[1]:
```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings("ignore")
```

[2]:
```python
football_data = pd.read_csv("C:/Users/mutis/Desktop/EPLDATA/FootballDataset.csv" ,encoding="unicode_escape")
```

[3]:
```python
football_data.head()
```

Out[3]:

| | Name | Club | Nationality | Position | Age | Matches | Starts | Mins | Goals | Assists | Passes_Attempted | Perc_Passes_Completed | Penalty_Goals | Penalty_Attempted | xG | xA | Yellow_Cards | Red_Cards |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | Mason Mount | Chelsea | ENG | MF,FW | 21 | 36 | 32 | 2890 | 6 | 5 | 1881 | 82.3 | 1 | 1 | 0.21 | 0.24 | 2 | 0 |
| 1 | Edouard Mendy | Chelsea | SEN | GK | 28 | 31 | 31 | 2745 | 0 | 0 | 1007 | 84.6 | 0 | 0 | 0.00 | 0.00 | 2 | 0 |
| 2 | Timo Werner | Chelsea | GER | FW | 24 | 35 | 29 | 2602 | 6 | 8 | 826 | 77.2 | 0 | 0 | 0.41 | 0.21 | 2 | 0 |
| 3 | Ben Chilwell | Chelsea | ENG | DF | 23 | 27 | 27 | 2286 | 3 | 5 | 1806 | 78.6 | 0 | 0 | 0.10 | 0.11 | 3 | 0 |
| 4 | Reece James | Chelsea | ENG | DF | 20 | 32 | 25 | 2373 | 1 | 2 | 1987 | 85.0 | 0 | 0 | 0.06 | 0.12 | 3 | 0 |

[4]:
```python
football_data.describe
```

Out[4]:
```
<bound method NDFrame.describe of                  Name             Club Nationality Position  Age  Matches  \
0         Mason Mount          Chelsea         ENG    MF,FW   21       36
1       Edouard Mendy          Chelsea         SEN       GK   28       31
2         Timo Werner          Chelsea         GER       FW   24       35
3        Ben Chilwell          Chelsea         ENG       DF   23       27
4         Reece James          Chelsea         ENG       DF   20       32
..                ...              ...         ...      ...  ...      ...
527       Lys Mousset  Sheffield United         FRA    FW,MF   24       11
528    Jack O'Connell  Sheffield United         ENG       DF   26        2
529     Iliman Ndiaye  Sheffield United         FRA       MF   21        1
530  Antwoine Hackford  Sheffield United        ENG    DF,FW   16        1
531       Femi Seriki  Sheffield United         ENG       DF   17        1

     Starts  Mins  Goals  Assists  Passes_Attempted  Perc_Passes_Completed  \
0        32  2890      6        5              1881                   82.3
1        31  2745      0        0              1007                   84.6
2        29  2602      6        8               826                   77.2
3        27  2286      3        5              1806                   78.6
4        25  2373      1        2              1987                   85.0
..      ...   ...    ...      ...               ...                    ...
527       2   296      0        0                50                   80.0
528       2   180      0        0                77                   77.9
529       0    12      0        0                 3                  100.0
530       0    11      0        0                 1                  100.0
531       0     1      0        0                 0                   -1.0

     Penalty_Goals  Penalty_Attempted    xG    xA  Yellow_Cards  Red_Cards
0                1                  1  0.21  0.24             2          0
1                0                  0  0.00  0.00             2          0
2                0                  0  0.41  0.21             2          0
3                0                  0  0.10  0.11             3          0
4                0                  0  0.06  0.12             3          0
..             ...                ...   ...   ...           ...        ...
527              0                  0  0.22  0.10             0          0
528              0                  0  0.00  0.00             0          0
529              0                  0  0.00  0.00             0          0
530              0                  0  1.16  0.00             0          0
531              0                  0  0.00  0.00             0          0

[532 rows x 18 columns]>
```

# Cleaning the Dataset

[5]:
```python
football_data.isnull()
```

Out[5]:

| | Name | Club | Nationality | Position | Age | Matches | Starts | Mins | Goals | Assists | Passes_Attempted | Perc_Passes_Completed | Penalty_Goals | Penalty_Attempted | xG | xA | Yellow_Cards | Red_Cards |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False |
| 1 | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False |
| 2 | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False |
| 3 | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False |
| 4 | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 527 | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False |
| 528 | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False |
| 529 | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False |
| 530 | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False |
| 531 | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False | False |

532 rows × 18 columns

[6]:
```python
football_data.isna().sum()
```

Out[6]:
```
Name                     0
Club                     0
Nationality              0
Position                 0
Age                      0
Matches                  0
Starts                   0
Mins                     0
Goals                    0
Assists                  0
Passes_Attempted         0
Perc_Passes_Completed    0
Penalty_Goals            0
Penalty_Attempted        0
xG                       0
xA                       0
Yellow_Cards             0
Red_Cards                0
dtype: int64
```

# Data Exploration

[7]:
```python
#adding 2 new columns to  the dataset to further explore
football_data ['MinsPerMatch'] = (football_data ['Mins'] / football_data['Matches']).astype(int)
football_data ['GoalsPerMatch'] = (football_data['Goals']/ football_data['Matches']).astype(float)
football_data.head()
```

Out[7]:

| | Name | Club | Nationality | Position | Age | Matches | Starts | Mins | Goals | Assists | Passes_Attempted | Perc_Passes_Completed | Penalty_Goals | Penalty_Attempted | xG | xA | Yellow_Cards | Red_Cards | MinsPerMatch | GoalsPerMatch |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | Mason Mount | Chelsea | ENG | MF,FW | 21 | 36 | 32 | 2890 | 6 | 5 | 1881 | 82.3 | 1 | 1 | 0.21 | 0.24 | 2 | 0 | 80 | 0.166667 |
| 1 | Edouard Mendy | Chelsea | SEN | GK | 28 | 31 | 31 | 2745 | 0 | 0 | 1007 | 84.6 | 0 | 0 | 0.00 | 0.00 | 2 | 0 | 88 | 0.000000 |
| 2 | Timo Werner | Chelsea | GER | FW | 24 | 35 | 29 | 2602 | 6 | 8 | 826 | 77.2 | 0 | 0 | 0.41 | 0.21 | 2 | 0 | 74 | 0.171429 |
| 3 | Ben Chilwell | Chelsea | ENG | DF | 23 | 27 | 27 | 2286 | 3 | 5 | 1806 | 78.6 | 0 | 0 | 0.10 | 0.11 | 3 | 0 | 84 | 0.111111 |
| 4 | Reece James | Chelsea | ENG | DF | 20 | 32 | 25 | 2373 | 1 | 2 | 1987 | 85.0 | 0 | 0 | 0.06 | 0.12 | 3 | 0 | 74 | 0.031250 |

[8]:
```python
Total_Goals = football_data['Goals'].sum()
print(Total_Goals)
```

986

[9]:
```python
def country(x):
    return football_data[football_data['Nationality'] == x][['Name','Position','Goals','MinsPerMatch','GoalsPerMatch' ]]
```

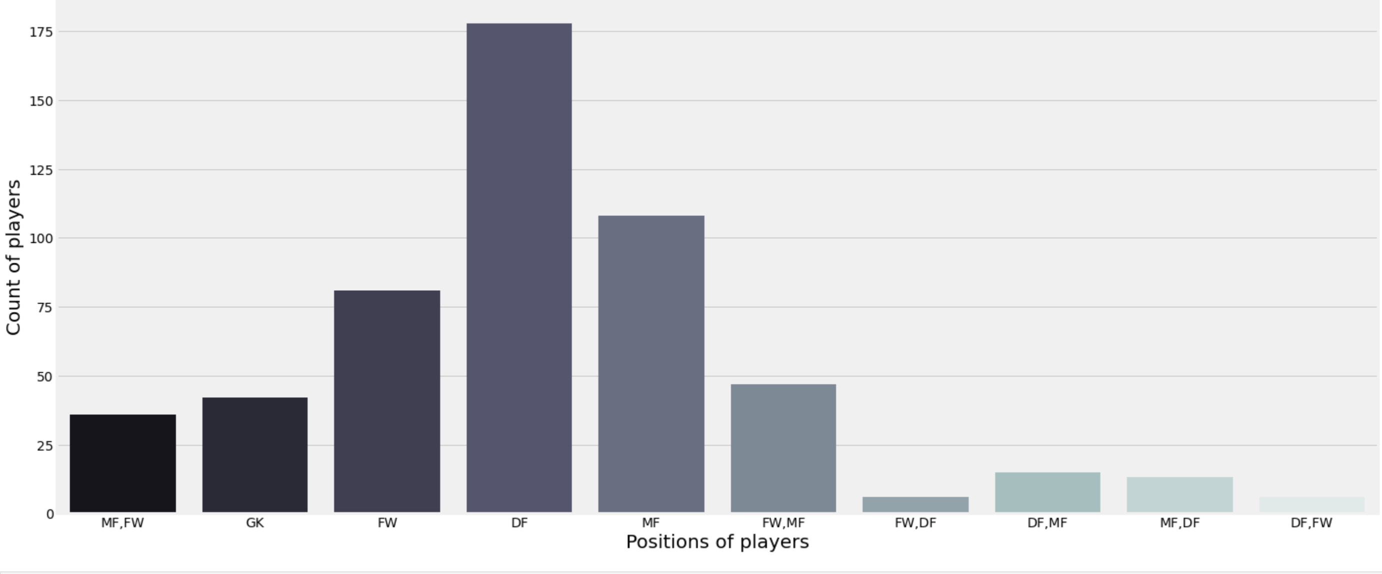
[10]:
```python
country('ENG')
```

[10]:

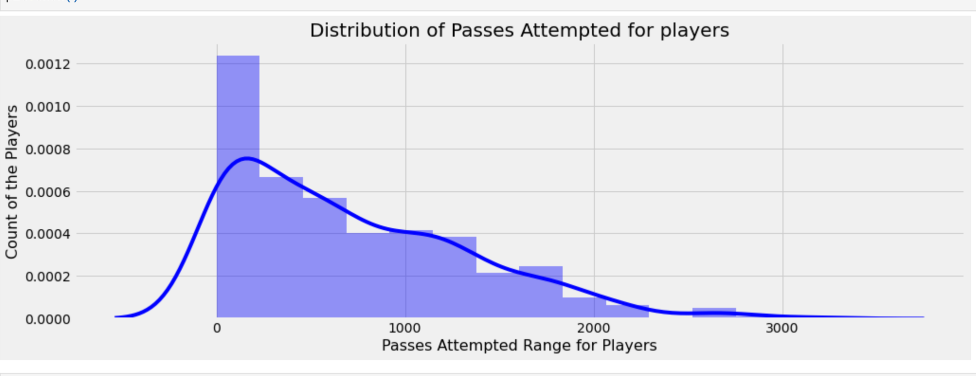
| | Name | Position | Goals | MinsPerMatch | GoalsPerMatch |
|---|---|---|---|---|---|
| 0 | Mason Mount | MF,FW | 6 | 80 | 0.166667 |
| 3 | Ben Chilwell | DF | 3 | 84 | 0.111111 |
| 4 | Reece James | DF | 1 | 74 | 0.031250 |
| 16 | Tammy Abraham | FW | 6 | 47 | 0.272727 |
| 18 | Callum Hudson-Odoi | FW,DF | 2 | 46 | 0.086957 |
| ... | ... | ... | ... | ... | ... |
| 525 | Phil Jagielka | DF | 0 | 52 | 0.000000 |
| 526 | Daniel Jebbison | FW | 1 | 71 | 0.250000 |
| 528 | Jack O'Connell | DF | 0 | 90 | 0.000000 |
| 530 | Antwoine Hackford | DF,FW | 0 | 11 | 0.000000 |
| 531 | Femi Seriki | DF | 0 | 1 | 0.000000 |

192 rows × 5 columns

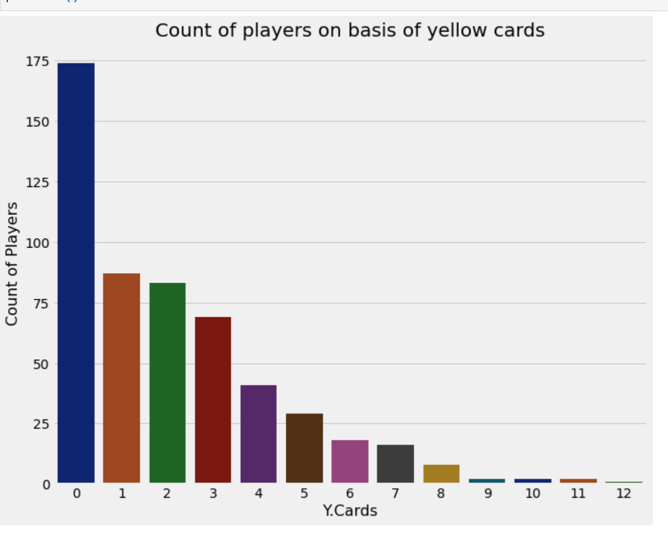
[11]:
```python
# different positions acquired by the players
plt.figure(figsize=(25,10))
plt.style.use('fivethirtyeight')
a=sns.countplot(football_data['Position'],palette='bone')
a.set_xlabel(xlabel='Positions of players',fontsize=20)
a.set_ylabel(ylabel='Count of players',fontsize=20)
plt.show()
```




[12]:
```python
#comparing the players passes attempted
import warnings
warnings.filterwarnings('ignore')
#plt.rcParams['figure.figsize']=(15,5)
plt.figure(figsize=(15,5))
a=sns.distplot(football_data['Passes_Attempted'],color='blue')
a.set_xlabel('Passes Attempted Range for Players',fontsize=16)
a.set_ylabel('Count of the Players',fontsize=16)
a.set_title('Distribution of Passes Attempted for players',fontsize=20)
plt.show()
```




[13]:
```python
plt.figure(figsize=(10,8))
p=sns.countplot(football_data['Yellow_Cards'],palette='dark')
p.set_title('Count of players on basis of yellow cards',fontsize=20)
p.set_xlabel(xlabel='Y.Cards',fontsize=16)
p.set_ylabel(ylabel='Count of Players',fontsize=16)
plt.show()
```




[14]:
```python
# best players per each position with their age, club, and nationality based on their xG

football_data.iloc[football_data.groupby(football_data['Position'])['xG'].idxmax()][['Position', 'Name',
                                                    'Age', 'Club', 'Nationality']].style.background_gradient('Blues')
```

[14]:

| | Position | Name | Age | Club | Nationality |
|---|---|---|---|---|---|
| 279 | DF | Kortney Hause | 25 | Aston Villa | ENG |
| 530 | DF,FW | Antwoine Hackford | 16 | Sheffield United | ENG |
| 387 | DF,MF | Yan Valery | 21 | Southampton | FRA |
| 110 | FW | Jamie Vardy | 33 | Leicester City | ENG |
| 418 | FW,DF | Andi Zeqiri | 21 | Brighton | SUI |
| 339 | FW,MF | Patrick Cutrone | 22 | Wolverhampton Wanderers | ITA |
| 1 | GK | Edouard Mendy | 28 | Chelsea | SEN |
| 51 | MF | Bruno Fernandes | 25 | Manchester United | POR |
| 401 | MF,DF | Pascal Groß | 29 | Brighton | GER |
| 257 | MF,FW | Joshua King | 28 | Everton | NOR |

[15]:
```python
#  worst players per each position with their age, club, and nationality based on their xG

football_data.iloc[football_data.groupby(football_data['Position'])['xG'].idxmin()][['Position', 'Name',
                                                    'Age', 'Club', 'Nationality']].style.background_gradient('Blues')
```

[15]:

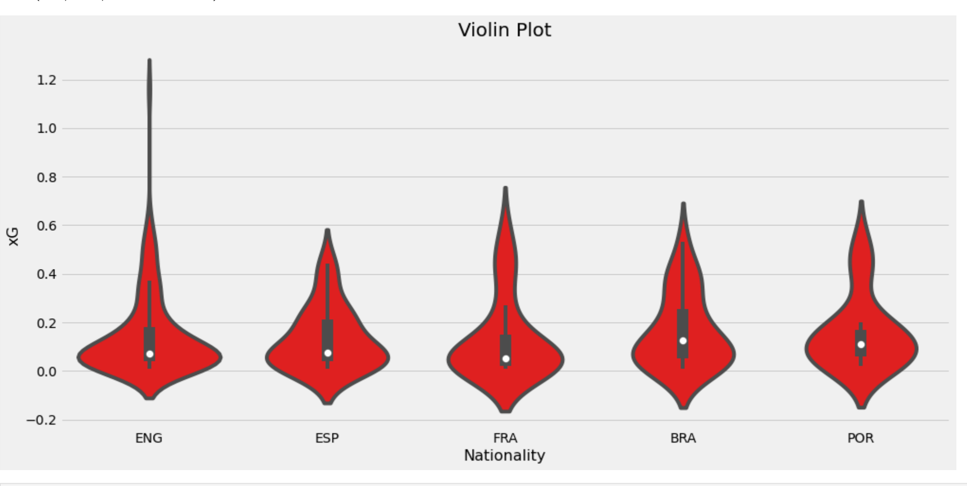
| | Position | Name | Age | Club | Nationality |
|---|---|---|---|---|---|
| 24 | DF | Emerson Palmieri | 25 | Chelsea | ITA |
| 255 | DF,FW | Niels Nkounkou | 19 | Everton | FRA |
| 435 | DF,MF | Erik Pieters | 31 | Burnley | NED |
| 23 | FW | Ruben Loftus-Cheek | 24 | Chelsea | ENG |
| 157 | FW,DF | Robert Snodgrass | 32 | West Ham United | SCO |
| 158 | FW,MF | Felipe Anderson | 27 | West Ham United | BRA |
| 1 | GK | Edouard Mendy | 28 | Chelsea | SEN |
| 78 | MF | Hannibal Mejbri | 17 | Manchester United | FRA |
| 228 | MF,DF | Jamie Shackleton | 20 | Leeds United | ENG |
| 393 | MF,FW | Caleb Watts | 18 | Southampton | AUS |

[16]:
```python
#Countries with most number of players
football_data['Nationality'].value_counts().head(10)
```

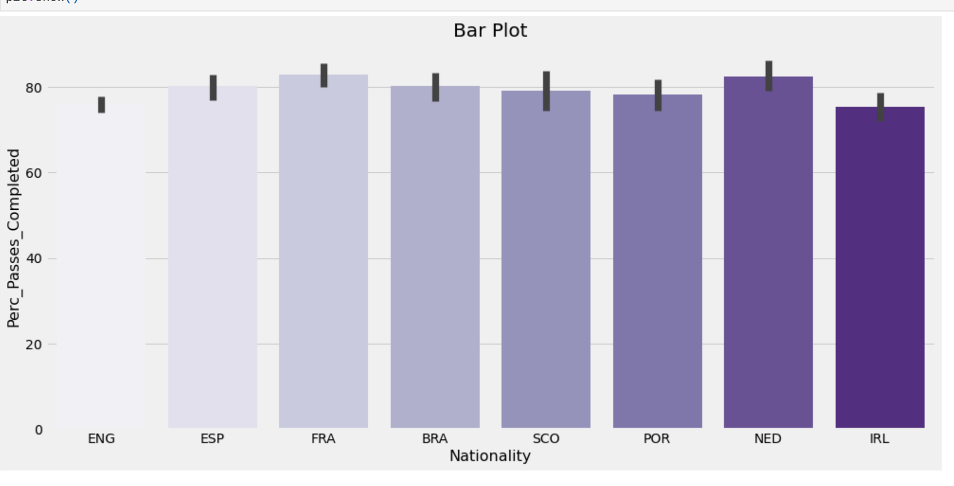
[16]:
```
ENG    192
FRA     31
BRA     27
ESP     26
IRL     21
POR     21
SCO     20
NED     16
WAL     12
BEL     11
Name: Nationality, dtype: int64
```

[17]:
```python
# Every Nations' Player and their xG
plt.rcParams['figure.figsize']=(15,7)
countriesl=['ENG','FRA','ENG','BRA','ESP','POR']
football_data_countries=football_data.loc[football_data['Nationality'].isin(countriesl) & football_data['xG']]
a=sns.violinplot(x=football_data_countries['Nationality'],y=football_data_countries['xG'],color='red')
a.set_xlabel(xlabel='Nationality',fontsize=16)
a.set_ylabel(ylabel='xG',fontsize=16)
a.set_title(label='Violin Plot',fontsize=20)
```

[17]: Text(0.5, 1.0, 'Violin Plot')




[18]:
```python
# Every Nations' Player and their Perc_Passes_Completed

plt.rcParams['figure.figsize']=(15,7)
countriesl=['ENG','ESP','BRA','POR','FRA','NED','SCO','IRL']
football_data_countries=football_data.loc[football_data['Nationality'].isin(countriesl) & football_data['Perc_Passes_Completed']]
a=sns.barplot(x=football_data_countries['Nationality'],y=football_data_countries['Perc_Passes_Completed'],palette='Purples')
a.set_xlabel(xlabel='Nationality',fontsize=16)
a.set_ylabel(ylabel='Perc_Passes_Completed',fontsize=16)
a.set_title(label='Bar Plot',fontsize=20)
plt.show()
```

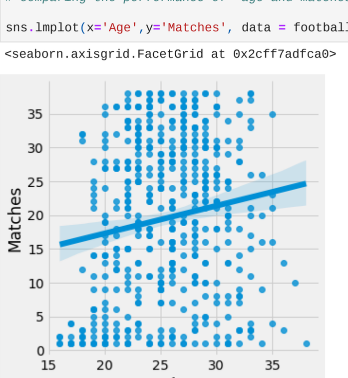



I now want to find out whether age can influence a player's expected goals. i therefore posed the question and found the most suitable algorithm to use.

## Can Age influence player performance through expected Goals?

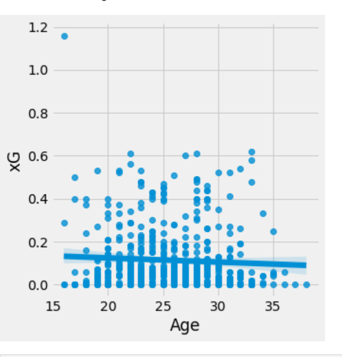
[19]:
```python
#I decided to explore age and the other columns
# comparing the performance of  age and matches of footballers

sns.lmplot(x='Age',y='Matches', data = football_data)
```

[19]: <seaborn.axisgrid.FacetGrid at 0x2cff7adfca0>



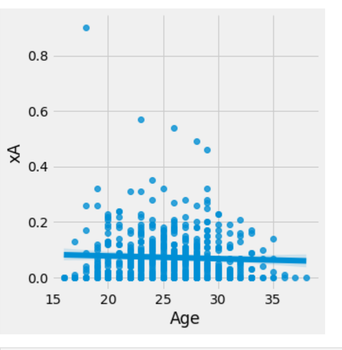

[20]:
```python
# comparing the performance of  age and xG of footballers

sns.lmplot(x='Age',y='xG', data = football_data)
```

[20]: <seaborn.axisgrid.FacetGrid at 0x2cff7dcd880>



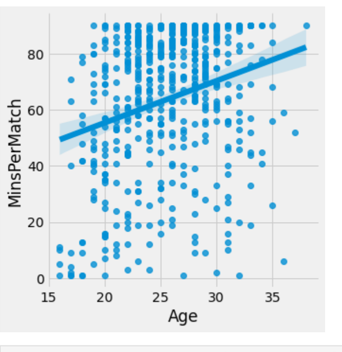

[21]:
```python
# comparing the performance of  age and xA of footballers

sns.lmplot(x='Age',y='xA', data = football_data)
```

[21]: <seaborn.axisgrid.FacetGrid at 0x2cff7d816d0>



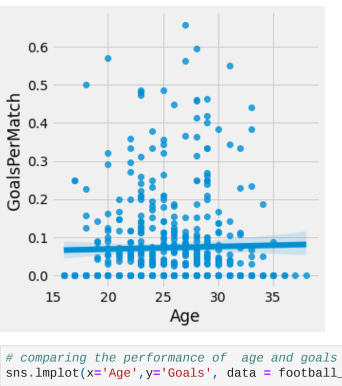

[22]:
```python
# comparing the performance of  age and minutes per match of footballers
sns.lmplot(x='Age',y='MinsPerMatch', data = football_data)
```

[22]: <seaborn.axisgrid.FacetGrid at 0x2cff7e84400>



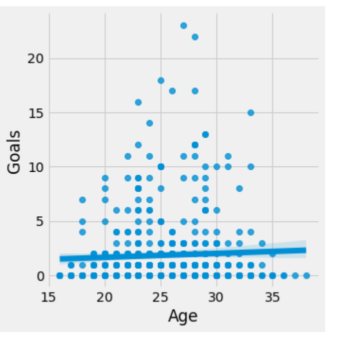

[23]:
```python
# comparing the performance of  age and  goals per match of footballers
sns.lmplot(x='Age',y='GoalsPerMatch', data = football_data)
```

[23]: <seaborn.axisgrid.FacetGrid at 0x2cff820e6a0>




[24]:
```python
# comparing the performance of  age and goals of footballers
sns.lmplot(x='Age',y='Goals', data = football_data)
```

[24]: <seaborn.axisgrid.FacetGrid at 0x2cff7e71370>




## Linear Regression algorithm to find the probability that age can influence expected goals outcome of a player

[25]:
```python
#step 1 import the required modules

from sklearn.datasets import make_classification
from matplotlib import pyplot as plt
from sklearn.linear_model import LinearRegression
```

[26]:
```python
# we need to set our independent and dependent variables
y = football_data.xG
x = football_data.Age.values.reshape(-1,1)
```

[27]:
```python
print(x.shape , y.shape)
```

(532, 1) (532,)

[28]:
```python
model = LinearRegression().fit(x,y)
```

[29]:
```python
r_sq = model.score(x,y)
intercept = model.intercept_
slope = model.coef_
```

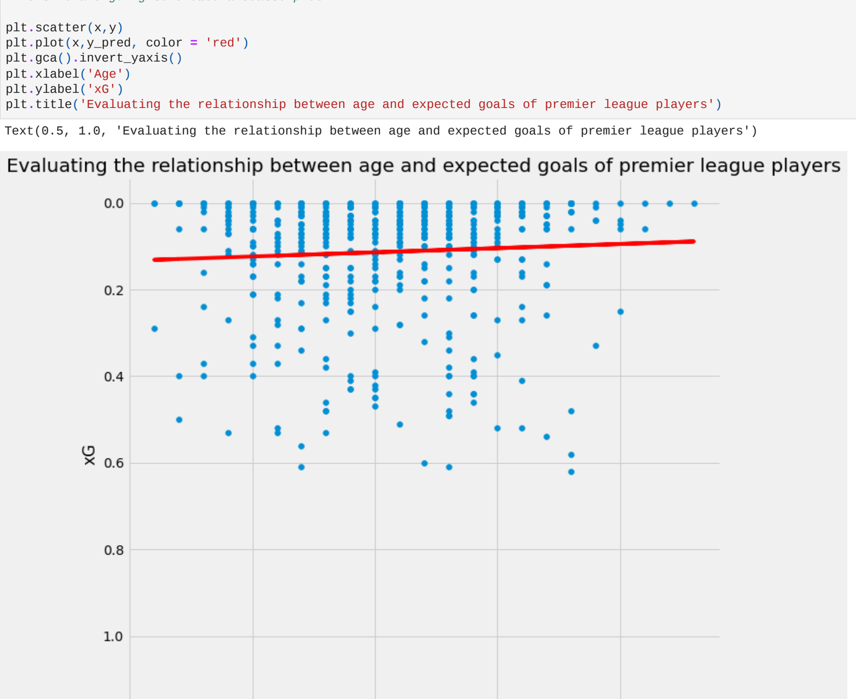
[30]:
```python
print(r_sq)
```

0.0031469845589144896

[31]:
```python
y_pred = intercept + slope* x
```

[32]:
```python
fig, ax = plt.subplots(figsize = (10,10))
# next we are going to create a scatterplot

plt.scatter(x,y)
plt.plot(x,y_pred, color = 'red')
plt.gca().invert_yaxis()
plt.xlabel('Age')
plt.ylabel('xG')
plt.title('Evaluating the relationship between age and expected goals of premier league players')
```

[32]: Text(0.5, 1.0, 'Evaluating the relationship between age and expected goals of premier league players')




In conclusion, the study indicates R^2 of the study to be 0.00315 after using Linear Regression. This means that age doesn't influence the expected Goals of a premier league player, this is indicated in the scatterplot as players who are regarded to be in their prime years (18-30) can have a lower expected goal, and players on their 30s have also shown they can have a higher expected goal stat. However, the study included all premier league players, not regarding their playing position as a limiting factor. Therefore, the study can be improved further by considering player position for example assessing whether age can influence the expected goals of Premier League Fowards(CF).

[ ]: