Simulação Ganhador da Copa do Mundo 2022

Aluno:

Rodrigo Lacerda

Professor:

José Roberto Castilho Piqueira

21/10/2022

Sumário

Introdução

A cada 4 anos acontece um dos eventos esportivos mais vistos do mundo, o Campeonato Mundial de Futebol, onde 32 países disputam para conquistar o título de campeão do mundo.

A ideia do projeto será desenvolver um algoritmo que simula as disputas começando da fase de “mata-mata”, quando restam 16 equipes, até a final, onde restarão somente duas seleções. Para determinar o ganhador de cada disputa será criado um algoritmo de machine learning para determinar a probabilidade de cada time de vencer sobre o outro.

Para conseguir treinar o modelo, usaremos conjuntos de dados sobre todas as partidas internacionais disputadas pelos países disponibilizadas pelo site Kaggle e usaremos como ferramenta o JupyterNotebook com as bibliotecas de ML do Python.

Esse trabalho e todos scripts foram publicados na minha página no Kaggle para contribuir com a comunidade.

Análise exploratória dos dados

O conjunto de dados foi adquirido no portal Kaggle, no seguinte link . O dataset possui todas as partidas de futebol internacionais oficiais e amigáveis disputadas desde 1930.

Nessa parte, o objetivo é entender melhor sobre os dados importados, tentaremos responder as seguintes perguntas:

- Quais são as equipes com melhores ranks da Fifa Atualmente?
- Quais são as equipes com maiores pontos?
- Quais são as equipes com melhores ataques?
- Quais são as equipes com melhores defesas?
- Quais são as equipes com melhores de forma geral?

1. Análise geral do banco de dados.

	date	home_team	away_team	home_team_continent	away_team_continent	home_team_fifa_rank	away_team_fifa_rank	home_team_total_fifa_points	away_team_total_fifa_points
23916	2022-06-14	Moldova	Andorra	Europe	Europe	180	153	932	1040
23917	2022-06-14	Liechtenstein	Latvia	Europe	Europe	192	135	895	1105
23918	2022-06-14	Chile	Ghana	South America	Africa	28	60	1526	1387
23919	2022-06-14	Japan	Tunisia	Asia	Africa	23	35	1553	1499
23920	2022-06-14	Korea Republic	Egypt	Asia	Africa	29	32	1519	1500

5 rows × 25 columns

No total temos 25 variáveis sobre as partidas internacionais, como a data da partida, qual time era de casa e qual era de fora, os ranks e também os pontos FIFA no momento da partida, e variáveis quantitativas das partidas, como a quantidade de pontos marcados, o score do ataque, defesa e meio de campo dos times, entre outros.

Com a função *describe*, podemos obter alguns insights dos dados numéricos. Nesse dataset não existe valores faltantes, o que poderia requerer algum tipo de tratamento mais na frente. Podemos observar que os menores e maiores valores são 1 e 211 respectivamente, indicando a quantidade de seleções distintas que estão dentro da FIFA. Outro ponto curioso de se observar é que a média do rank FIFA é maior para a seleção de fora da casa, enquanto que os pontos FIFA médio são menores comprados com o da casa.

```
[3]
df.describe()
```

	home_team_fifa_rank	away_team_fifa_rank	home_team_total_fifa_points	away_team_total_fifa_points	home_team_score	away_team_score
count	23921.000000	23921.000000	23921.000000	23921.000000	23921.000000	23921.000000
mean	77.854688	80.797375	323.401488	315.453576	1.609214	1.068266
std	52.355225	53.232902	500.825725	490.944273	1.630127	1.263944
min	1.000000	1.000000	0.000000	0.000000	0.000000	0.000000
25%	33.000000	36.000000	0.000000	0.000000	0.000000	0.000000
50%	71.000000	73.000000	0.000000	0.000000	1.000000	1.000000
75%	115.000000	119.000000	547.000000	523.000000	2.000000	2.000000
max	211.000000	211.000000	2164.000000	2164.000000	31.000000	21.000000

2. Análise exploratória

2.1 Top ranks da FIFA mais recentes

Determinar os últimos top 10 ranks da FIFA encontrados no banco de dados, com a ideia de determinar qual time está mais forte atualmente.

```
#let's re-shape the data a bit
fifa_rank = df[['date', 'home_team', 'away_team', 'home_team_fifa_rank', 'away_team_fifa_rank',
'away_team_total_fifa_points','home_team_total_fifa_points']]
home = fifa_rank[['date', 'home_team','home_team_fifa_rank','home_team_total_fifa_points']].rename(columns = {'home_team': 'team', 'home_team_fifa_rank' : 'rank','home_team_total_fi
away = fifa_rank[['date', 'away_team','away_team_fifa_rank','away_team_total_fifa_points']].rename(columns = {'away_team': 'team', 'away_team_fifa_rank' : 'rank','away_team_total_fi
fifa_rank = home.append(away)

#select for each country the latest match
fifa_rank  = fifa_rank.sort_values(['team', 'date'], ascending=[True, False])
fifa_rank['row_number'] = fifa_rank.groupby('team').cumcount()+1
fifa_rank_top = fifa_rank[fifa_rank['row_number']==1].drop('row_number',axis=1).nsmallest(10, 'rank')
#fifa_points_top = fifa_rank[fifa_rank['row_number']==1].drop('row_number',axis=1).nlargest(10, 'rank_points')

#let's see the 5 strongest teams
fifa_rank_top
```

	date	team	rank	rank_points
23760	2022-06-06	Brazil	1	1832
23909	2022-06-14	Belgium	2	1827
23885	2022-06-13	France	3	1789
23741	2022-06-05	Argentina	4	1765
23906	2022-06-14	England	5	1761
23907	2022-06-14	Italy	6	1723
23866	2022-06-12	Spain	7	1709
23867	2022-06-12	Portugal	8	1674
23903	2022-06-14	Mexico	9	1658
23908	2022-06-14	Netherlands	10	1658

Como observado, Brasil Bélgica e França são as seleções que estão mais recentemente no pódio do rank FIFA.

2.2 Melhores Ataques gerais

```
offense = df[['date', 'home_team', 'away_team', 'home_team_mean_offense_score', 'away_team_mean_offense_score']]
home = offense [['date', 'home_team','home_team_mean_offense_score']].rename(columns = {'home_team': 'team', 'home_team_mean_offense_score' : 'offense_score'})
away = offense [['date', 'away_team','away_team_mean_offense_score']].rename(columns = {'away_team': 'team', 'away_team_mean_offense_score' : 'offense_score'})
offense  = home.append(away)
# offense
#last match that each country played
offense   = offense .sort_values(['team', 'date'], ascending=[True, False])
offense ['row_number'] = offense .groupby('team').cumcount()+1
offense_top_data = offense [offense ['row_number']==1].drop('row_number',axis=1).nlargest(10, 'offense_score')
offense_top_data
```

	date	team	offense_score
23741	2022-06-05	Argentina	89.0
23885	2022-06-13	France	88.3
23906	2022-06-14	England	88.0
23760	2022-06-06	Brazil	86.3
23867	2022-06-12	Portugal	86.0
23909	2022-06-14	Belgium	85.7
23907	2022-06-14	Italy	85.3
23866	2022-06-12	Spain	85.0
23909	2022-06-14	Poland	84.7
23862	2022-06-11	Uruguay	84.3

Os maiores ataques mudam um pouco, apresentando Argentina, liderada por Lionel Messi, como o melhor ataque atualmente.

2.3 Melhores Defesas

```
defense = df[['date', 'home_team', 'away_team', 'home_team_mean_defense_score', 'away_team_mean_defense_score']]
home = defense [['date', 'home_team','home_team_mean_defense_score']].rename(columns = {'home_team': 'team', 'home_team_mean_defense_score' : 'defense_score'})
away = defense [['date', 'away_team','away_team_mean_defense_score']].rename(columns = {'away_team': 'team', 'away_team_mean_defense_score' : 'defense_score'})
defense  = home.append(away)
# offense
#last match that each country played
defense   = defense .sort_values(['team', 'date'], ascending=[True, False])
defense ['row_number'] = defense .groupby('team').cumcount()+1
defense_top_data = defense [offense ['row_number']==1].drop('row_number',axis=1).nlargest(10, 'defense_score')
defense_top_data
```

	date	team	defense_score
23866	2022-06-12	Spain	86.5
23908	2022-06-14	Netherlands	85.2
23867	2022-06-12	Portugal	85.2
23906	2022-06-14	England	85.0
23760	2022-06-06	Brazil	84.8
23885	2022-06-13	France	84.2
23907	2022-06-14	Italy	84.2
23907	2022-06-14	Germany	84.0
23741	2022-06-05	Argentina	82.2
23879	2022-06-13	Morocco	81.2

Nesse resultado, temos Espanha e Holanda como as melhores defesas do mundo, realmete faz sentido levando em conta a Zaga forte dessas duas seleções, com Sergio Ramos (Espanha) e Van Dijk (Holanda).

2.4 Melhores Meio Campo

```
midfield = df[['date', 'home_team', 'away_team', 'home_team_mean_midfield_score', 'away_team_mean_midfield_score']]
home = midfield [['date', 'home_team','home_team_mean_midfield_score']].rename(columns = {'home_team': 'team', 'home_team_mean_midfield_score' : 'midfield_score'})
away = midfield [['date', 'away_team','away_team_mean_midfield_score']].rename(columns = {'away_team': 'team', 'away_team_mean_midfield_score' : 'midfield_score'})
midfield  = home.append(away)
# offense
#last match that each country played
midfield   = midfield .sort_values(['team', 'date'], ascending=[True, False])
midfield ['row_number'] = defense .groupby('team').cumcount()+1
midfield_top_data = midfield [offense ['row_number']==1].drop('row_number',axis=1).nlargest(10, 'midfield_score')
midfield_top_data
```

	date	team	midfield_score
23907	2022-06-14	Germany	87.8
23885	2022-06-13	France	86.8
23866	2022-06-12	Spain	86.0
23909	2022-06-14	Belgium	85.5
23760	2022-06-06	Brazil	85.5
23907	2022-06-14	Italy	84.5
23867	2022-06-12	Portugal	84.5
23885	2022-06-13	Croatia	84.2
23741	2022-06-05	Argentina	84.0
23906	2022-06-14	England	84.0

Nesse resultado temos Alemanha e Franca com o melhor meio campo, também condizente com o que as duas seleções são especialistas. Com N'Golo Kanté (França) e Thomas Muller (Alemanha).

3. Feature Engineering

Para conseguir gerar um modelo mais assertivo, podemos criar novos parâmetros combinando duas colunas, como por exemplo

Rank Difference – Diferença de Rank FIFA entre os dois times;

Average Rank – Média de rank FIFA entre os dois times;

Point Difference – Diferença de pontos totais FIFA entre os dois times.

Is Won – Se a diferenca de pontos da partida entre o time da casa de fora for maior que 0, significa que o time da casa (home) venceu a partida, caso seja menor ou igual (empate) a 0, será considerado que o time da casa perdeu.

Modelo de Regressão Logística

Nessa sessão iremos modelar o algoritmo de Machine Learning para desenvolver um modelo gerado a partir de inputs de partidas internacionais, e depois, gerar uma previsão com base estatística com novos dados, para determinar quem será o ganhador caso disputassem uma partida.

Como modelo, será utilizado a Regressão Logística disponibilizada na biblioteca Sci-Kit Learn. Como parâmetros, foram utilizados os paremtros criados por Feature Engineering, Rank Difference, Average Difference e Points Difference.

Como valor alvo para se preve será utilizado o campo Is_Won.

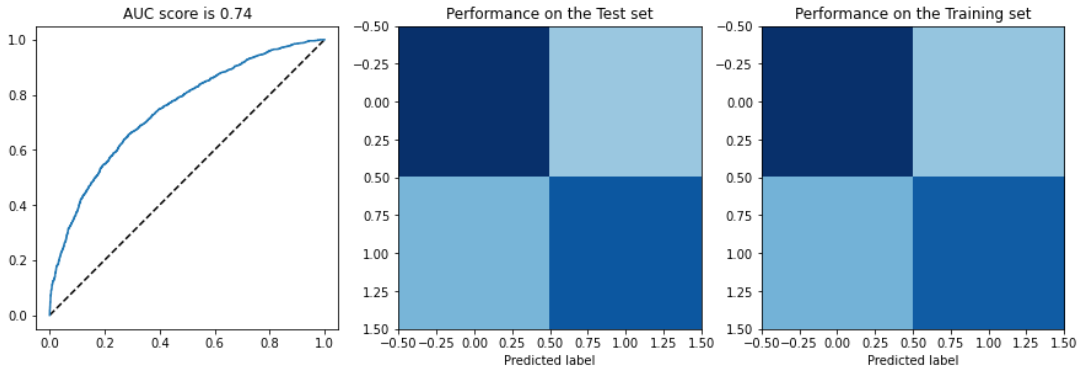
Foi dividido o dataset entre dois subconjuntos de dados, um para treinar com 80% do total, e o de teste, para testar o modelo, com 20%.

```
from sklearn import linear_model
from sklearn import ensemble
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix, roc_curve, roc_auc_score
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import PolynomialFeatures

#X, y = df.loc[:,['average_rank', 'rank_difference', 'point_difference','is_stake']], df['is_won']
X, y = df.loc[:,['average_rank', 'rank_difference', 'point_difference']], df['is_won']
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42)

logreg = linear_model.LogisticRegression(C=1e-5)
features = PolynomialFeatures(degree=2)
model = Pipeline([
    ('polynomial_features', features),
    ('logistic_regression', logreg)
])
model = model.fit(X_train, y_train)
```

Para testar a performance desse modelo, foi avaliado com base no score AUC (Area under curve). O resultado foi um score de 0.74, o que significa que o modelo irá prever 74% das vezes o resultado correto do time que irá vencer a disputa. Por fins de aprendizagem, como já temos um resultado melhor do que um Cara-Coroa, podemos adotar esse modelo para a simulação.



Simulação

Nessa sessão, após desenvolver o modelo, vamos montar um simulador da copa do mundo 2022 e ajustar o dataset para ela. como limitante, o simulador considera somente a fase “mata-mata” até o final, as fases de grupos foram decididas pelo autor com base em experiencia própria sobre o futebol.

Também, foi limitado o conjunto de dados para data a partir de 01-01-2021 até 14-06-2022, isso para selecionar dados mais recentes, se considerado todo o dataset de partidas a partir de 1930, pediríamos ter resultados mais fora do comum.

Outra limitação da simulação, foi considerado como time da casa ou time fora da casa dependendo da sua colocação dentro da lista de países selecionados para o Mata-Mata, significando que cada time numa posição impar será considerado como da casa (Senegal, England ...) e times numa posição par como fora da casa (Holanda, USA ...)

Times selecionados para Mata-Mata (16):

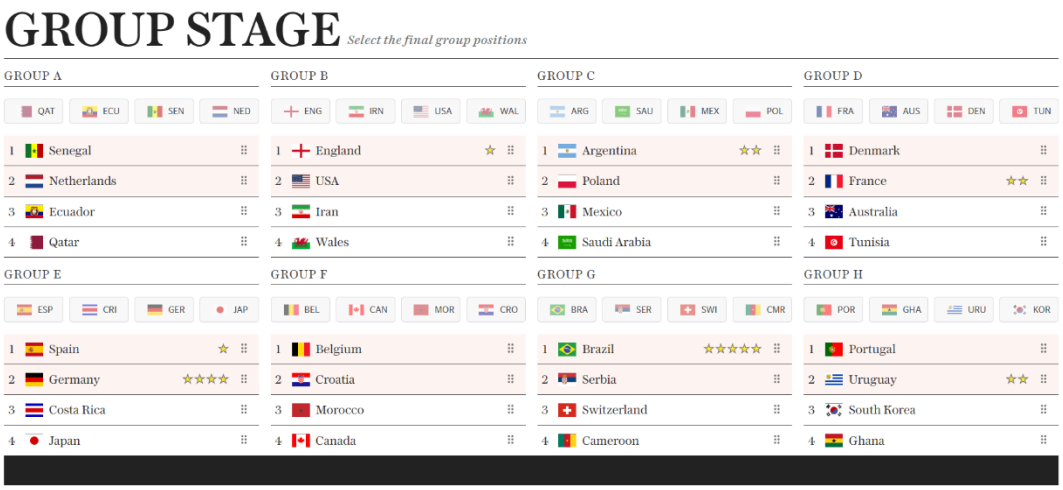
Senegal, Holanda, England, USA, Argentina, Poland, France, Denmark, Spain, Germany, Belgium, Croatia, Brazil, Serbia, Portugal, Uruguay.

O dataset usado na simulação será desenvolvido considerando todas essas premissas e agregando os dados para termos os valores médios de rank e total_points.

```
teams_worldcup = ['Qatar', 'Ecuador', 'Senegal', 'Netherlands', 'England', 'Iran', 'USA',
                  'Wales', 'Argentina', 'Saudi Arabia', 'Mexico', 'Poland', 'France',
                  'Australia', 'Denmark', 'Tunisia', 'Spain', 'Costa Rica', 'Germany',
                  'Japan', 'Belgium', 'Canada', 'Morocco', 'Croatia', 'Brazil', 'Serbia',
                  'Switzerland', 'Cameroon', 'Portugal', 'Ghana', 'Uruguay', 'South Korea']
✓ 0.7s

world_cup_rankings_home = df[['home_team','home_team_fifa_rank','home_team_total_fifa_points']].loc[df['home_team'].isin(teams_worldcup) & (df['date']>'2021-01-01')]
world_cup_rankings_away = df[['away_team','away_team_fifa_rank','away_team_total_fifa_points']].loc[df['away_team'].isin(teams_worldcup)& (df['date']>'2021-01-01')]
world_cup_rankings_home = world_cup_rankings_home.set_index(['home_team'])
✓ 0.6s

#world_cup_rankings = world_cup_rankings.groupby('away_team').mean()
world_cup_rankings_home = world_cup_rankings_home.groupby('home_team').mean()
world_cup_rankings_away = world_cup_rankings_away.groupby('away_team').mean()
✓ 0.4s
```



Script:

O script funciona como uma iteração de iterações, a primeira iteração é a quantidade de simulações adotadas, para esse estudo foram feitas 1000 simulações. Depois, temos a iteração da etapa do torneio (round_16, quarter-final, semi-final e final). Por fim, temos a iteração entre os times enfrentados.

Para cada partida simulada, pegamos os nomes das equipes e retiramos os valores de rank e pontos para calcular os três parâmetros para usarmos no modelo de regressão logística, que são o a diferença de rank da partida (Rank Difference), o rank médio da partida (Average Rank) e a diferença de pontos FIFA (Point Difference). Tendo esses 3 paremtros da partida, coloca-se no modelo para determinar a porcentagem de chance de o time da casa vencer (home_prob_win).

Após saber a probabilidade de o time da casa vencer, é usado uma função randômica para determinar aleatoriamente se ele vencerá a partida ou não (essa é a parte mais importante do simulador, pois, determina aleatoriamente se o time com mais chances de vencer, realmente vencerá a partida ou não, o que acontece na vida real, nem sempre o time mais provável de vencer vence).

Exemplo de simulação:

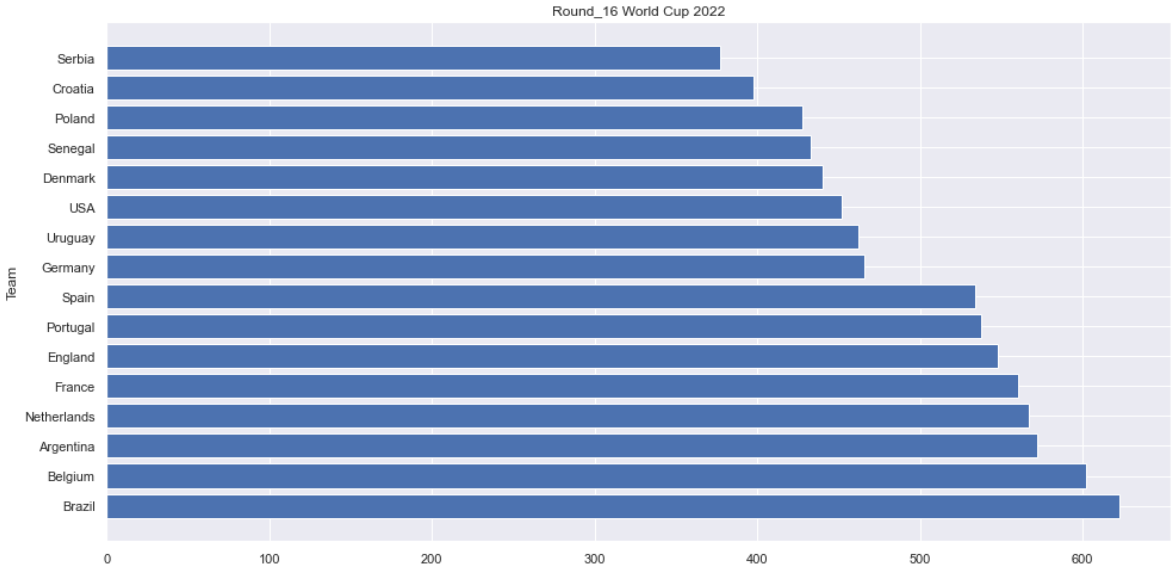
```
STEP: 8
[('Netherlands', 1), ('USA', 1), ('Poland', 1), ('Denmark', 1), ('Spain', 1), ('Belgium', 1), ('Brazil', 1), ('Uruguay', 1)]
STEP: 4
[('USA', 1), ('Poland', 1), ('Belgium', 1), ('Brazil', 1)]
STEP: 2
[('Poland', 1), ('Belgium', 1)]
STEP: 1
[('Belgium', 1)]
```

Aqui como exemplo de simulação, mostra os 8 times que vencerão a etapa round_16 (step 8), depois os 4 que vencerão o quarter-final (step 4), depois os 2 que vencerão a semi-final (step 2) e pôr fim a seleção ganhadora Bélgica.

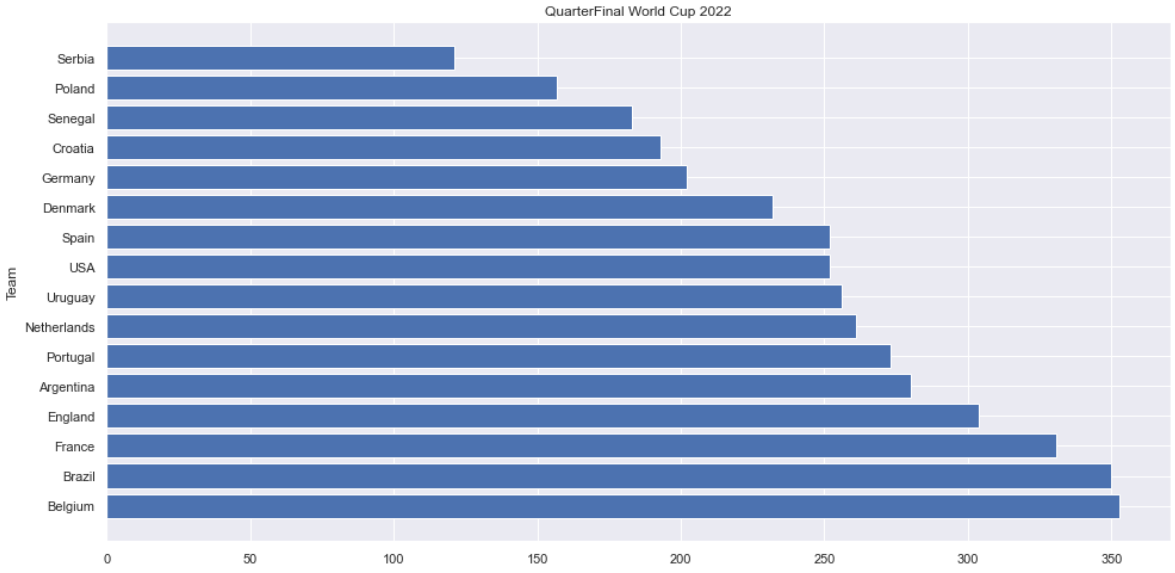
Resultados 1000 simulações:

Após rodar o simulador 1000 vezes, temos algumas interpretações interessantes para analisar:

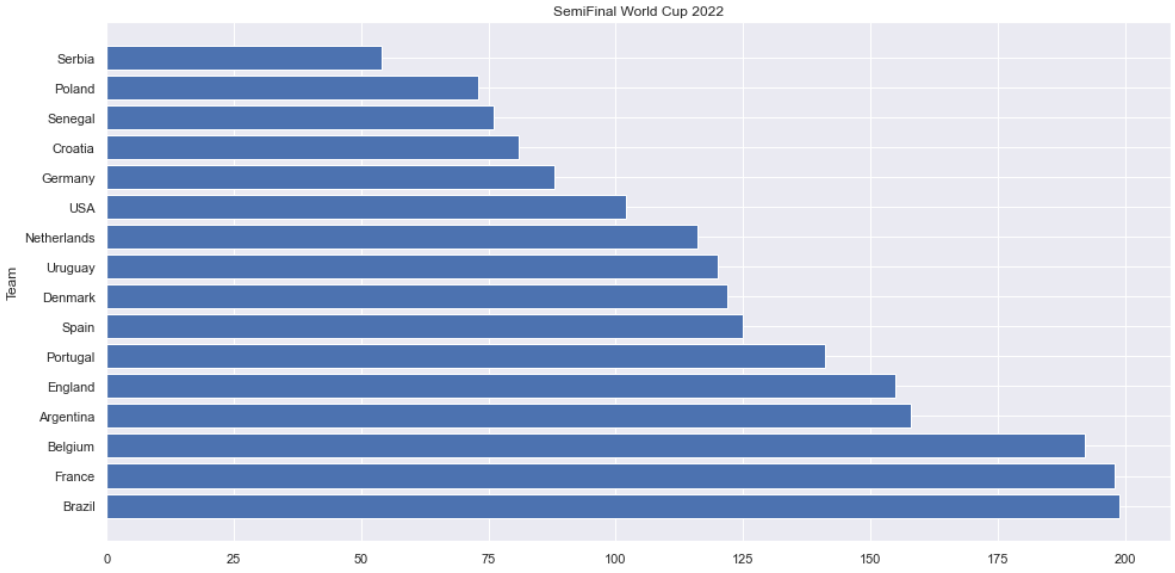
1) Quantidade de vezes que cada seleção avança para a fase da quarta de finais.



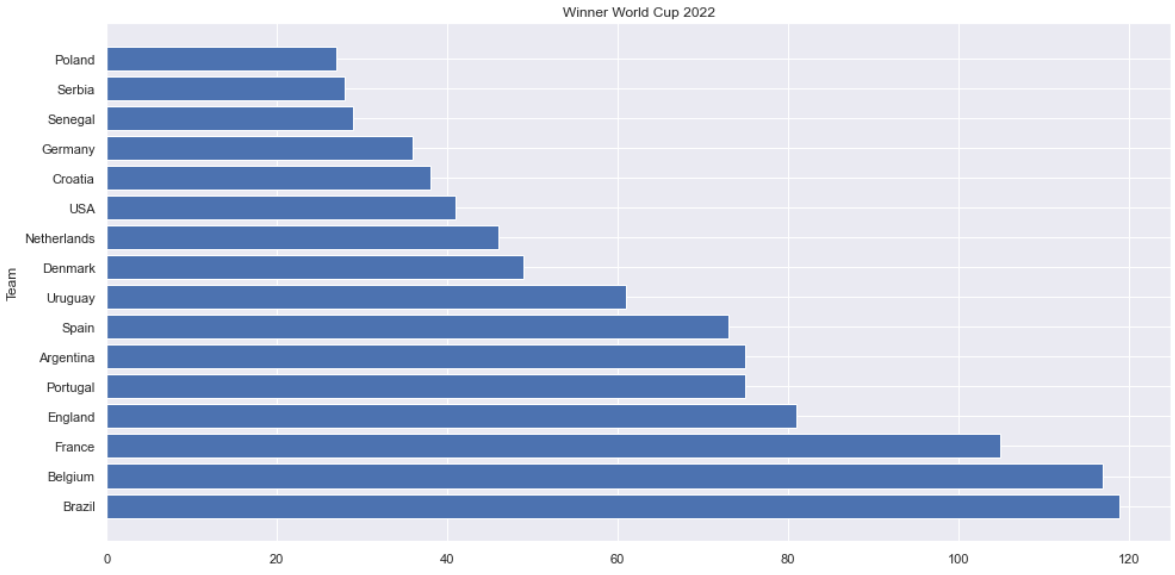
2) Quantidade de vezes que cada seleção avança para a fase da semifinais.



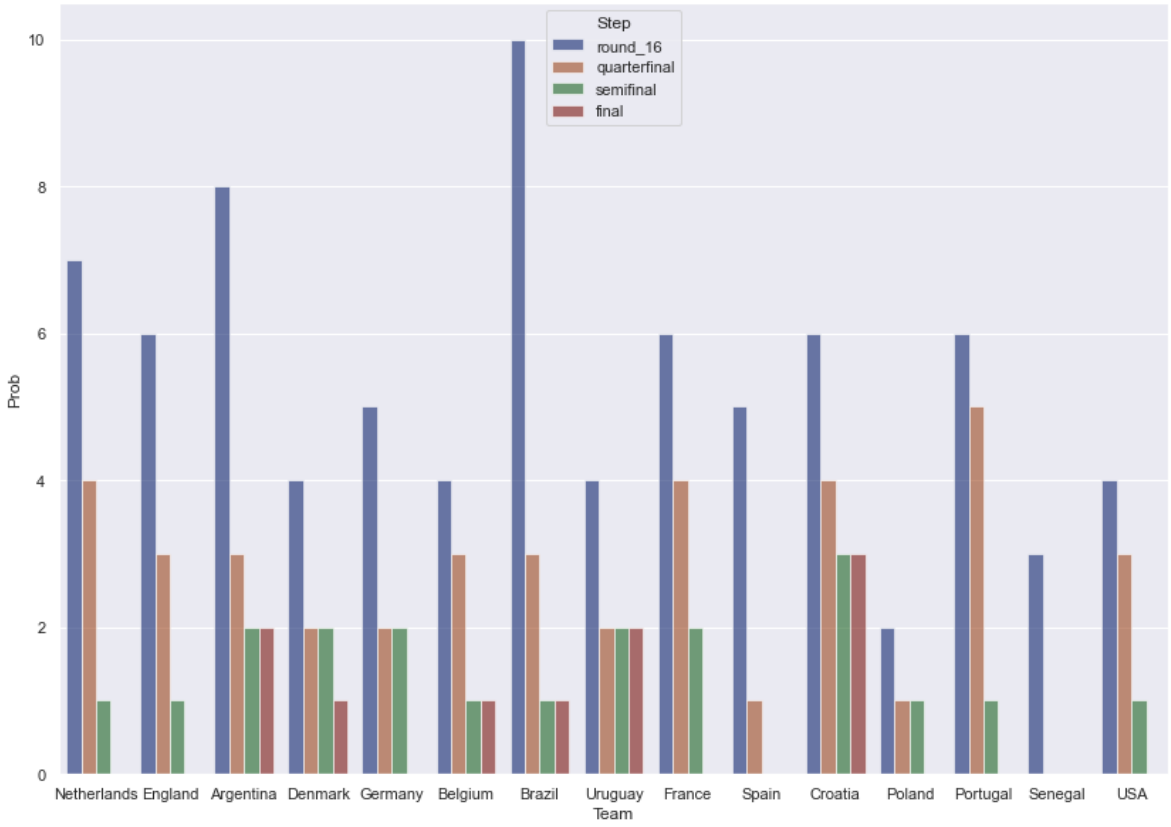
3) Quantidade de vezes que cada seleção avança para a fase das finais.



4) Quantidade de vezes que cada seleção vence a Copa do Mundo.



5) Visão compilada dos resultados.



Interpretações

1. Senegal e Sérvia provavelmente são as equipes mais fracas do torneio, por menos presença nas finais;

2. Croácia e Sérvia têm menos de 50% de chance de passar para as quartas de final (< 400/800 em 1.000 simulações);

3. A Bélgica é a 2ª com maior probabilidade de vencer, embora a França apareça mais nas meias-finais do que a Bélgica;

4. Os 3 melhores times são Brasil, Bélgica e França com mais de 10% de chances de vencer o torneio.

5. A cada etapa fica mais evidente a disparidade de força entre os 3 times favoritos e os outros, onde os favoritos a partis das quartas de final, já apresentam uma probabilidade de passar maior que o restante.

Conclusões

Após 1000 simulações o Brasil fica em primeiro lugar, seguido pela Bélgica quase empatada, que são 2 times muito bons favoritos neste torneio. É bom notar que a cada nova tentativa os pedidos mudam um pouco, a França e a Espanha aparecem em 1º ou 2º lugar às vezes. No geral resultados, foram bastante satisfatórios e esse projeto proporcionou desenvolver mais conhecimentos de programação, técnicas de machine learning e simulação.

Para projetos futuros, procurar um modelo de classificação com uma maior precisão, testar mais modelos com mais parâmetros e buscar uma configuração de dataset final para o simulador onde não importa tanto os valores de casa e fora, pois na copa do mundo praticamente todas seleções estão fora de casa.