweden	1974	4	2	6
533	Spain	Sweden	1978	1	0	1
531	Austria	Sweden	1978	1	0	1
530	Brazil	Sweden	1978	1	1	2
649	Brazil	Sweden	1990	2	1	3
260	Romania	Sweden	1994	2	2	4
251	Saudi Arabia	Sweden	1994	1	3	4
679	Cameroon	Sweden	1994	2	2	4
684	Brazil	Sweden	1994	1	1	2
788	England	Sweden	2002	1	1	2
297	Germany	Sweden	2006	2	0	2
812	Trinidad and Tobago	Sweden	2006	0	0	0
426	Germany	Sweden	2018	2	1	3
428	Mexico	Sweden	2018	0	3	3

Không có cột nào bị w/0 cũng như các hàng các cột đã được thay đổi, xóa khoảng trắng. Thành công.

```
# đọc file
df = pd.read_csv('data/players_22.csv', low_memory=False)

# khai tên cho các cột trong dữ liệu
df = df[['short_name', 'age', 'nationality_name', 'overall', 'potential',
         'club_name', 'value_eur', 'wage_eur', 'player_positions']]

# chọn lại chỉ duy nhất 1 vị trí của cầu thủ đó
df['player_positions'] = df['player_positions'].str.split(',', expand=True)[0]

# loại những cầu thủ có giá trị null
df.dropna(inplace=True)
```

```
players_missing_worldcup = ['K. Benzema', 'S. Mané', 'S. Agüero', 'Sergio Ramos', 'P. Pogba',
                            'M. Reus', 'Diogo Jota', 'A. Harit', 'N. Kanté', 'G. Lo Celso', 'Piqué']

# loại những cầu thủ bị chấn thương
drop_index = df[df['short_name'].isin(players_missing_worldcup)].index
df.drop(drop_index, axis=0, inplace=True)
```

```
teams_worldcup = [
    'Qatar', 'Brazil', 'Belgium', 'France', 'Argentina', 'England', 'Spain', 'Portugal',
    'Mexico', 'Netherlands', 'Denmark', 'Germany', 'Uruguay', 'Switzerland', 'United States', 'Croatia',
    'Senegal', 'Iran', 'Japan', 'Morocco', 'Serbia', 'Poland', 'South Korea', 'Tunisia',
    'Cameroon', 'Canada', 'Ecuador', 'Saudi Arabia', 'Ghana', 'Wales', 'Costa Rica', 'Australia'
]

# Sap xep chi co nhung doi tuyen quoc gia duoc tham du World Cup
df = df[df['nationality_name'].isin(teams_worldcup)]
```

```
# Cầu thủ xuất sắc nhất được sắp xếp theo 3 tiêu chí overall, potenial và value theo tỉ giá Euro.
df.sort_values(by=['overall', 'potential', 'value_eur'], ascending=False, inplace=True)
```

Ta tiếp tục làm sạch dữ liệu ở bảng players, để phân tích ở bước 3

Hình dạng của dict_table

```
import pandas as pd
import pickle
from scipy.stats import poisson
```

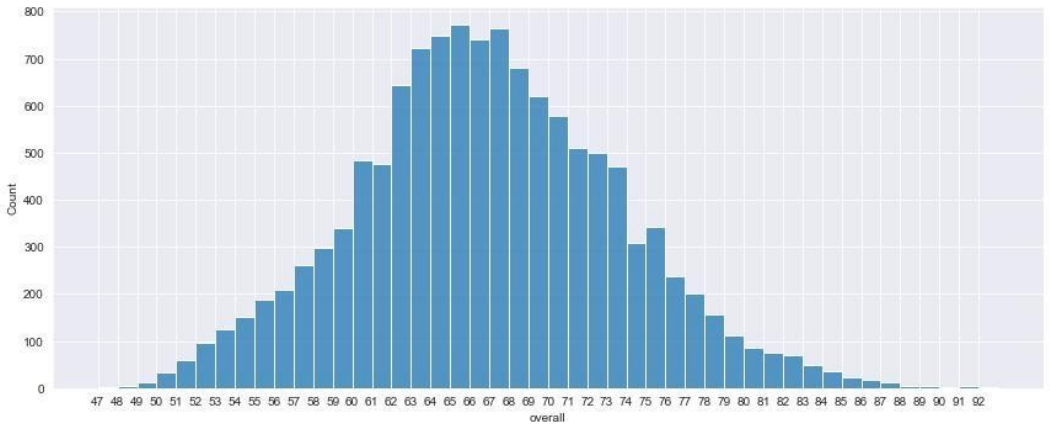
```
dict_table = pickle.load(open('data/dict_table','rb'))
df_historical_data = pd.read_csv('data/clean_fifa_worldcup_matches.csv')
df_fixture = pd.read_csv('data/clean_fifa_worldcup_fixture.csv')
```

```
dict_table
```

```
{'Group A':    Pos         Team  Pld  W  D  L  GF  GA  GD  Pts
 0    1    Qatar (H)    0  0  0  0   0   0   0    0
 1    2      Ecuador    0  0  0  0   0   0   0    0
 2    3      Senegal    0  0  0  0   0   0   0    0
 3    4  Netherlands    0  0  0  0   0   0   0    0,
 'Group B':    Pos           Team  Pld  W  D  L  GF  GA  GD  Pts
 0    1        England    0  0  0  0   0   0   0    0
 1    2           Iran    0  0  0  0   0   0   0    0
 2    3  United States    0  0  0  0   0   0   0    0
 3    4          Wales    0  0  0  0   0   0   0    0,
 'Group C':    Pos          Team  Pld  W  D  L  GF  GA  GD  Pts
 0    1     Argentina    0  0  0  0   0   0   0    0
 1    2  Saudi Arabia    0  0  0  0   0   0   0    0
 2    3        Mexico    0  0  0  0   0   0   0    0
 3    4        Poland    0  0  0  0   0   0   0    0,
 'Group D':    Pos       Team  Pld  W  D  L  GF  GA  GD  Pts
 0    1     France    0  0  0  0   0   0   0    0
 1    2  Australia    0  0  0  0   0   0   0    0
 2    3    Denmark    0  0  0  0   0   0   0    0
 3    4    Tunisia    0  0  0  0   0   0   0    0,
 'Group E':    Pos        Team  Pld  W  D  L  GF  GA  GD  Pts
 0    1       Spain    0  0  0  0   0   0   0    0
 1    2  Costa Rica    0  0  0  0   0   0   0    0
 2    3     Germany    0  0  0  0   0   0   0    0
 3    4       Japan    0  0  0  0   0   0   0    0,
 'Group F':    Pos     Team  Pld  W  D  L  GF  GA  GD  Pts
 0    1  Belgium    0  0  0  0   0   0   0    0
 1    2   Canada    0  0  0  0   0   0   0    0
 2    3  Morocco    0  0  0  0   0   0   0    0
 3    4  Croatia    0  0  0  0   0   0   0    0,
 'Group G':    Pos         Team  Pld  W  D  L  GF  GA  GD  Pts
 0    1       Brazil    0  0  0  0   0   0   0    0
 1    2       Serbia    0  0  0  0   0   0   0    0
 2    3  Switzerland    0  0  0  0   0   0   0    0
 3    4     Cameroon    0  0  0  0   0   0   0    0,
 'Group H':    Pos         Team  Pld  W  D  L  GF  GA  GD  Pts
 0    1     Portugal    0  0  0  0   0   0   0    0
 1    2        Ghana    0  0  0  0   0   0   0    0
 2    3      Uruguay    0  0  0  0   0   0   0    0
```

Bước 3: Phân tích dữ liệu

```
import numpy as np
fig, ax = plt.subplots(figsize=(12, 5), tight_layout=True)

sns.histplot(df, x='overall', binwidth=1)

bins = np.arange(df['overall'].min(), df['overall'].max(), 1)
plt.xticks(bins)
plt.show()
```



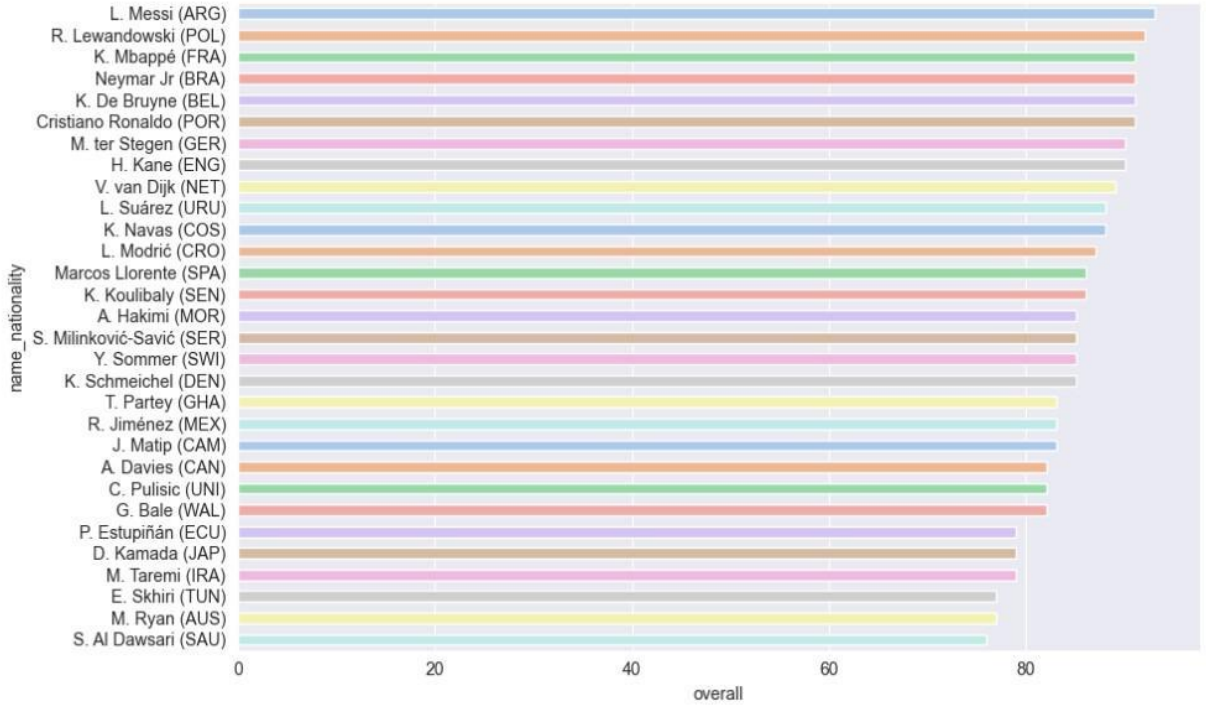
Phân bố chất lượng của cầu thủ, đa phần từ 60 - 73.

```
df.drop_duplicates('player_positions')
```

	short_name	age	nationality_name	overall	potential	club_name	value_eur	wage_eur	player_positions
0	L. Messi	34	Argentina	93	93	Paris Saint-Germain	78000000.0	320000.0	RW
1	R. Lewandowski	32	Poland	92	92	FC Bayern München	119500000.0	270000.0	ST
3	Neymar Jr	29	Brazil	91	91	Paris Saint-Germain	129000000.0	270000.0	LW
4	K. De Bruyne	30	Belgium	91	91	Manchester City	125500000.0	350000.0	CM
8	M. ter Stegen	29	Germany	90	92	FC Barcelona	99000000.0	250000.0	GK
19	J. Kimmich	26	Germany	89	90	FC Bayern München	108000000.0	160000.0	CDM
15	V. van Dijk	29	Netherlands	89	89	Liverpool	86000000.0	230000.0	CB
28	Bruno Fernandes	26	Portugal	88	89	Manchester United	107500000.0	250000.0	CAM
44	T. Alexander-Arnold	22	England	87	92	Liverpool	114000000.0	150000.0	RB
45	J. Sancho	21	England	87	91	Manchester United	116500000.0	150000.0	RM
41	P. Dybala	27	Argentina	87	88	Juventus	93000000.0	160000.0	CF
64	K. Coman	25	France	86	87	FC Bayern München	81000000.0	120000.0	LM
50	Jordi Alba	32	Spain	86	86	FC Barcelona	47000000.0	200000.0	LB
180	Angeliño	24	Spain	83	86	RB Leipzig	46000000.0	77000.0	LWB
379	R. James	21	England	81	86	Chelsea	37000000.0	76000.0	RWB

Sau đó chúng ta có được một đội hình gồm những cầu thủ giỏi nhất ở kì WC 2022, với đặc điểm chung là overall, potential, và giá trị chuyển nhượng value theo Euro đều cao ngất ngưỡng ở vị trí đó.

```
df_best_players = df.copy()
df_best_players = df_best_players.drop_duplicates('nationality_name').reset_index(drop=True)
country_short =  df_best_players['nationality_name'].str.extract('(^\w{3})', expand=False).str.upper()
df_best_players['name_nationality'] = df_best_players['short_name'] +' (' + country_short + ')'

fig, ax = plt.subplots(figsize=(10, 6), tight_layout=True)

sns.barplot(df_best_players, x='overall', y='name_nationality',
            palette=sns.color_palette('pastel'), width=0.5)
plt.show()
```



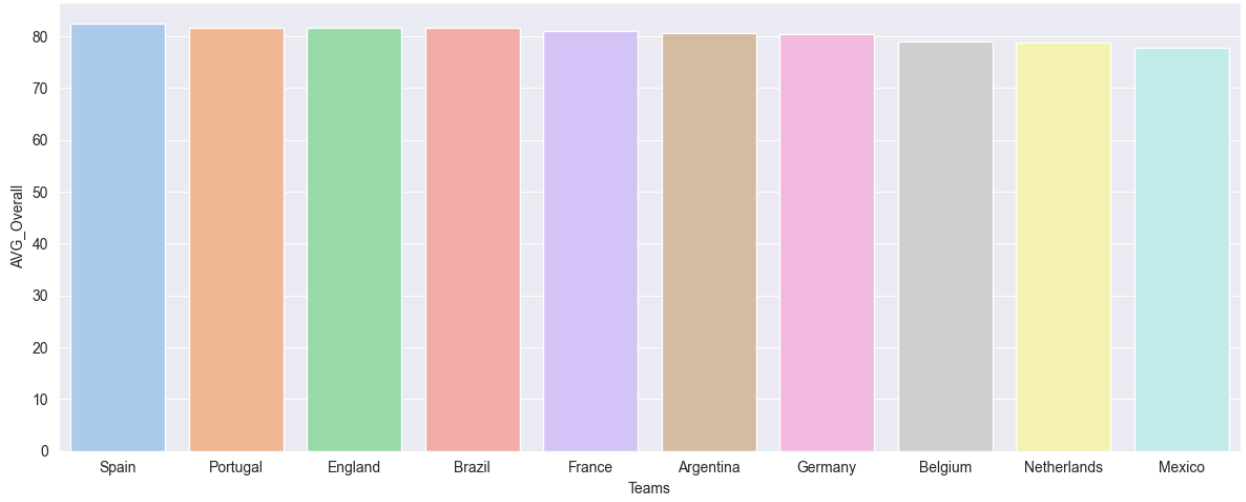
Đây biểu đồ nhiệt cho những cầu thủ có chỉ số tổng cao nhất giải

```
average_overall = [best_squad(team)['overall'].mean() for team in teams_worldcup]

df_average_overall = pd.DataFrame({'Teams': teams_worldcup, 'AVG_Overall': average_overall})
df_average_overall = df_average_overall.dropna()
df_average_overall = df_average_overall.sort_values('AVG_Overall', ascending=False)
df_average_overall
```

	Teams	AVG_Overall
6	Spain	82.400000
7	Portugal	81.733333
5	England	81.700000
1	Brazil	81.666667
3	France	81.000000
4	Argentina	80.566667
11	Germany	80.433333
2	Belgium	79.034483
9	Netherlands	78.758621
8	Mexico	77.727273
15	Croatia	76.760000
12	Uruguay	76.692308

Dĩ nhiên, khi đã biết được các cầu thủ xuất sắc nhất là ai, chúng ta sẽ tiếp tục quan tâm tới, vậy đội tuyển quốc gia nào đang được đánh giá cao nhất.

```
fig, ax = plt.subplots(figsize=(12, 5), tight_layout=True)

sns.barplot(df_average_overall[:10], x='Teams', y='AVG_Overall',
            palette=sns.color_palette('pastel'))
plt.show()
```



Đây là biểu đồ cột dành cho 10 đội mạnh nhất trong kì World Cup 2022.

<u>Bước 4</u>: Xây dựng mô hình sử dụng phân phối poisson.

Trước hết chúng ta cần tính toán sức mạnh đội hình của từng đội tuyển và kỳ vọng bàn thắng họ phải nhận và ghi được.

```
df_home = df_historical_data[['HomeTeam', 'HomeGoals', 'AwayGoals']]
df_away = df_historical_data[['AwayTeam', 'HomeGoals', 'AwayGoals']]

df_home = df_home.rename(columns={'HomeTeam':'Team', 'HomeGoals': 'GoalsScored', 'AwayGoals': 'GoalsConceded'})
df_away = df_away.rename(columns={'AwayTeam':'Team', 'HomeGoals': 'GoalsConceded', 'AwayGoals': 'GoalsScored'})

df_team_strength = pd.concat([df_home, df_away], ignore_index=True).groupby(['Team']).mean()
df_team_strength
```

Team	GoalsScored	GoalsConceded
Algeria	1.000000	1.461538
Angola	0.333333	0.666667
Argentina	1.691358	1.148148
Australia	0.812500	1.937500
Austria	1.482759	1.620690
...	...	...
Uruguay	1.553571	1.321429
Wales	0.800000	0.800000
West Germany	2.112903	1.241935
Yugoslavia	1.666667	1.272727
Zaire	0.000000	4.666667

85 rows × 2 columns

Sau đó dựa vào hàm df_steam_strength, ta xây dựng được hàm predict_point dựa vào phân phối poisson như sau:

```
def predict_points(home, away):
    if home in df_team_strength.index and away in df_team_strength.index:
        lamb_home = df_team_strength.at[home,'GoalsScored'] * df_team_strength.at[away,'GoalsConceded']
        lamb_away = df_team_strength.at[away,'GoalsScored'] * df_team_strength.at[home,'GoalsConceded']
        prob_home, prob_away, prob_draw = 0, 0, 0
        for x in range(0,11):
            for y in range(0, 11):
                p = poisson.pmf(x, lamb_home) * poisson.pmf(y, lamb_away)
                if x == y:
                    prob_draw += p
                elif x > y:
                    prob_home += p
                else:
                    prob_away += p

        points_home = 3 * prob_home + prob_draw
        points_away = 3 * prob_away + prob_draw
        return (points_home, points_away)
    else:
        return (0, 0)
```

Chạy thử hàm

```
print(predict_points('England', 'United States'))
print(predict_points('Argentina', 'Mexico'))
print(predict_points('Qatar (H)', 'Ecuador'))
```

```
(2.2356147635326007, 0.5922397535606193)
(2.3129151525530505, 0.5378377125059863)
(0, 0)
```

Sau đó chúng ta bắt đầu dự đoán kết quả của kì WC 2022

Dự đoán kết quả WorldCup

Vòng Bảng(Group States)

```
df_fixture_group_48 = df_fixture[:48].copy()
df_fixture_knockout = df_fixture[48:56].copy()
df_fixture_quarter = df_fixture[56:60].copy()
df_fixture_semi = df_fixture[60:62].copy()
df_fixture_final = df_fixture[62:].copy()
```

```
for group in dict_table:
    teams_in_group = dict_table[group]['Team'].values
    df_fixture_group_6 = df_fixture_group_48[df_fixture_group_48['home'].isin(teams_in_group)]
    for index, row in df_fixture_group_6.iterrows():
        home, away = row['home'], row['away']
        points_home, points_away = predict_points(home, away)
        dict_table[group].loc[dict_table[group]['Team'] == home, 'Pts'] += points_home
        dict_table[group].loc[dict_table[group]['Team'] == away, 'Pts'] += points_away

    dict_table[group] = dict_table[group].sort_values('Pts', ascending=False).reset_index()
    dict_table[group] = dict_table[group][['Team', 'Pts']]
    dict_table[group] = dict_table[group].round(0)
```

```
dict_table['Group A']
```

	Team	Pts
0	Netherlands	4.0
1	Senegal	2.0
2	Ecuador	2.0
3	Qatar (H)	0.0

Vòng loại trực tiếp(Knock_out)

```
df_fixture_knockout
```

	home	score	away	year
48	Winners Group A	Match 49	Runners-up Group B	2022
49	Winners Group C	Match 50	Runners-up Group D	2022
50	Winners Group D	Match 52	Runners-up Group C	2022
51	Winners Group B	Match 51	Runners-up Group A	2022
52	Winners Group E	Match 53	Runners-up Group F	2022
53	Winners Group G	Match 54	Runners-up Group H	2022
54	Winners Group F	Match 55	Runners-up Group E	2022
55	Winners Group H	Match 56	Runners-up Group G	2022

```
for group in dict_table:
    group_winner = dict_table[group].loc[0, 'Team']
    runners_up = dict_table[group].loc[1, 'Team']
    df_fixture_knockout.replace({f'Winners {group}':group_winner,
                                 f'Runners-up {group}':runners_up}, inplace=True)

df_fixture_knockout['winner'] = '?'
df_fixture_knockout
```

	home	score	away	year	winner
48	Netherlands	Match 49	Wales	2022	?
49	Argentina	Match 50	Denmark	2022	?
50	France	Match 52	Poland	2022	?
51	England	Match 51	Senegal	2022	?
52	Germany	Match 53	Belgium	2022	?
53	Brazil	Match 54	Uruguay	2022	?
54	Croatia	Match 55	Spain	2022	?
55	Portugal	Match 56	Switzerland	2022	?

```
def get_winner(df_fixture_updated):
    for index, row in df_fixture_updated.iterrows():
        home, away = row['home'], row['away']
        points_home, points_away = predict_points(home, away)
        if points_home > points_away:
            winner = home
        else:
            winner = away
        df_fixture_updated.loc[index, 'winner'] = winner
    return df_fixture_updated
```

```
get_winner(df_fixture_knockout)
```

	home	score	away	year	winner
48	Netherlands	Match 49	Wales	2022	Netherlands
49	Argentina	Match 50	Denmark	2022	Argentina
50	France	Match 52	Poland	2022	France
51	England	Match 51	Senegal	2022	England
52	Germany	Match 53	Belgium	2022	Germany
53	Brazil	Match 54	Uruguay	2022	Brazil
54	Croatia	Match 55	Spain	2022	Spain
55	Portugal	Match 56	Switzerland	2022	Portugal

Tứ Kết(Quarter Final)

```
def update_table(df_fixture_round_1, df_fixture_round_2):
    for index, row in df_fixture_round_1.iterrows():
        winner = df_fixture_round_1.loc[index, 'winner']
        match = df_fixture_round_1.loc[index, 'score']
        df_fixture_round_2.replace({f'Winners {match}':winner}, inplace=True)
    df_fixture_round_2['winner'] = '?'
    return df_fixture_round_2
```

```
update_table(df_fixture_knockout, df_fixture_quarter)
```

	home	score	away	year	winner
56	Germany	Match 58	Brazil	2022	?
57	Netherlands	Match 57	Argentina	2022	?
58	Spain	Match 60	Portugal	2022	?
59	England	Match 59	France	2022	?

```
get_winner(df_fixture_quarter)
```

	home	score	away	year	winner
56	Germany	Match 58	Brazil	2022	Brazil
57	Netherlands	Match 57	Argentina	2022	Netherlands
58	Spain	Match 60	Portugal	2022	Portugal
59	England	Match 59	France	2022	France

Bán kết(SemiFinal)

```
update_table(df_fixture_quarter, df_fixture_semi)
```

	home	score	away	year	winner
60	Netherlands	Match 61	Brazil	2022	?
61	France	Match 62	Portugal	2022	?

```
get_winner(df_fixture_semi)
```

	home	score	away	year	winner
60	Netherlands	Match 61	Brazil	2022	Brazil
61	France	Match 62	Portugal	2022	France

Chung Kết(Final)

```
update_table(df_fixture_semi, df_fixture_final)
```

	home	score	away	year	winner
62	Losers Match 61	Match 63	Losers Match 62	2022	?
63	Brazil	Match 64	France	2022	?

```
get_winner(df_fixture_final)
```

	home	score	away	year	winner
62	Losers Match 61	Match 63	Losers Match 62	2022	Losers Match 62
63	Brazil	Match 64	France	2022	Brazil

KẾT LUẬN

Dễ thấy, kết quả này chỉ một nửa là dự đoán đúng đó là đội tuyển Pháp sẽ vào chung kết, trong khi đó thực tế Brazil đã bị loại từ Tứ Kết, và người chiến thắng chung cuộc cũng không phải là đội tuyển Pháp.

Nếu chúng ta xét rộng ra hơn ở những vòng đấu loại và kết quả vòng bảng, cũng có sự sai lệch nhất định và tại sao lại như vậy ?

Ví dụ điển hình là trận khai mạc World Cup 2022 năm nay, Argentina đã thua sốc trước Saudi Arabia với tỉ số 2 - 1, dù trong bộ dữ liệu, sức mạnh tổng của Argentina là rất vượt trội so với đội tuyển từ Châu Á này (81 so với 68).

Một trường hợp khác là Ma Rốc, đội tuyển này có Overall 75, theo dự đoán thậm chí không thể vượt qua được vòng bảng, nhưng trong thực tế kỳ WC năm nay lại tiến rất sâu khi vào đến tận bán kết. Trong khi một số đội tuyển như Bồ Đào Nha đã bị loại từ tứ kết, nhưng dự đoán theo số liệu thì lại tiến đến tận Bán kết.

Câu trả lời khá đơn giản, chúng ta đang sống trong một thế giới phi tuyến. Dù cho kết quả của bóng đá lại theo kiểu của boolean tức là chỉ thắng hoặc thua. Nhưng những thứ tác động đến kết quả đó lại là phi tuyến với vô số các biến chúng ta không thể kiểm soát hết được. Ví dụ như tại sao Bồ Đào Nha lại được dự đoán vào bán kết, đơn giản là vì căn cứ theo số liệu phong độ trong quá khứ trước kì World Cup 2022, các cầu thủ của họ đều vào độ chín và có chỉ số đánh giá overall rất cao.

Nhưng, cầu thủ nổi bật nhất của Bồ đào nha là Ronaldo trước kì World Cup 2022, lại xuống phong độ thảm hại, dĩ nhiên, chúng ta nếu không cập nhật những tin tức xung quanh về thế giới bóng đá nói chung, sẽ rất nhiều người bất ngờ với kết quả của Bồ Đào Nha tại kì World Cup năm nay.

Cũng vậy tai nạn của Argentina với Ma Rốc ở trận khai mạc, là do một yếu tố khác, đó là thời tiết, sự thích nghi và phong độ của cầu thủ chưa được ổn định. Cần phải biết các quốc gia ở Nam Mỹ thường không nóng như cách mà Ả Rập Xê út đang phải

hứng chịu. Khi nhiệt độ đột ngột tăng hoặc giảm, tất nhiên các cầu thủ sẽ khó thích nghi nhanh được, dẫn đến vô số những trạng thái không tốt, ảnh hưởng đến phong độ của cầu thủ đó.

Rõ ràng, các thuật toán dự đoán kết quả trận đấu đang có mặt ở hiện tại chưa đủ thực tế dù đã cải thiện rất nhiều so với quá khứ, chúng ta còn vô số những biến khó kiểm soát như phong độ của một cầu thủ bị ảnh hưởng bởi cái gì, thời tiết hay là nhiệt độ, hay vì chuyện gia đình ?

Đã có đội bóng nào có những đấu pháp tinh tế và thú vị hơn để làm suy giảm đấu pháp của những đội mạnh ? Những câu hỏi này rất nhiều, tuy vậy để xây dựng được một hệ thống như thế chưa thể thực hiện ngay ở thời điểm hiện tại. Bởi vì tiền đổ vào những dự án như thế này là rất lớn.

Do đó cho đến khi có một công ty dữ liệu nào đó sẵn sàng chi hàng tấn tiền để mời từng cầu thủ tham gia thử nghiệm quá trình đánh giá phong độ của cầu thủ đó trong nhiều môi trường, nhiều trường hợp khác nhau, thì khi đó chúng ta vẫn sẽ phải trung thành với kiểu dự đoán tuyến tính được tính theo kiểu số lần ăn nhau giữa hai đội khi đối đầu với nhau trong quá khứ, thì ai hơn ai.