MACHINE LEARNING

WORLD CUP 2026

PREDICTION

MARIELA QUINTANAR DE LA MORA

INTRODUCTION

We live in a digital age, where technical advancements have led to the everyday generation of enormous amounts of data. These facts don't have much value on their own, but we can transform them into useful insights by using machine learning techniques. Machine learning initiatives have evolved into essential instruments for making predictions in a variety of disciplines, providing numerous advantages that have a favorable influence on our environment.

First and foremost, machine learning initiatives are crucial for making wise decisions across many industries. These algorithms can find hidden patterns and new trends by examining vast historical data sets, giving organizations the ability to predict possible outcomes. Having accurate predictions might mean the difference between success and failure in sectors including finance, healthcare, logistics, and e-commerce.

The capacity of machine learning projects to spur innovation and technical advancement is another crucial factor. New products and services that better suit the requirements of people and society as a whole can be developed by offering data-driven insights and predictions. This includes everything from improvements in sustainability and environmental management to advancements in medicine and healthcare.

The purpose of this project is to harness the power of data analysis and machine learning to make more informed and strategic decisions in an increasingly complex and digitized world. These projects not only give us a deeper understanding of the patterns and trends hidden in large volumes of data, but also allow us to anticipate future situations with greater precision.

STEP 1

FORM A HYPOTHESIS:

We can predict which team is going to win the FIFA world Cup in 2026.

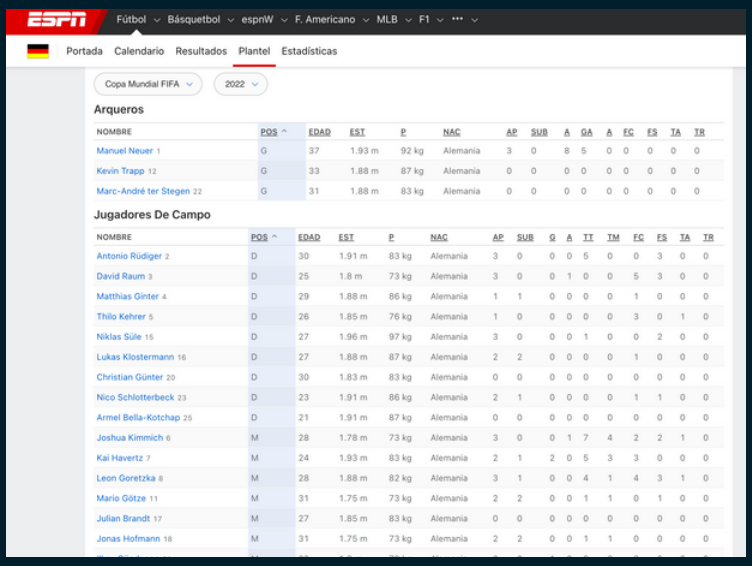
STEP 2

FIND THE NECESSARY DATA

The necessary data was found on the ESPN webpage since there is complete information on which teams played in each World Cup, who were the soccer players who participated, their age, weight, height, position, etc.

For this project the data found of which teams had participated in a certain world cup, number of players, players who participated, their age, weight, height and previous victories were used to make predictions.

ULR:https://www.espn.com.mx/futbol/equipo/plantel/_/id/481/ger



STEP 3

CLEAN DATA

Any prediction machine learning project must begin with the fundamental and crucial step of cleaning the data.

Before utilizing a data collection to train a machine learning model, data cleaning entails finding, fixing, or deleting mistakes, inconsistencies, and outliers.

This phase is crucial for a number of reasons:

Model accuracy: If there are outliers or missing data, the model may make predictions that are biased or wrong.

By improving the model's accuracy through data cleaning, predictions become more trustworthy and accurate.

Computational effectiveness: The size of the data collection is decreased by cleaning the data and eliminating redundant or unneeded observations. This enhances computing efficiency and quickens the model training process.

In this step we will remove the missing values from our data as they will not help our model to make accurate predictions.

STEP 4

ERROR METRIC

It is used to evaluate the performance of our machine learning model.

Mean absolute error is the error metric we will be using.

$$\frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

To obtain the error we subtract the predictions - the wins.

TEAM	YEAR	WINS	PREDICTION	ERROR
Brazil	1958, 1962, 1970, 1994 y 2002	5	1	4
France	1998, 2018	2	0	0
Australia	1980, 1996, 2000 y 2004	4	1	3

ERROR METRIC

After doing the subtractions we take the mean of all the individual errors.
We add up all the error values, and then we divide by the total number of predictions

STEP 5

SPLIT THE DATA

We need to split the data because we want to train on one part of the data and make predictions on the other part of the data.

We do this because if we train the algorithm in the same data that we use to evaluate it, it's like having an open book in a test, you might do great but it's probably because you memorized the answers not because you understand.

So we want to give our algorithm a new set of data that hasn't been trained on.
That new set of data will tell us how well the algorithm works.

STEP 6

TRAIN A MODEL

We are going to use linear regression to train our model.
Linear regression's formula:

$$Y = ax + B$$

In this model:

y: is the dependent variable that we want to predict.

x: is the independent variable or features used to make the prediction.

m: is the slope of the line (represents the relationship between x and y).

b: is the ordinate to the origin (the value of y when x is equal to 0)

Making predictions: Once the model is trained and has found the optimal values of m and b, it can be used to make predictions on new data. You simply enter the x value for which you want to predict the y value, and the model uses the linear regression equation to give the prediction.

CODE

IMPORT PANDAS AND DATA

```
[ ] import pandas as pd
    teams = pd.read_csv('teams.csv')
```

```
[ ] teams
```

	Team	Year	Athletes	Age	Height	Weight	Win	Prev_win	Prev_win_2
0	Brazil	2014	22	27.830	1.80	77.28	0	0	5
1	Brazil	2018	22	33.312	1.79	76.27	0	0	0
2	Brazil	2022	22	28.600	1.80	74.96	0	0	0
3	Cameroon	2010	22	38.180	1.82	77.82	0	0	2
4	Cameroon	2014	22	35.280	1.81	76.55	0	0	0
...	...	...	...	...	...	...	...	...	...
73	South Korea	2018	22	32.390	1.82	74.48	0	0	0
74	South Korea	2022	22	28.230	1.80	72.62	0	0	0
75	Russia	2002	22	39.330	1.81	74.15	0	0	0
76	Russia	2014	22	36.180	1.81	74.73	0	0	0
77	Russia	2018	22	33.430	1.85	77.04	0	0	0

DELETE EXTRA COLUMNS

```
# delete extra columns
teams = teams[["Team", "Year", "Athletes", "Age", "Win", "Prev_win"]]
```

```
teams
```

	Team	Year	Athletes	Age	Win	Prev_win
0	Brazil	2014	22	27.830	0	0
1	Brazil	2018	22	33.312	0	0
2	Brazil	2022	22	28.600	0	0
3	Cameroon	2010	22	38.180	0	0
4	Cameroon	2014	22	35.280	0	0
...	...	...	...	...	...	...
73	South Korea	2018	22	32.390	0	0
74	South Korea	2022	22	28.230	0	0
75	Russia	2002	22	39.330	0	0
76	Russia	2014	22	36.180	0	0
77	Russia	2018	22	33.430	0	0

CODE

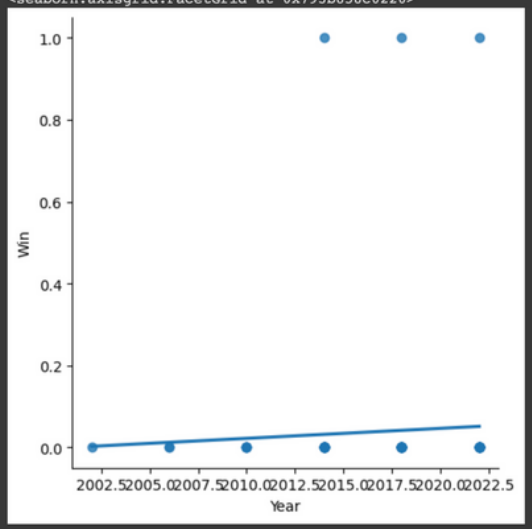
IMPORT PANDAS AND DATA

```
[ ] # checar si se pueden hacer predicciones
    teams.corr()['Win']

    <ipython-input-9-036e92e8696a>:2: FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future version, it will default to F
      teams.corr()['Win']
    Year        0.055874
    Athletes    0.022792
    Age        -0.041787
    Win         1.000000
    Prev_win   -0.032444
    Name: Win, dtype: float64
```

We check if we can make predictions with the data that we have.

PLOT YEAR AND WIN COLUMNS

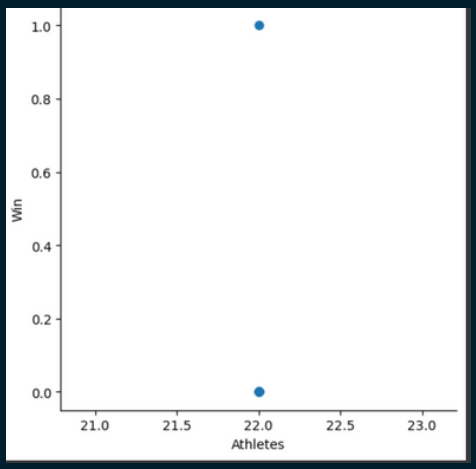
```
import seaborn as sns
sns.lmplot(x = "Year", y ="Win", data = teams, fit_reg = True, ci = None)

<seaborn.axisgrid.FacetGrid at 0x793b858c0220>
```

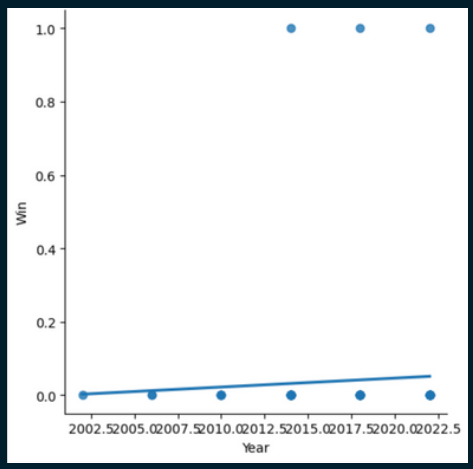


To make this plot the columns Years and Win were chosen because they had a higher correlations.

This means that can be used to make predictions.

PLOT ATHLETES AND WIN COLUMNS





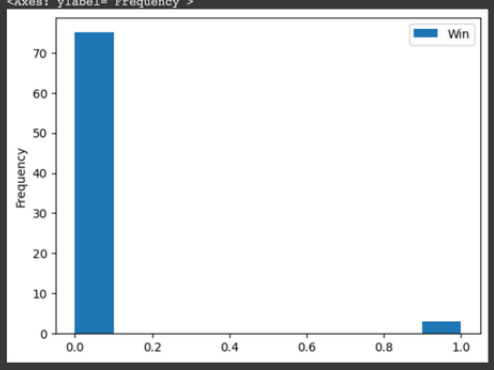
We compare both graphs and came to the conclusion that using the athlete and win parameters would not be adequate to make a prediction since it is not a linear correlation that increases or decreases.c

CODE

WINS HISTOGRAM

```
teams.plot.hist(y='Win')
```

```
<Axes: ylabel='Frequency'>
```



We made a histogram to look at how many countries fall into each bin of number of World Cups won.

In our histogram we can see that more than 70 countries have participated in world cups, and that most of these countries have never won a world cup while less than 10 countries have won even 1 world cup.

CODE

CLEANING THE DATA

Some of the countries have not participated in the selected world cups, so we must locate those missing values and eliminate them.

```
[23] teams[teams.isnull().any(axis=1)]
```

	Team	Year	Athletes	Age	Win	Prev_win
5	Cameroon	2022	NaN	NaN	NaN	NaN
23	Colombia	2022	NaN	NaN	NaN	NaN

```
teams = teams.dropna()
teams
```

	Team	Year	Athletes	Age	Win	Prev_win
0	Brazil	2014	22.0	27.830	0.0	0.0
1	Brazil	2018	22.0	33.312	0.0	0.0
2	Brazil	2022	22.0	28.600	0.0	0.0
3	Cameroon	2014	22.0	35.280	0.0	0.0
4	Cameroon	2018	22.0	21.460	0.0	0.0
...	...	...	...	...	...	...
73	South Korea	2018	22.0	32.390	0.0	0.0
74	South Korea	2022	22.0	28.230	0.0	0.0
75	Russia	2002	22.0	39.330	0.0	0.0
76	Russia	2014	22.0	36.180	0.0	0.0
77	Russia	2018	22.0	33.430	0.0	0.0

CODE

SPLITTING THE DATA

We are going to divide our data depending on its time series.
We will put the years after 2018 in the test model and the years before 2018 in the train model.

We will do this because if we want to predict who will win the 2026 World Cup we don't have access to the data from the 2030 World Cup. We only have data from the past so when we train our machine learning model we want to respect the order of our data and make sure we don't use future data to predict.

We split the data to train our model on the train set and use the other set to evaluate how well the model is working.

```
[ ] # split data
    train = teams[teams['Year'] < 2018].copy()
    test = teams[teams['Year'] >= 2018].copy()

[ ] train.shape

    (34, 6)

[ ] test.shape

    (44, 6)
```

CODE

IMPORTING LINEAR REGRESSION CLASS

Linear regression class is going to enable us to train and make predictions with a linear model

```
[32] # Training the model

     from sklearn.linear_model import LinearRegression
     reg = LinearRegression()
```

Now we will define which columns we will use for the prediction. (Year and Prev_win)

Then what we will define is our target, which is 'Win' since we want to be able to predict which country will win the next World Cup.

And we fit our linear regression model.

```
35] predictors = ['Year', 'Prev_win']
    target = 'Win'

36] # fitting the regresssion model
    reg.fit(train[predictors], train['Win'])
    LinearRegression()

    ▾ LinearRegression
    LinearRegression()
```

CODE

PREDICTIONS

```
[37] predictions = reg.predict(test[predictors])
     predictions

     array([0.05807623, 0.07803993, 0.05807623, 0.05807623, 0.07803993,
            0.05807623, 0.07803993, 0.05807623, 0.07803993, 0.05807623,
            0.07803993, 0.05807623, 0.07803993, 0.05807623, 0.05807623,
            0.07803993, 0.05807623, 0.07803993, 0.05807623, 0.07803993,
            0.05807623, 0.07803993, 0.05807623, 0.07803993, 0.07803993,
            0.05807623, 0.0399274 , 0.05807623, 0.07803993, 0.05807623,
            0.05807623, 0.07803993, 0.05807623, 0.07803993, 0.07803993,
            0.07803993, 0.05807623, 0.07803993, 0.05807623, 0.07803993,
            0.05807623, 0.07803993, 0.05807623])
```

As we can notice the numbers are decimal and you cannot win 0.267 of victory so we will make some changes in the model

```
[39] test.loc[test['predictions'] < 0, 'predictions'] = 0
     test['predictions'] = test['predictions'].round()
```

```
test
```

	Team	Year	Athletes	Age	Win	Prev_win	predictions
1	Brazil	2018	22.0	33.312	0.0	0.0	0.0
2	Brazil	2022	22.0	28.600	0.0	0.0	0.0
4	Cameroon	2018	22.0	21.460	0.0	0.0	0.0
7	Mexico	2018	22.0	33.870	0.0	0.0	0.0

What we did now is that the numbers that were less than zeros were replaced by zeros.
Let the numbers be rounded.

CODE

IMPORT MEAN ABSOLUTE ERROR

```
[41] from sklearn.metrics import mean_absolute_error
     error = mean_absolute_error(test['Win'], test['predictions'])
     error

     0.046511627906976744
```

The mean absolute error we obtained means that on average we were within 0.046512 wins of how many world cups a team actually won

```
[ ] teams.describe()['Win']

count    76.000000
mean      0.039474
std       0.196013
min       0.000000
25%       0.000000
50%       0.000000
75%       0.000000
max       1.000000
Name: Win, dtype: float64
```

The describe method shows us the minimum value in the column, the percentiles and the standard deviation.
If the error was higher than the standard deviation then we would be using predictors that don't give us any information to predict so we would need to change them.

CODE

CHECK OUR PREDICTIONS IN SOME TEAMS

```
[24] test[test['Team'] == 'Mexico']
```

	Team	Year	Athletes	Age	Win	Prev_win	predictions
7	Mexico	2018	22.0	33.87	0.0	0.0	0.0
8	Mexico	2022	22.0	29.08	0.0	0.0	0.0

```
[25] test[test['Team'] == 'Argentina']
```

	Team	Year	Athletes	Age	Win	Prev_win	predictions
49	Argentina	2018	22.0	33.91	0.0	0.0	0.0
50	Argentina	2022	22.0	28.50	1.0	0.0	0.0

CHECK ERRORS

```
errors = (test['Win'] - test['predictions']).abs()
errors
```

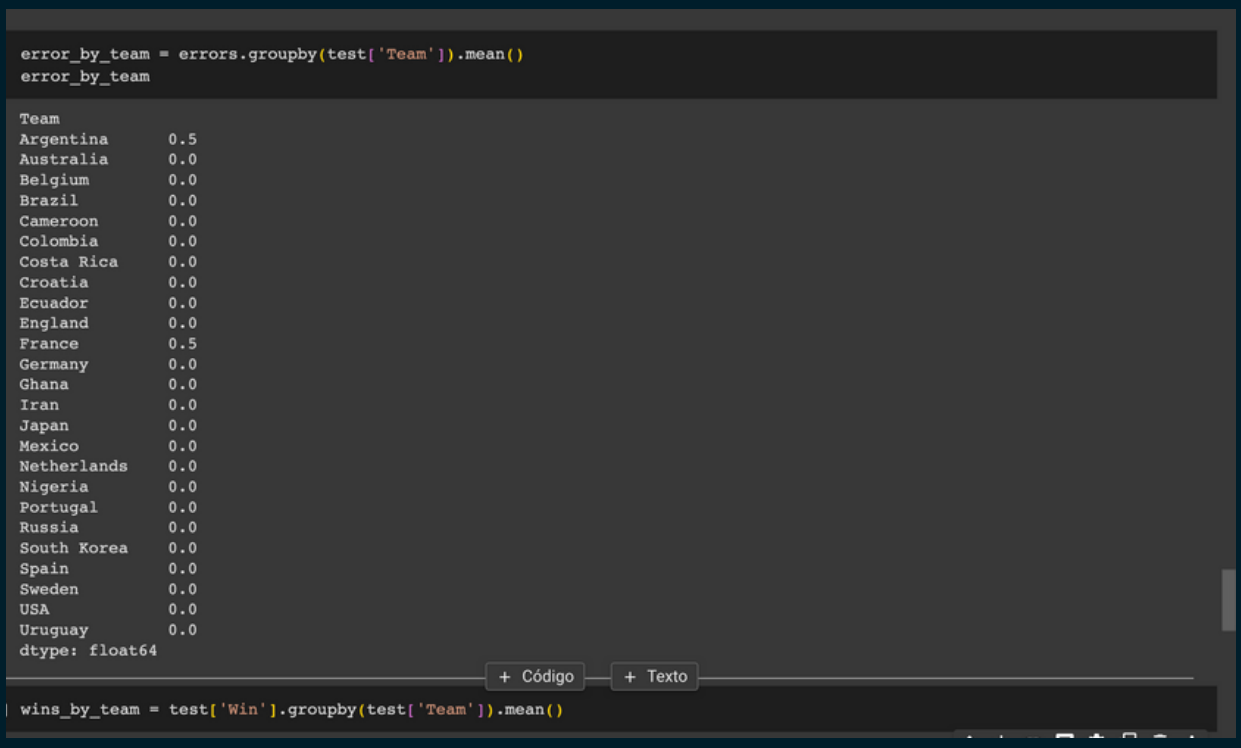
```
1     0.0
2     0.0
4     0.0
7     0.0
8     0.0
10    0.0
11    0.0
13    0.0
14    0.0
16    0.0
17    0.0
19    0.0
20    0.0
```

We look at our errors by country just to see how are we doing on a country by country basis.

CODE

GROUP ERRORS BY TEAM

The group by pandas method will create a separate group for each team and then we will calculate the mean of each group.
And then we do the same process for the Win column.



```
error_by_team = errors.groupby(test['Team']).mean()
error_by_team
```

```
Team
Argentina      0.5
Australia      0.0
Belgium        0.0
Brazil         0.0
Cameroon       0.0
Colombia       0.0
Costa Rica     0.0
Croatia        0.0
Ecuador        0.0
England        0.0
France         0.5
Germany        0.0
Ghana          0.0
Iran           0.0
Japan          0.0
Mexico         0.0
Netherlands    0.0
Nigeria        0.0
Portugal       0.0
Russia         0.0
South Korea    0.0
Spain          0.0
Sweden         0.0
USA            0.0
Uruguay        0.0
dtype: float64
```

+ Código + Texto

```
wins_by_team = test['Win'].groupby(test['Team']).mean()
```

CODE

FIND THE RATIO BETWEEN THE ERRORS

Because most teams win the world championships average is zero and we are dividing by zero so we are getting a missing value.
We will remove all missing values.

```
error_ratio = error_by_team / wins_by_team
error_ratio
```

```
Team
Argentina      1.0
Australia      NaN
Belgium        NaN
Brazil         NaN
Cameroon       NaN
Colombia       NaN
Costa Rica     NaN
Croatia        NaN
Ecuador        NaN
England        NaN
France         1.0
Germany        NaN
Ghana          NaN
Iran           NaN
Japan          NaN
Mexico         NaN
Netherlands    NaN
Nigeria        NaN
Portugal       NaN
Russia         NaN
South Korea    NaN
Spain          NaN
Sweden         NaN
USA            NaN
Uruguay        NaN
dtype: float64
```

```
[ ] error_ratio[~pd.isnull(error_ratio)]
```

```
Team
Argentina    1.0
France       1.0
dtype: float64
```

To prevent infinite values we use the numpy.isfinite method

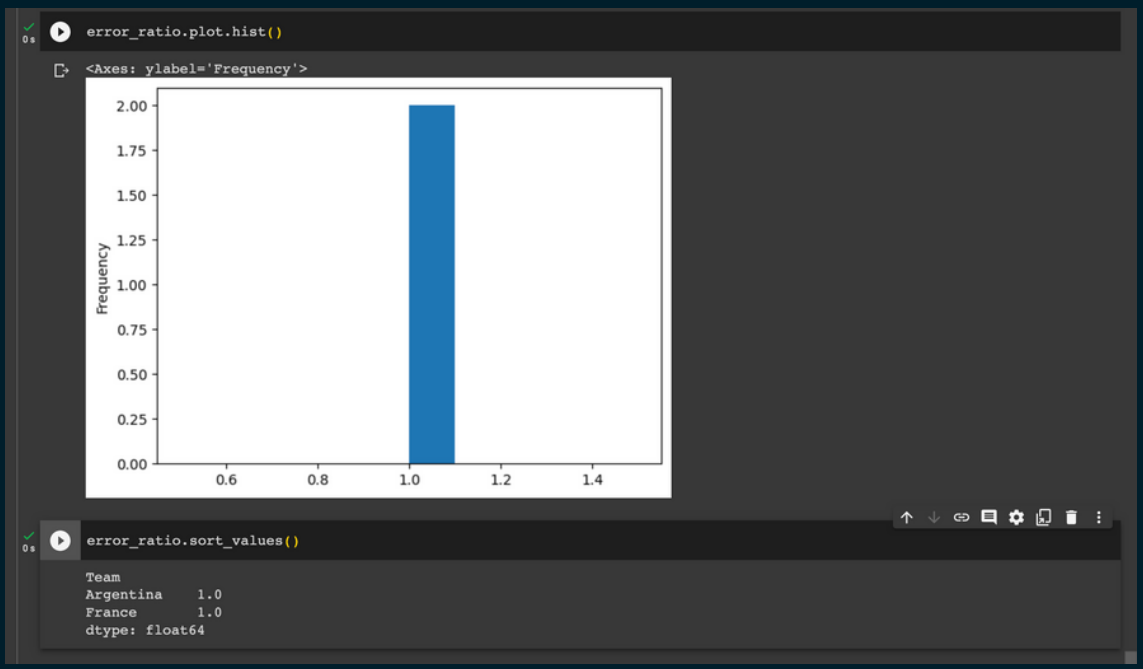
```
[31] import numpy as np
     error_ratio = error_ratio[np.isfinite(error_ratio)]
     error_ratio
```

CODE

WE CREATE A HISTOGRAM

This histogram tells us that an error ratio of 1 means that we are within 50% of the actual win count.

Now that we know that are model is efficient we can sort the values.
This will tell us which countries are most likely to win the 2026 FIFA World Cup.



As results we obtained that both France and Argentina have a very high probability of being the winners of the world cup in 2026.

CONCLUSION

In conclusion, I have discovered that predicting the winner of the next World Cup using machine learning is a very complex and difficult process. We have learned a variety of things and have a solid understanding of the fundamentals, which is crucial to emphasize in the conclusion:

I have realized that soccer is an unpredictable sport by nature, despite machine learning's capacity to evaluate previous data and patterns to create predictions. The outcome is influenced by a wide range of variables, including player skill, team strategy, and match dynamics. Because of this, no forecast can absolutely guarantee the outcomes.

The importance of clean data became evident to me as I worked on the project. I now understand that a machine learning project's performance depends heavily on the quality of the data used. The model must be properly cleaned and prepared before it can identify significant patterns and generate reliable predictions.

Throughout the project's growth, I discovered how important it is to choose the right machine learning model. By comparing different algorithms and strategies, I was able to determine which ones are best suited for our data set and prediction goal.

Understanding the significance of results interpretation when using machine learning to make predictions was an important lesson I gained from this project. I comprehended how the model arrives at its findings and which features most strongly affect the prediction.

I also encountered a number of challenges and restrictions during the project, such as data availability, selecting pertinent features, and processing missing data. By learning how to overcome these challenges, I acquired crucial abilities for future machine learning projects.

In summary, I have learned a great deal about the complexity of the sport, the value of clean data, the selection of the best model, and the interpretation of findings by conducting a machine learning project to predict the winner of the next World Cup. Additionally, I have encountered difficulties and considered the moral implications of AI use. These lessons will encourage me to approach forecasting with humility and an understanding of its inherent limits to strengthen my abilities in subsequent initiatives.

BIBLIOGRAPHY

Plantel de Alemania. (n.d.). ESPN. https://www.espn.com.mx/futbol/equipo/plantel/_/id/481/ger

Plantel de Colombia. (n.d.). ESPN. https://www.espn.com.mx/futbol/equipo/plantel/_/id/208/liga/FIFA.WORLD

Plantel de México. (n.d.). ESPN. https://www.espn.com.mx/futbol/equipo/plantel/_/id/203/mex

Plantel de Camerún. (n.d.). ESPN. https://www.espn.com.mx/futbol/equipo/plantel/_/id/656/liga/FIFA.WORLD

Plantel de Camerún. (n.d.). ESPN. https://www.espn.com.mx/futbol/equipo/plantel/_/id/656/liga/FIFA.WORLD

Plantel de Croacia. (n.d.). ESPN. https://www.espn.com.mx/futbol/equipo/plantel/_/id/477/cro

Plantel de España. (n.d.-b). ESPN. https://www.espn.com.mx/futbol/equipo/plantel/_/id/164/liga/FIFA.WORLD

Plantel de Australia. (n.d.). ESPN. https://www.espn.com.mx/futbol/equipo/plantel/_/id/628/liga/FIFA.WORLD

Plantel de Países Bajos. (n.d.). ESPN. https://www.espn.com.mx/futbol/equipo/plantel/_/id/449/liga/FIFA.WORLD

Plantel de Japón. (n.d.). ESPN. https://www.espn.com.mx/futbol/equipo/plantel/_/id/627/liga/FIFA.WORLD

Plantel de Bélgica. (n.d.). ESPN. https://www.espn.com.mx/futbol/equipo/plantel/_/id/459/liga/FIFA.WORLD

Plantel de Corea del Sur. (n.d.). ESPN. https://www.espn.com.mx/futbol/equipo/plantel/_/id/451/liga/FIFA.WORLD

Plantel de Uruguay. (n.d.). ESPN. https://www.espn.com.mx/futbol/equipo/plantel/_/id/212/uru

Plantel de Italia. (n.d.). ESPN. https://www.espn.com.mx/futbol/equipo/plantel/_/id/162/ita

Plantel de Costa Rica. (n.d.). ESPN. https://www.espn.com.mx/futbol/equipo/plantel/_/id/214/liga/FIFA.WORLD

Plantel de Inglaterra. (n.d.). ESPN. https://www.espn.com.mx/futbol/equipo/plantel/_/id/448/liga/FIFA.WORLD

Plantel de Suiza. (n.d.). ESPN. https://www.espn.com.mx/futbol/equipo/plantel/_/id/475/liga/FIFA.WORLD

Plantel de Ecuador. (n.d.). ESPN. https://www.espn.com.mx/futbol/equipo/plantel/_/id/209/ecu

Plantel de Francia. (n.d.). ESPN. https://www.espn.com.mx/futbol/equipo/plantel/_/id/478/liga/FIFA.WORLD

Plantel de Argentina. (n.d.). ESPN. https://www.espn.com.ar/futbol/equipo/plantel/_/id/202/arg

Plantel de Nigeria. (n.d.). ESPN. https://www.espn.com.pe/futbol/equipo/plantel/_/id/657/liga/FIFA.WORLD

Plantel de Irán. (n.d.). ESPN. https://www.espn.com.mx/futbol/equipo/plantel/_/id/469/liga/FIFA.WORLD

Plantel de Ghana. (n.d.). ESPN. https://espn.com.co/futbol/equipo/plantel/_/id/4469/liga/FIFA.WORLD

Plantel de Estados Unidos. (n.d.). ESPN. https://www.espn.com.mx/futbol/equipo/plantel/_/id/660/liga/FIFA.WORLD

Plantel de Portugal. (n.d.). ESPN. https://www.espn.com.mx/futbol/equipo/plantel/_/id/482/liga/FIFA.WORLD

Plantel de Rusia. (n.d.). - ESPN. https://www.espn.com.uy/futbol/equipo/plantel/_/id/454/liga/FIFA.WORLD

Beers, B. (2023). What is Regression? Definition, Calculation, and Example. Investopedia. https://www.investopedia.com/terms/r/regression.asp

Dataquest. (2022, July 18). Build your first machine learning Project [Full Beginner Walkthrough] [Video]. YouTube. https://www.youtube.com/watch?v=Hr06nSA-qww