Load Packages

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from scipy.stats import f_oneway
from scipy.stats import shapiro
from scipy.stats import levene
from scipy.stats import pearsonr
import statsmodels.api as sm
from sklearn.preprocessing import LabelEncoder
from sklearn.cluster import KMeans
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, confusion_matrix
```

```
---------------------------------------------------------------------------
-----
ModuleNotFoundError                       Traceback (most recent call
last)
Cell In[3], line 1
----> 1 import pandas as pd
      2 import numpy as np
      3 import matplotlib.pyplot as plt

ModuleNotFoundError: No module named 'pandas'
```

Load Dataset

```
data = pd.read_excel('Strikers_performance.xlsx')
data.head()
```

```
   Striker_ID Nationality    Footedness Marital Status  Goals
Scored  \
0           1       Spain   Left-footed             No     17.483571

1           2      France   Left-footed            Yes     14.308678

2           3     Germany   Left-footed             No     18.238443

3           4      France  Right-footed             No     22.615149

4           5      France   Left-footed            Yes     13.829233


     Assists  Shots on Target  Shot Accuracy  Conversion Rate  \
```

```
0  10.778533        34.795488       0.677836         0.166241
1  13.728250        31.472436       0.544881         0.192774
2   3.804297        25.417413       0.518180         0.160379
3   9.688908        20.471443       0.599663         0.184602
4   6.048072        29.887563       0.582982         0.105319

   Dribbling Success  Movement off the Ball  Hold-up Play  Aerial
Duels Won  \
0           0.757061              50.921924     71.806409
15.682532
1           0.796818              61.396150     53.726866
19.843983
2           0.666869              65.863945     60.452227
20.090084
3           0.638776              88.876877     60.511979
22.363152
4           0.591485              75.565531     54.982158
13.165708

   Defensive Contribution  Big Game Performance  Consistency  \
0               30.412215              6.152481     0.820314
1               26.474913              6.093172     0.803321
2               24.164116              3.408714     0.766540
3               44.129989              6.339820     0.611798
4               37.859323              8.465658     0.701638

   Penalty Success Rate  Impact on Team Performance  Off-field Conduct

0              0.922727                    8.570370          11.451388

1              0.678984                    3.444638           8.243689

2              0.843858                    8.429491           9.506835

3              0.662997                    6.532552           8.199653

4              0.906538                    8.414915           6.665333
```

Data Cleaning

Missing values

```
data.isnull().sum()

---------------------------------------------------------------------------
-----
NameError                                 Traceback (most recent call
```

```
last)
Cell In[1], line 1
----> 1 data.isnull().sum()

NameError: name 'data' is not defined
```

```
from sklearn.impute import SimpleImputer

imputer = SimpleImputer(strategy='median')
data[['Movement off the Ball',
      'Big Game Performance',
      'Penalty Success Rate']] = imputer.fit_transform(data[['Movement
off the Ball',
                                                             'Big Game
Performance',
                                                             'Penalty
Success Rate']])
data.isnull().sum()
```

```
Striker_ID                    0
Nationality                   0
Footedness                    0
Marital Status                0
Goals Scored                  0
Assists                       0
Shots on Target               0
Shot Accuracy                 0
Conversion Rate               0
Dribbling Success             0
Movement off the Ball         0
Hold-up Play                  0
Aerial Duels Won              0
Defensive Contribution        0
Big Game Performance          0
Consistency                   0
Penalty Success Rate          0
Impact on Team Performance    0
Off-field Conduct             0
dtype: int64
```

Data Types

```
data.dtypes
```

```
Striker_ID                     int64
Nationality                   object
Footedness                    object
Marital Status                object
Goals Scored                 float64
Assists                      float64
```

```
Shots on Target                float64
Shot Accuracy                  float64
Conversion Rate                float64
Dribbling Success              float64
Movement off the Ball          float64
Hold-up Play                   float64
Aerial Duels Won               float64
Defensive Contribution         float64
Big Game Performance           float64
Consistency                    float64
Penalty Success Rate           float64
Impact on Team Performance     float64
Off-field Conduct              float64
dtype: object
```

```
variables = ['Goals Scored', 'Assists',
             'Shots on Target',
             'Movement off the Ball',
             'Hold-up Play',
             'Aerial Duels Won',
             'Defensive Contribution',
             'Big Game Performance',
             'Impact on Team Performance',
             'Off-field Conduct']

for var in variables:
    data[var] = data[var].astype('int')

data.dtypes
```

```
Striker_ID                      int64
Nationality                    object
Footedness                     object
Marital Status                 object
Goals Scored                    int64
Assists                         int64
Shots on Target                 int64
Shot Accuracy                  float64
Conversion Rate                float64
Dribbling Success              float64
Movement off the Ball           int64
Hold-up Play                    int64
Aerial Duels Won                int64
Defensive Contribution          int64
Big Game Performance            int64
Consistency                    float64
Penalty Success Rate           float64
Impact on Team Performance      int64
Off-field Conduct               int64
dtype: object
```

```
data.head()
```

```
   Striker_ID Nationality    Footedness Marital Status  Goals Scored
Assists  \
0           1       Spain   Left-footed             No            17
10
1           2      France   Left-footed            Yes            14
13
2           3     Germany   Left-footed             No            18
3
3           4      France  Right-footed             No            22
9
4           5      France   Left-footed            Yes            13
6

   Shots on Target  Shot Accuracy  Conversion Rate  Dribbling Success
\
0               34       0.677836         0.166241           0.757061

1               31       0.544881         0.192774           0.796818

2               25       0.518180         0.160379           0.666869

3               20       0.599663         0.184602           0.638776

4               29       0.582982         0.105319           0.591485


   Movement off the Ball  Hold-up Play  Aerial Duels Won  \
0                     50            71                15
1                     61            53                19
2                     65            60                20
3                     88            60                22
4                     75            54                13

   Defensive Contribution  Big Game Performance  Consistency  \
0                      30                     6     0.820314
1                      26                     6     0.803321
2                      24                     3     0.766540
3                      44                     6     0.611798
4                      37                     8     0.701638

   Penalty Success Rate  Impact on Team Performance  Off-field Conduct

0              0.922727                           8                 11

1              0.678984                           3                  8

2              0.843858                           8                  9

3              0.662997                           6                  8
```

```
4               0.906538                           8                   6
```

Exploratory Data Analysis

Perform descriptive analysis

```
round(data.describe(), 2)
```

```
       Striker_ID  Goals Scored  Assists  Shots on Target  Shot
Accuracy  \
count      500.00        500.00   500.00           500.00
500.00
mean       250.50         14.52     7.60            25.26
0.60
std        144.48          4.91     2.96             7.08
0.10
min          1.00          0.00     0.00             4.00
0.31
25%        125.75         11.00     6.00            20.00
0.54
50%        250.50         15.00     8.00            25.00
0.60
75%        375.25         18.00     9.00            30.00
0.67
max        500.00         34.00    15.00            43.00
0.92

       Conversion Rate  Dribbling Success  Movement off the Ball  \
count           500.00             500.00                 500.00
mean              0.20               0.70                  69.28
std               0.05               0.10                  10.33
min               0.05               0.40                  40.00
25%               0.17               0.64                  62.00
50%               0.20               0.70                  69.00
75%               0.23               0.76                  76.00
max               0.36               1.00                  98.00

       Hold-up Play  Aerial Duels Won  Defensive Contribution  \
count        500.00            500.00                  500.00
mean          59.33             19.04                   39.47
std           10.15              4.95                    9.90
min           35.00              4.00                    8.00
25%           52.00             16.00                   33.00
50%           60.00             19.00                   39.50
75%           66.00             22.00                   46.00
max           92.00             34.00                   71.00
```

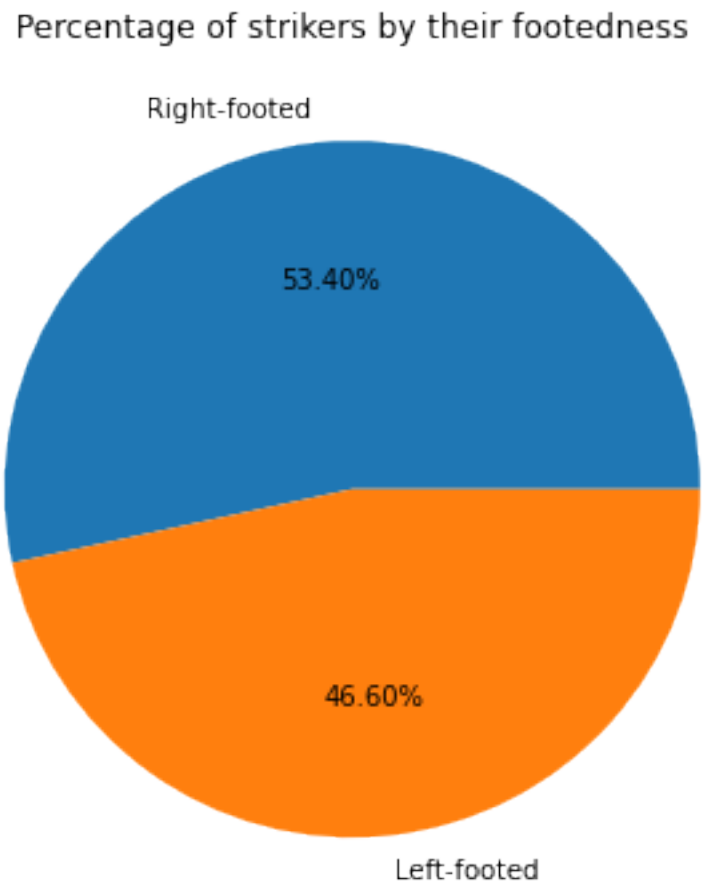
```
       Big Game Performance  Consistency  Penalty Success Rate  \
count                500.00       500.00                500.00
mean                   6.43         0.74                  0.80
std                    1.98         0.10                  0.10
min                    2.00         0.46                  0.53
25%                    5.00         0.68                  0.72
50%                    6.00         0.75                  0.80
75%                    8.00         0.81                  0.87
max                   12.00         1.00                  1.00

       Impact on Team Performance  Off-field Conduct
count                      500.00             500.00
mean                         6.52               7.57
std                          2.11               2.09
min                          0.00               0.00
25%                          5.00               6.00
50%                          6.50               8.00
75%                          8.00               9.00
max                         13.00              13.00
```

Perform percentage analysis

```
freq_Footedness = data['Footedness'].value_counts()
perc_Footedness = freq_Footedness/len(data['Footedness'])*100
perc_Footedness
```

```
Right-footed    53.4
Left-footed     46.6
Name: Footedness, dtype: float64
```

```
plt.figure(figsize=(12, 6))
perc_Footedness.plot(kind='pie', autopct='%1.2f%%')
plt.title('Percentage of strikers by their footedness')
plt.ylabel('')
plt.show()
```



Which nationality strikers have the highest average number of goals scored?

```
goals_by_nationality = data.groupby('Nationality')['Goals
Scored'].mean().sort_values(ascending=False)
round(goals_by_nationality)
```

```
Nationality
Brazil     15.0
Spain      15.0
France     14.0
Germany    14.0
England    14.0
Name: Goals Scored, dtype: float64
```

What is the average conversion rate for players based on their footedness?

```
conversion_rate_by_footedness = data.groupby('Footedness')['Conversion
Rate'].mean()
conversion_rate_by_footedness
```

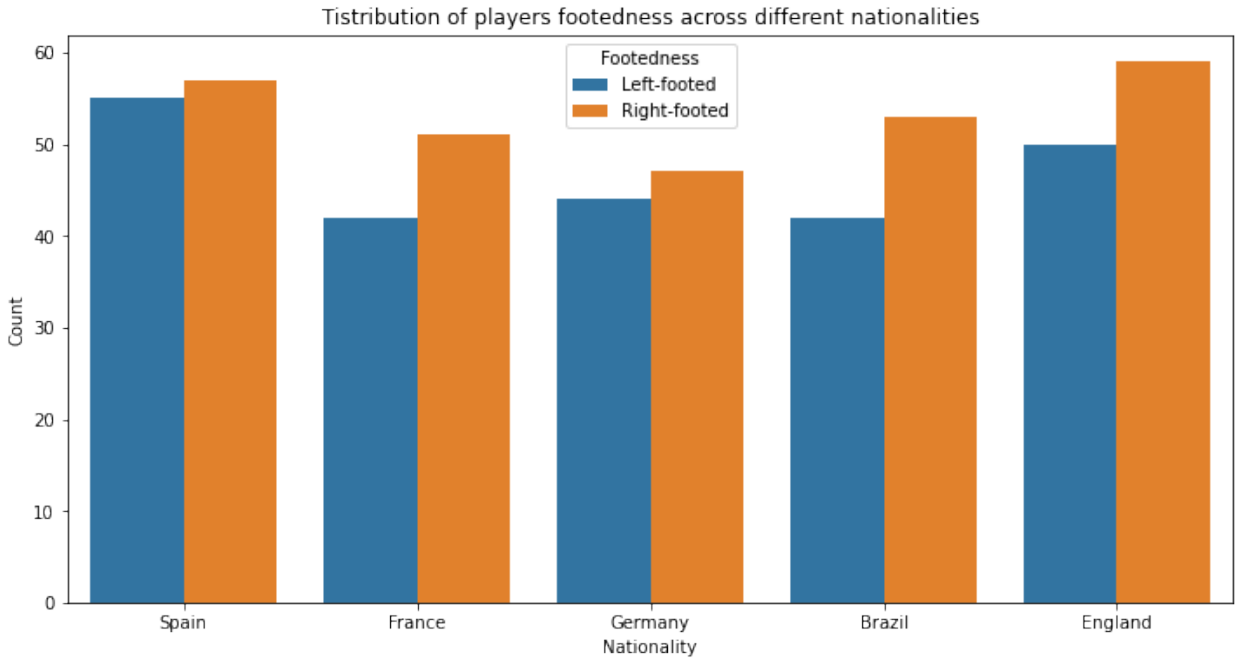
```
Footedness
Left-footed     0.198086
Right-footed    0.200592
Name: Conversion Rate, dtype: float64
```

What is the distribution of players' footedness across different nationalities?

```
footedness_by_nationality = pd.crosstab(data['Nationality'],
data['Footedness'])
footedness_by_nationality
```

Footedness	Left-footed	Right-footed
Nationality		
Brazil	42	53
England	50	59
France	42	51
Germany	44	47
Spain	55	57

```
plt.figure(figsize=(12, 6))
sns.countplot(x='Nationality', hue='Footedness', data=data)
plt.title('Tistribution of players footedness across different
nationalities')
plt.xlabel('Nationality')
plt.ylabel('Count')
plt.show()
```



Create a correlation matrix with a heatmap

```
num_variables = data.select_dtypes(include = ['number']).columns

correl_matrix = round(data[num_variables].corr(), 3)
correl_matrix
```

```
                           Striker_ID  Goals Scored  Assists  \
Striker_ID                      1.000         0.025    0.066
Goals Scored                    0.025         1.000   -0.071
Assists                         0.066        -0.071    1.000
Shots on Target                 0.003        -0.059    0.072
Shot Accuracy                   0.018         0.066   -0.020
Conversion Rate                 0.047        -0.011    0.028
Dribbling Success              -0.031         0.034    0.050
Movement off the Ball           0.060         0.016    0.011
Hold-up Play                   -0.036         0.003   -0.046
Aerial Duels Won                0.016        -0.047    0.001
Defensive Contribution          0.012         0.014   -0.011
Big Game Performance           -0.052        -0.004    0.026
Consistency                    -0.017         0.045   -0.007
Penalty Success Rate           -0.006        -0.021   -0.081
Impact on Team Performance     -0.027         0.103   -0.022
Off-field Conduct               0.103        -0.053   -0.015

                           Shots on Target  Shot Accuracy  Conversion
Rate  \
Striker_ID                           0.003          0.018
0.047
Goals Scored                        -0.059          0.066           -
0.011
Assists                              0.072         -0.020
0.028
Shots on Target                      1.000         -0.021           -
0.044
Shot Accuracy                       -0.021          1.000           -
0.088
Conversion Rate                     -0.044         -0.088
1.000
Dribbling Success                    0.010          0.023           -
0.020
Movement off the Ball               -0.062         -0.018
0.038
Hold-up Play                        -0.102         -0.044
0.059
Aerial Duels Won                    -0.068          0.012
0.059
Defensive Contribution               0.011          0.027           -
0.002
Big Game Performance                 0.039         -0.019
```

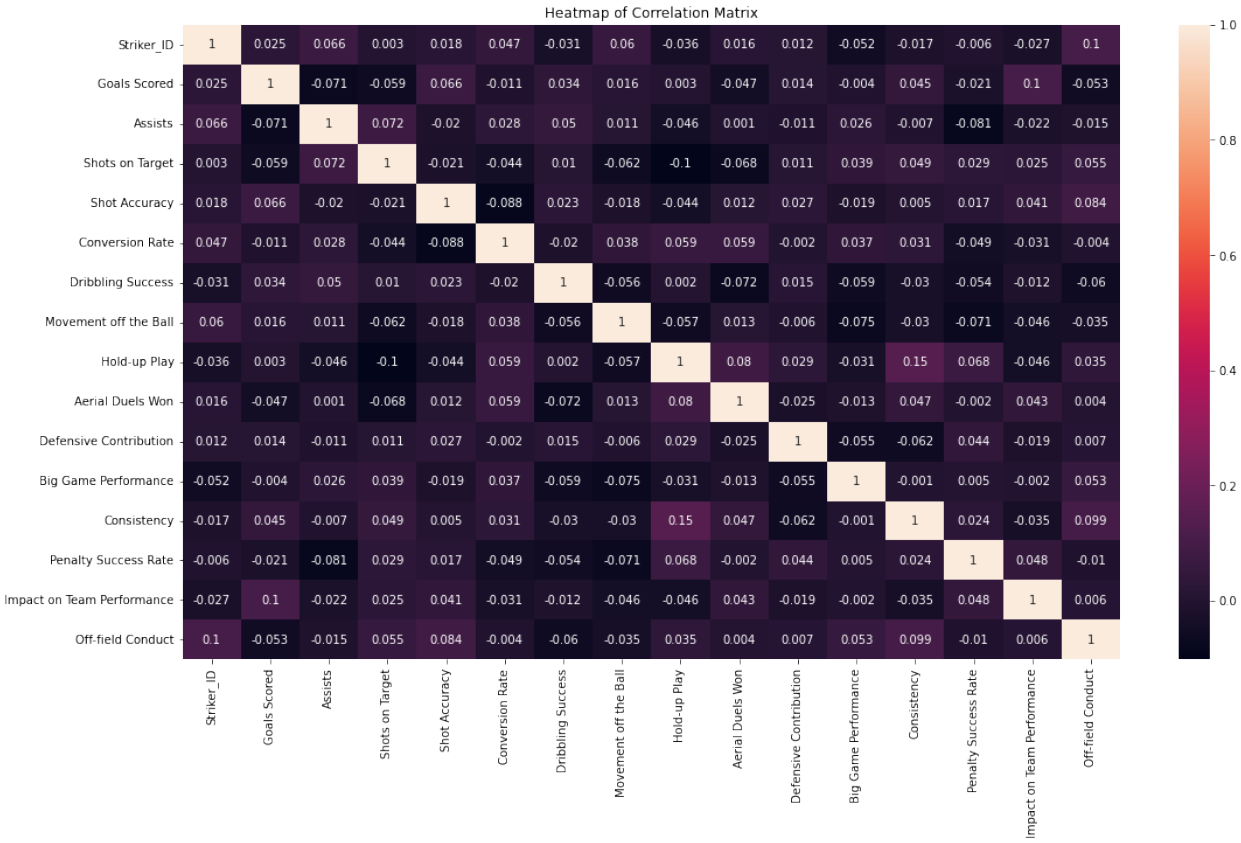
```
0.037
Consistency                          0.049          0.005
0.031
Penalty Success Rate                 0.029          0.017          -
0.049
Impact on Team Performance           0.025          0.041          -
0.031
Off-field Conduct                    0.055          0.084          -
0.004

                             Dribbling Success  Movement off the
Ball  \
Striker_ID                           -0.031                  0.060

Goals Scored                          0.034                  0.016

Assists                               0.050                  0.011

Shots on Target                       0.010                 -0.062

Shot Accuracy                         0.023                 -0.018

Conversion Rate                      -0.020                  0.038

Dribbling Success                     1.000                 -0.056

Movement off the Ball                -0.056                  1.000

Hold-up Play                          0.002                 -0.057

Aerial Duels Won                     -0.072                  0.013

Defensive Contribution                0.015                 -0.006

Big Game Performance                 -0.059                 -0.075

Consistency                          -0.030                 -0.030

Penalty Success Rate                 -0.054                 -0.071

Impact on Team Performance           -0.012                 -0.046

Off-field Conduct                    -0.060                 -0.035


                             Hold-up Play  Aerial Duels Won  \
Striker_ID                         -0.036             0.016
Goals Scored                        0.003            -0.047
Assists                            -0.046             0.001
Shots on Target                    -0.102            -0.068
Shot Accuracy                      -0.044             0.012
```

```
Conversion Rate                    0.059             0.059
Dribbling Success                  0.002            -0.072
Movement off the Ball             -0.057             0.013
Hold-up Play                       1.000             0.080
Aerial Duels Won                   0.080             1.000
Defensive Contribution             0.029            -0.025
Big Game Performance              -0.031            -0.013
Consistency                        0.147             0.047
Penalty Success Rate               0.068            -0.002
Impact on Team Performance        -0.046             0.043
Off-field Conduct                  0.035             0.004

                           Defensive Contribution  Big Game
Performance  \
Striker_ID                                  0.012                -
0.052
Goals Scored                                0.014                -
0.004
Assists                                    -0.011
0.026
Shots on Target                             0.011
0.039
Shot Accuracy                               0.027                -
0.019
Conversion Rate                            -0.002
0.037
Dribbling Success                           0.015                -
0.059
Movement off the Ball                      -0.006                -
0.075
Hold-up Play                                0.029                -
0.031
Aerial Duels Won                           -0.025                -
0.013
Defensive Contribution                      1.000                -
0.055
Big Game Performance                       -0.055
1.000
Consistency                                -0.062                -
0.001
Penalty Success Rate                        0.044
0.005
Impact on Team Performance                 -0.019                -
0.002
Off-field Conduct                           0.007
0.053

                           Consistency  Penalty Success Rate  \
Striker_ID                      -0.017                -0.006
```

Goals Scored	0.045	-0.021
Assists	-0.007	-0.081
Shots on Target	0.049	0.029
Shot Accuracy	0.005	0.017
Conversion Rate	0.031	-0.049
Dribbling Success	-0.030	-0.054
Movement off the Ball	-0.030	-0.071
Hold-up Play	0.147	0.068
Aerial Duels Won	0.047	-0.002
Defensive Contribution	-0.062	0.044
Big Game Performance	-0.001	0.005
Consistency	1.000	0.024
Penalty Success Rate	0.024	1.000
Impact on Team Performance	-0.035	0.048
Off-field Conduct	0.099	-0.010

	Impact on Team Performance	Off-field Conduct
Striker_ID	-0.027	0.103
Goals Scored	0.103	-0.053
Assists	-0.022	-0.015
Shots on Target	0.025	0.055
Shot Accuracy	0.041	0.084
Conversion Rate	-0.031	-0.004
Dribbling Success	-0.012	-0.060
Movement off the Ball	-0.046	-0.035
Hold-up Play	-0.046	0.035
Aerial Duels Won	0.043	0.004
Defensive Contribution	-0.019	0.007
Big Game Performance	-0.002	0.053
Consistency	-0.035	0.099
Penalty Success Rate	0.048	-0.010
Impact on Team Performance	1.000	0.006

```
Off-field Conduct                                    0.006
1.000

plt.figure(figsize=(18, 10))
sns.heatmap(correl_matrix, annot=True)
plt.title('Heatmap of Correlation Matrix')
plt.show()
```



Statistical Test

Find whether there is any significant difference in consistency rates among strikers from various nationality

```
# Normality test
stat, p_value = shapiro(data['Consistency'])
print('P value: ', round(p_value, 3))

P value:  0.451

# Filtering data
Spain = data.query('Nationality == "Spain"')['Consistency']
France = data.query('Nationality == "France"')['Consistency']
```

```
Germany = data.query('Nationality == "Germany"')['Consistency']
Brazil = data.query('Nationality == "Brazil"')['Consistency']
England = data.query('Nationality == "England"')['Consistency']

# Levene test for statistics
stats, p_value = levene(Spain, France, Germany, Brazil, England)
print("P value: ", round(p_value, 3))
```

```
P value:  0.808
```

```
# One way ANOVA

Test_stat, p_value = f_oneway(Spain, France, Germany, Brazil, England)
print("P value: ", round(p_value, 2))
```

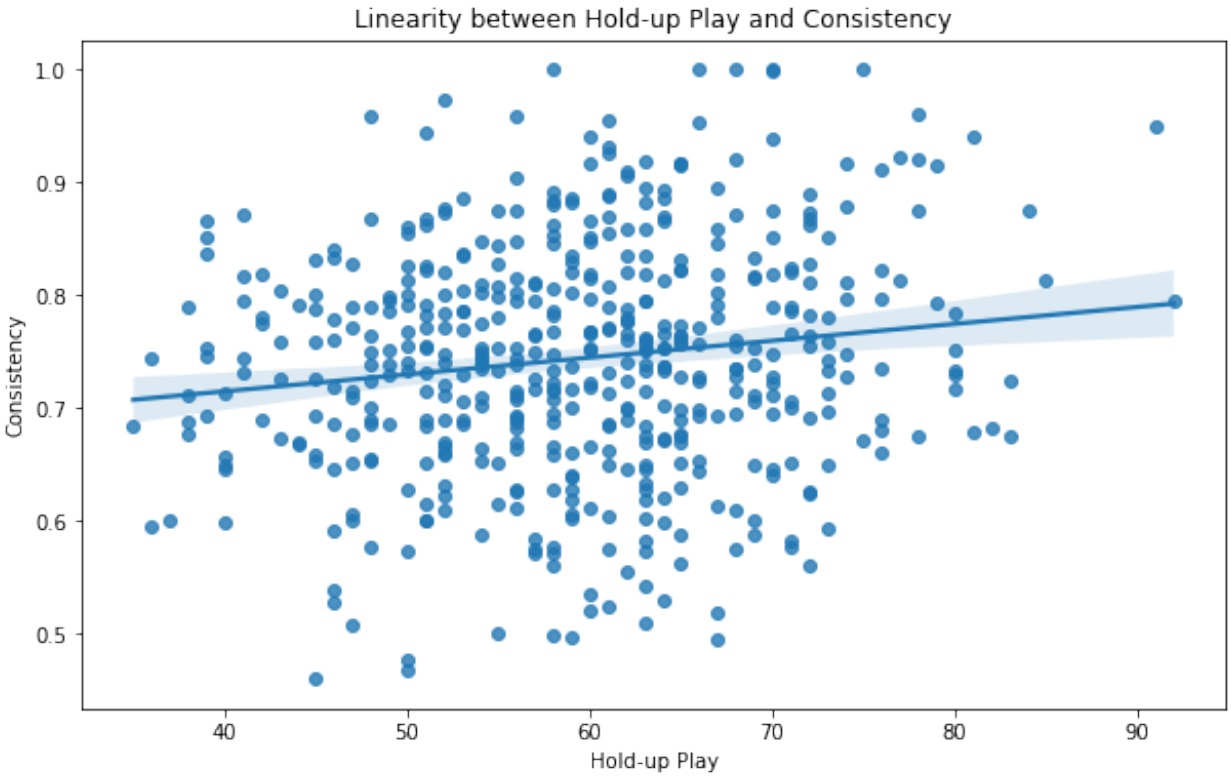
```
P value:  0.19
```

Check if there is any significant correlation between strikers' Hold-up play and consistency rate

```
# Normality test
stat, p_value = shapiro(data['Hold-up Play'])
print('P value: ', round(p_value, 3))
```

```
P value:  0.151
```

```
# Linearity test
plt.figure(figsize = (10, 6))
sns.regplot(x = 'Hold-up Play', y = 'Consistency', data = data)
plt.title('Linearity between Hold-up Play and Consistency')
plt.xlabel('Hold-up Play')
plt.ylabel('Consistency')
plt.show()
```




```
# Pearson correlation
HU_play = data['Hold-up Play']
Consistency = data['Consistency']

corr, p_value = pearsonr(HU_play, Consistency)
print("Correlation coefficient: ", round(corr, 3))
print("P value: ", round(p_value, 3))
```

```
Correlation coefficient:  0.147
P value:  0.001
```

Check if strikers' hold-up play significantly influences their consistency rate

```
x = data['Hold-up Play']
y = data['Consistency']

x_constant = sm.add_constant(x)
model = sm.OLS(y, x_constant).fit()

print(model.summary())
```

```
                            OLS Regression Results

==============================================================================
========
```

```
Dep. Variable:            Consistency   R-squared:
0.021
Model:                            OLS   Adj. R-squared:
0.020
Method:                 Least Squares   F-statistic:
10.93
Date:                Thu, 18 Apr 2024   Prob (F-statistic):
0.00101
Time:                        17:19:02   Log-Likelihood:
429.97
No. Observations:                 500   AIC:
-855.9
Df Residuals:                     498   BIC:
-847.5
Df Model:                           1

Covariance Type:            nonrobust

==============================================================================
                 coef    std err          t      P>|t|      [0.025
0.975]
------------------------------------------------------------------------------
const          0.6548      0.027     24.031      0.000       0.601
0.708
Hold-up Play   0.0015      0.000      3.306      0.001       0.001
0.002
==============================================================================
Omnibus:                        1.708   Durbin-Watson:
2.135
Prob(Omnibus):                  0.426   Jarque-Bera (JB):
1.744
Skew:                          -0.100   Prob(JB):
0.418
Kurtosis:                       2.791   Cond. No.
358.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is
correctly specified.
```

Feature Engineering

Create a new feature - Total contribution score

```
data['Total contribution score'] = (data['Goals Scored'] +
data['Assists'] + data['Shots on Target'] + data['Dribbling Success']
+ data['Aerial Duels Won'] + data['Defensive Contribution'] +
data['Big Game Performance'] + data['Consistency'])
data.head()
```

```
   Striker_ID Nationality    Footedness Marital Status  Goals Scored
Assists  \
0           1       Spain   Left-footed             No            17
10
1           2      France   Left-footed            Yes            14
13
2           3     Germany   Left-footed             No            18
3
3           4      France  Right-footed             No            22
9
4           5      France   Left-footed            Yes            13
6

   Shots on Target  Shot Accuracy  Conversion Rate  Dribbling Success
\
0               34       0.677836         0.166241           0.757061

1               31       0.544881         0.192774           0.796818

2               25       0.518180         0.160379           0.666869

3               20       0.599663         0.184602           0.638776

4               29       0.582982         0.105319           0.591485


   Movement off the Ball  Hold-up Play  Aerial Duels Won  \
0                     50            71                15
1                     61            53                19
2                     65            60                20
3                     88            60                22
4                     75            54                13

   Defensive Contribution  Big Game Performance  Consistency  \
0                      30                     6     0.820314
1                      26                     6     0.803321
2                      24                     3     0.766540
3                      44                     6     0.611798
4                      37                     8     0.701638
```

```
   Penalty Success Rate  Impact on Team Performance  Off-field Conduct
\
0              0.922727                           8                 11

1              0.678984                           3                  8

2              0.843858                           8                  9

3              0.662997                           6                  8

4              0.906538                           8                  6


   Total contribution score
0                113.577376
1                110.600139
2                 94.433410
3                124.250575
4                107.293123
```

Encode the Footedness and marital status by LabelEncoder

```
encoder = LabelEncoder()
data['Footedness'] = encoder.fit_transform(data['Footedness'])
data['Marital Status'] = encoder.fit_transform(data['Marital Status'])
data.head()
```

```
   Striker_ID Nationality  Footedness  Marital Status  Goals Scored
Assists  \
0           1       Spain           0               0            17
10
1           2      France           0               1            14
13
2           3     Germany           0               0            18
3
3           4      France           1               0            22
9
4           5      France           0               1            13
6

   Shots on Target  Shot Accuracy  Conversion Rate  Dribbling Success
\
0               34       0.677836         0.166241           0.757061

1               31       0.544881         0.192774           0.796818

2               25       0.518180         0.160379           0.666869

3               20       0.599663         0.184602           0.638776

4               29       0.582982         0.105319           0.591485
```

```
   Movement off the Ball  Hold-up Play  Aerial Duels Won  \
0                     50            71                15
1                     61            53                19
2                     65            60                20
3                     88            60                22
4                     75            54                13

   Defensive Contribution  Big Game Performance  Consistency  \
0                      30                     6     0.820314
1                      26                     6     0.803321
2                      24                     3     0.766540
3                      44                     6     0.611798
4                      37                     8     0.701638

   Penalty Success Rate  Impact on Team Performance  Off-field Conduct
\
0              0.922727                           8                 11

1              0.678984                           3                  8

2              0.843858                           8                  9

3              0.662997                           6                  8

4              0.906538                           8                  6


   Total contribution score
0                113.577376
1                110.600139
2                 94.433410
3                124.250575
4                107.293123
```

Create the dummies for Nationality and add with the data

```
dummies = pd.get_dummies(data['Nationality'])
processed_df = pd.concat([data, dummies], axis = 1)
processed_df = processed_df.drop('Nationality', axis = 1)
processed_df.head()

   Striker_ID  Footedness  Marital Status  Goals Scored  Assists  \
0           1           0               0            17       10
1           2           0               1            14       13
2           3           0               0            18        3
3           4           1               0            22        9
4           5           0               1            13        6

   Shots on Target  Shot Accuracy  Conversion Rate  Dribbling Success
```

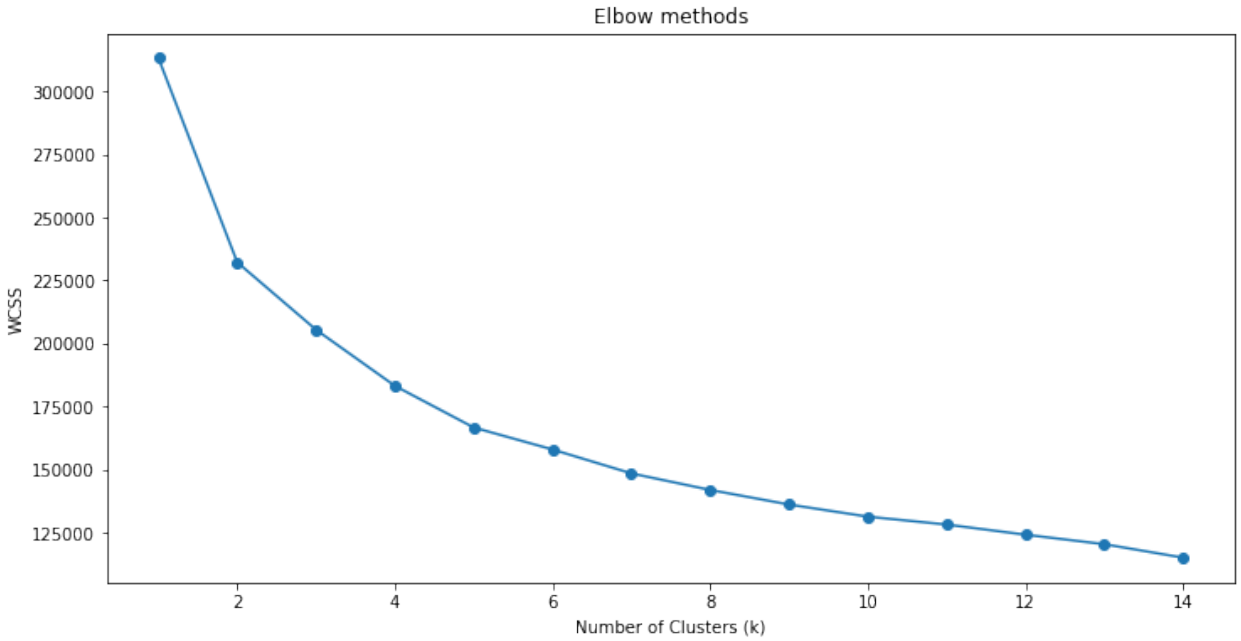
```
\
0                34       0.677836          0.166241            0.757061

1                31       0.544881          0.192774            0.796818

2                25       0.518180          0.160379            0.666869

3                20       0.599663          0.184602            0.638776

4                29       0.582982          0.105319            0.591485


   Movement off the Ball  ...  Consistency  Penalty Success Rate  \
0                     50  ...     0.820314              0.922727
1                     61  ...     0.803321              0.678984
2                     65  ...     0.766540              0.843858
3                     88  ...     0.611798              0.662997
4                     75  ...     0.701638              0.906538

   Impact on Team Performance  Off-field Conduct  Total contribution
score  \
0                           8                 11
113.577376
1                           3                  8
110.600139
2                           8                  9
94.433410
3                           6                  8
124.250575
4                           8                  6
107.293123

   Brazil  England  France  Germany  Spain
0       0        0       0        0      1
1       0        0       1        0      0
2       0        0       0        1      0
3       0        0       1        0      0
4       0        0       1        0      0

[5 rows x 24 columns]
```

Cluster Analysis

Perform KMeans clsutering

```
# Selecting features
x = processed_df.drop('Striker_ID', axis = 1)
```

```
# Calculating WCSS score
wcss = []

for i in range(1, 15):
    kmeans = KMeans(n_clusters = i, init = 'k-means++')
    kmeans.fit(x)
    wcss_score = kmeans.inertia_
    wcss.append(wcss_score)

# Plotting elbow chart
plt.figure(figsize = (12, 6))
plt.plot(range(1, 15), wcss, marker = 'o')
plt.title('Elbow methods')
plt.xlabel('Number of Clusters (k)')
plt.ylabel('WCSS')
plt.show()
```




```
# Building KMeans with k = 2
final_km = KMeans(n_clusters = 2)
final_km.fit(x)

# Generating labels
labels = final_km.labels_
labels
```

```
array([1, 1, 1, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 1, 1, 0, 1, 1, 0,
0,
       1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 1, 0, 0, 0, 1,
1,
       1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 1, 0, 1, 1, 0,
```

```
1,
       1, 1, 0, 0, 1, 0, 0, 1, 1, 1, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0,
0,
       1, 1, 1, 0, 0, 1, 0, 1, 0, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 1,
0,
       1, 1, 0, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1,
0,
       0, 0, 1, 1, 0, 1, 1, 1, 0, 0, 1, 0, 0, 0, 1, 1, 1, 1, 0, 0, 1,
1,
       0, 0, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1, 0,
1,
       1, 0, 0, 0, 0, 1, 0, 1, 1, 0, 1, 1, 1, 1, 1, 0, 0, 1, 1, 0, 0,
0,
       1, 0, 1, 1, 0, 0, 0, 1, 0, 1, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0,
0,
       0, 0, 0, 1, 1, 1, 0, 1, 0, 0, 1, 1, 1, 0, 0, 0, 1, 0, 1, 0, 1,
0,
       1, 0, 0, 1, 0, 1, 0, 1, 0, 0, 1, 1, 1, 0, 1, 0, 1, 0, 1, 0, 0,
1,
       1, 1, 1, 1, 1, 0, 0, 1, 0, 1, 1, 0, 0, 0, 1, 1, 0, 1, 1, 1, 0,
1,
       1, 0, 1, 1, 0, 0, 1, 0, 0, 1, 0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1,
0,
       0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 1, 0, 1, 0, 1, 1, 0, 0, 0,
1,
       0, 1, 0, 0, 1, 1, 1, 1, 0, 1, 1, 0, 0, 0, 0, 0, 1, 1, 1, 0, 1,
0,
       0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 1, 1, 0, 1, 0, 1, 1, 1, 0, 0, 1,
1,
       0, 1, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1,
1,
       1, 1, 0, 0, 1, 1, 1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0,
1,
       0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0,
1,
       0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0,
1,
       0, 0, 1, 1, 0, 0, 1, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 1, 1, 0, 0,
1,
       0, 0, 1, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1], dtype=int32)
```

```
# Adding labels
processed_df['Clusters'] = labels
processed_df.head()
```

```
   Striker_ID  Footedness  Marital Status  Goals Scored  Assists  \
0           1           0               0            17       10
1           2           0               1            14       13
2           3           0               0            18        3
3           4           1               0            22        9
```

```
4           5           0                1            13        6

   Shots on Target  Shot Accuracy  Conversion Rate  Dribbling Success
\
0               34       0.677836         0.166241           0.757061

1               31       0.544881         0.192774           0.796818

2               25       0.518180         0.160379           0.666869

3               20       0.599663         0.184602           0.638776

4               29       0.582982         0.105319           0.591485


   Movement off the Ball  ...  Penalty Success Rate  \
0                     50  ...              0.922727
1                     61  ...              0.678984
2                     65  ...              0.843858
3                     88  ...              0.662997
4                     75  ...              0.906538

   Impact on Team Performance  Off-field Conduct  Total contribution
score  \
0                           8                 11
113.577376
1                           3                  8
110.600139
2                           8                  9
94.433410
3                           6                  8
124.250575
4                           8                  6
107.293123

   Brazil  England  France  Germany  Spain  Clusters
0       0        0       0        0      1         1
1       0        0       1        0      0         1
2       0        0       0        1      0         1
3       0        0       1        0      0         0
4       0        0       1        0      0         1

[5 rows x 25 columns]
```

```python
# Checking clusters
round(processed_df.groupby('Clusters')['Total contribution
score'].mean(), 2)
```

```
Clusters
0    123.39
```

```
1    101.90
Name: Total contribution score, dtype: float64

# Assigning meaningfull names
mapping = {0:'Best strikers', 1:'Regular strikers'}
processed_df['Strikers types'] = processed_df['Clusters'].map(mapping)

# Deleting the Clusters variable
processed_df = processed_df.drop('Clusters', axis = 1)
processed_df.head()
```

	Striker_ID	Footedness	Marital Status	Goals Scored	Assists	\
0	1	0	0	17	10	
1	2	0	1	14	13	
2	3	0	0	18	3	
3	4	1	0	22	9	
4	5	0	1	13	6	

	Shots on Target	Shot Accuracy	Conversion Rate	Dribbling Success	\
0	34	0.677836	0.166241	0.757061	
1	31	0.544881	0.192774	0.796818	
2	25	0.518180	0.160379	0.666869	
3	20	0.599663	0.184602	0.638776	
4	29	0.582982	0.105319	0.591485	

	Movement off the Ball	...	Penalty Success Rate	\
0	50	...	0.922727	
1	61	...	0.678984	
2	65	...	0.843858	
3	88	...	0.662997	
4	75	...	0.906538	

	Impact on Team Performance	Off-field Conduct	Total contribution score	\
0	8	11	113.577376	
1	3	8	110.600139	
2	8	9	94.433410	
3	6	8	124.250575	
4	8	6	107.293123	

```
   Brazil  England  France  Germany  Spain    Strikers types
0       0        0       0        0      1  Regular strikers
1       0        0       1        0      0  Regular strikers
2       0        0       0        1      0  Regular strikers
3       0        0       1        0      0     Best strikers
4       0        0       1        0      0  Regular strikers

[5 rows x 25 columns]
```

Data Preprocessing for ML

New feature mapping

```
mapping = {'Best strikers':1, 'Regular strikers': 0}
processed_df['Strikers types'] = processed_df['Strikers
types'].map(mapping)
processed_df.head()
```

```
   Striker_ID  Footedness  Marital Status  Goals Scored  Assists  \
0           1           0               0            17       10
1           2           0               1            14       13
2           3           0               0            18        3
3           4           1               0            22        9
4           5           0               1            13        6

   Shots on Target  Shot Accuracy  Conversion Rate  Dribbling Success
\
0               34       0.677836         0.166241           0.757061

1               31       0.544881         0.192774           0.796818

2               25       0.518180         0.160379           0.666869

3               20       0.599663         0.184602           0.638776

4               29       0.582982         0.105319           0.591485


   Movement off the Ball  ...  Penalty Success Rate  \
0                     50  ...              0.922727
1                     61  ...              0.678984
2                     65  ...              0.843858
3                     88  ...              0.662997
4                     75  ...              0.906538

   Impact on Team Performance  Off-field Conduct  Total contribution
score  \
0                           8                 11
```

```
113.577376
1                              3                  8
110.600139
2                              8                  9
94.433410
3                              6                  8
124.250575
4                              8                  6
107.293123

   Brazil  England  France  Germany  Spain  Strikers types
0       0        0       0        0      1               0
1       0        0       1        0      0               0
2       0        0       0        1      0               0
3       0        0       1        0      0               1
4       0        0       1        0      0               0

[5 rows x 25 columns]
```

Selecting features

```
x = processed_df.drop(['Striker_ID', 'Strikers types'], axis = 1)
y = processed_df['Strikers types']
```

Scaling features

```
scaler = StandardScaler()
scaled_x = scaler.fit_transform(x)
scaled_x

array([[-1.07047781, -1.03252879,  0.5050467 , ..., -0.47801802,
        -0.47169258,  1.86125917],
       [-1.07047781,  0.968496  , -0.10638998, ...,  2.09197134,
        -0.47169258, -0.53727069],
       [-1.07047781, -1.03252879,  0.70885893, ..., -0.47801802,
         2.12002488, -0.53727069],
       ...,
       [-1.07047781,  0.968496  , -0.10638998, ..., -0.47801802,
        -0.47169258, -0.53727069],
       [ 0.93416229,  0.968496  , -0.9216389 , ..., -0.47801802,
        -0.47169258, -0.53727069],
       [-1.07047781, -1.03252879, -1.32926335, ..., -0.47801802,
        -0.47169258, -0.53727069]])
```

Train test split

```
from sklearn.model_selection import train_test_split

x_train, x_test, y_train, y_test = train_test_split(scaled_x, y,
test_size = 0.2, random_state = 42)
```

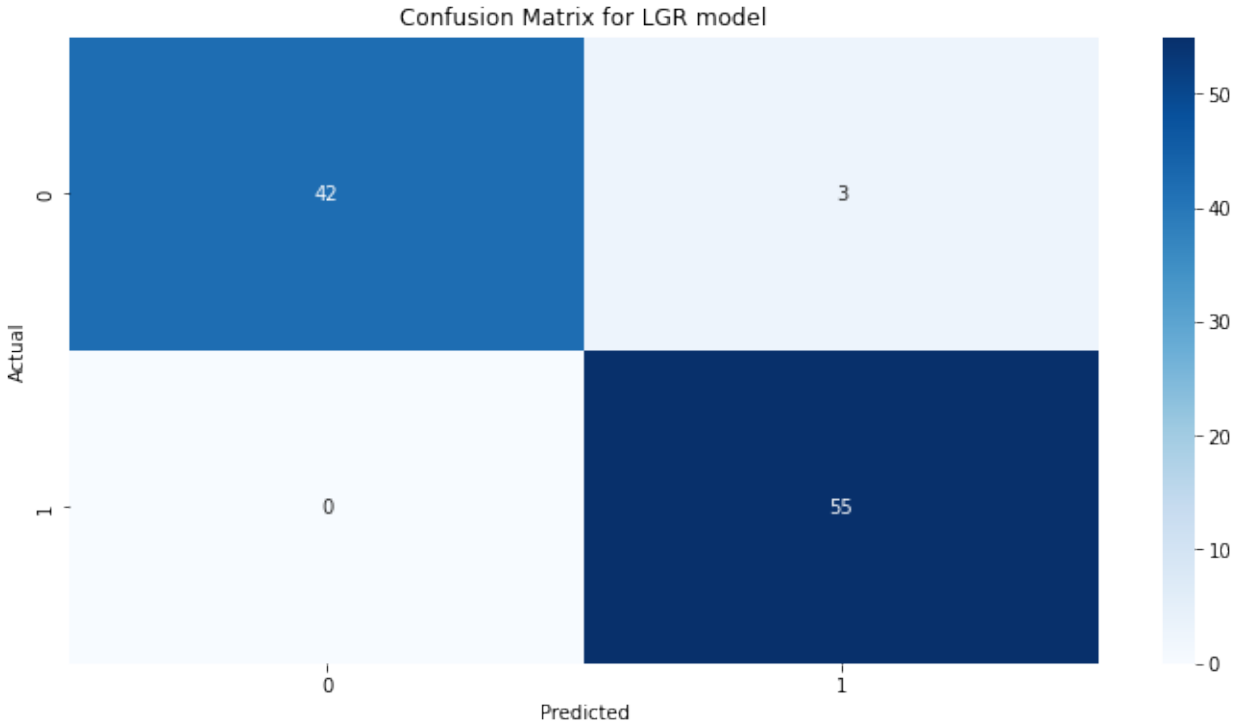
Predictive Classification Analytics

Build a logistic regression machine learning model to predict strikers type

```
# Model training
lgr_model = LogisticRegression()
lgr_model.fit(x_train, y_train)

#Prediction
y_lgr_pred = lgr_model.predict(x_test)

# Evaluation
accuracy_lgr = accuracy_score(y_test, y_lgr_pred)
print(accuracy_lgr*100,'%')
```

```
97.0 %
```

```
# Creating confusion matrix
conf_matrix_lgr = confusion_matrix(y_test, y_lgr_pred)

# Plotting confusion matrix
plt.figure(figsize = (12, 6))
sns.heatmap(conf_matrix_lgr, annot = True, fmt = "d", cmap = "Blues")
plt.title('Confusion Matrix for LGR model')
plt.xlabel('Predicted')
plt.ylabel('Actual')
plt.show()
```


Confusion Matrix for LGR model
42
3
0
55
Actual
0
1
Predicted
0
1
50
40
30
20
10
0

Thank you!