```python
import numpy as np
import pandas as pd
```

Loading Data

```python
player_stats=pd.read_excel('individual player
stats.xlsx',sheet_name='player table')
player_stats.head()
```

	full_name	age	birthday_GMT	league
0	Aaron Rowe	22	2000-09-07	Premier League
1	Abdelhamid Sabiri	25	1996-11-28	Premier League
2	Adalberto Peñaranda Maestre	25	1997-05-31	Premier League
3	Adrián San Miguel del Castillo	35	1987-01-03	Premier League
4	Adrien Sebastian Perruchet Silva	33	1989-03-15	Premier League

	season	position	Current Club	minutes_played_overall
0	2018/2019	Forward	Huddersfield Town	69
1	2018/2019	Midfielder	Huddersfield Town	49
2	2018/2019	Forward	Watford	0
3	2018/2019	Goalkeeper	West Ham United	0
4	2018/2019	Midfielder	Leicester City	88

	minutes_played_home	minutes_played_away	...	conceded_per_90_overall
0	14	55	...	1.30
1	0	49	...	5.51
2	0	0	...	0.00
3	0	0	...	0.00
4	8	80	...	1.02

	min_per_conceded_overall	min_per_match	min_per_card_overall
0	69	35	0
1	16	25	0
2	0	0	0
3	0	0	0

```
4                                88                   44                        0

    min_per_assist_overall  cards_per_90_overall
rank_in_league_top_attackers  \
0                        0                   0.0
-1
1                        0                   0.0
-1
2                        0                   0.0
-1
3                        0                   0.0
-1
4                        0                   0.0
-1

    rank_in_league_top_midfielders  rank_in_league_top_defenders  \
0                               -1                            -1
1                               -1                            -1
2                               -1                            -1
3                               -1                            -1
4                               -1                            -1

    rank_in_club_top_scorer
0                       31
1                       22
2                       -1
3                       -1
4                       15

[5 rows x 46 columns]

overall_league_stats=pd.read_excel('overall league stats.xlsx')

overall_league_stats.head()

    average_scored_home_team  average_scored_away_team  btts_percentage
\
0                       1.57                      1.25                51

    clean_sheets_percentage  prediction_risk
home_scored_advantage_percentage  \
0                        54               73
26

    home_defence_advantage_percentage  home_advantage_percentage  \
0                                   20                         23

    average_cards_per_match  average_cards_per_match_home_team   ...  \
0                       3.46                              1.62   ...
```

```
    goals_min_0_to_10  goals_min_11_to_20  goals_min_21_to_30  \
0                  82                 116                 108

    goals_min_31_to_40  goals_min_41_to_50  goals_min_51_to_60  \
0                  98                 118                 112

    goals_min_61_to_70  goals_min_71_to_80  goals_min_81_to_90  \
0                 133                 119                 186

    xg_avg_per_match
0                2.8

[1 rows x 22 columns]
```

```python
prem_team_stats_1=pd.read_excel('prem team stats 1.xlsx')
```

```python
prem_team_stats_1.head()
```

```
         common_name  wins  wins_home  wins_away  draws  draws_home  \
0             Arsenal    21         14          7      7           3
1   Tottenham Hotspur    23         12         11      2           2
2     Manchester City    32         18         14      2           0
3      Leicester City    15          8          7      7           3
4      Crystal Palace    14          5          9      7           5

    draws_away  losses  losses_home  losses_away  ...  \
0            4      10            2            8  ...
1            0      13            5            8  ...
2            2       4            1            3  ...
3            4      16            8            8  ...
4            2      17            9            8  ...

    goals_scored_min_81_to_90  goals_conceded_min_0_to_10  \
0                          13                           3
1                          10                           3
2                          10                           0
3                          14                           8
4                          11                           1

    goals_conceded_min_11_to_20  goals_conceded_min_21_to_30  \
0                            7                            5
1                            4                            1
2                            2                            3
3                            5                            1
4                            2                            3

    goals_conceded_min_31_to_40  goals_conceded_min_41_to_50  \
0                            5                            9
1                            2                            1
2                            3                            3
3                            7                            6
```

```
4                                4                                 7

   goals_conceded_min_51_to_60  goals_conceded_min_61_to_70  \
0                            1                           12
1                            6                            4
2                            3                            4
3                            4                            7
4                            9                            8

   goals_conceded_min_71_to_80  goals_conceded_min_81_to_90
0                            1                            8
1                            8                           10
2                            2                            3
3                            1                            9
4                            8                           11

[5 rows x 169 columns]
```

```python
prem_team_stats_2=pd.read_excel('prem team stats 2.xlsx')
```

```python
prem_team_stats_2.head()
```

```
        common_name  average_attendance_overall
average_attendance_home  \
0            Arsenal                       49412
59899
1  Tottenham Hotspur                       45931
54216
2     Manchester City                      45914
54130
3      Leicester City                      34948
31851
4      Crystal Palace                      32193
25455

   average_attendance_away  minutes_per_goal_scored_overall  \
0                    38925                               47
1                    37645                               51
2                    37698                               36
3                    38046                               67
4                    38932                               67

   minutes_per_goal_scored_home  minutes_per_goal_scored_away  \
0                            41                            55
1                            50                            52
2                            30                            45
3                            71                            63
4                            90                            54

   minutes_per_goal_conceded_overall
minutes_per_goal_conceded_home  \
```

```
0                                         67                           107

1                                         87                           107

2                                        148                           143

3                                         71                            86

4                                         65                            74


   minutes_per_goal_conceded_away  ...  wins_2h_percentage_home  \
0                              49  ...                       68
1                              74  ...                       57
2                             155  ...                       79
3                              61  ...                       42
4                              57  ...                       21

   wins_2h_percentage_away  draws_2h_home  draws_2h_away
losses_2h_home  \
0                       37              4              5
2
1                       26              5              7
3
2                       53              4              6
0
3                       63              7              4
4
4                       32              6              6
9

   losses_2h_away  clean_sheets_2h_home  clean_sheets_2h_away  \
0               7                    12                     6
1               7                    11                     9
2               3                    14                    13
3               3                    13                     9
4               7                     8                     6

   failed_to_score_2h_home  failed_to_score_2h_away
0                        4                       10
1                        5                        9
2                        2                        7
3                        9                        4
4                       10                        5

[5 rows x 96 columns]

stats_from_matches=pd.read_excel('stats from matches.xlsx')

stats_from_matches.head()
```

```
      timestamp                 date_GMT   attendance       home_team_name  \
0   1533927600    Aug 10 2018 - 7:00pm          74439    Manchester United
1   1533987000   Aug 11 2018 - 11:30am          51749      Newcastle United
2   1533996000    Aug 11 2018 - 2:00pm          10353       AFC Bournemouth
3   1533996000    Aug 11 2018 - 2:00pm          24821               Fulham
4   1533996000    Aug 11 2018 - 2:00pm          24121      Huddersfield Town

       away_team_name            referee   Game Week   Pre-Match PPG (Home)  \
0       Leicester City    Andre Marriner           1                    0.0

1    Tottenham Hotspur   Martin Atkinson           1                    0.0

2         Cardiff City       Kevin Friend           1                    0.0

3       Crystal Palace          Mike Dean           1                    0.0

4              Chelsea    Chris Kavanagh           1                    0.0


    Pre-Match PPG (Away)   home_ppg   ...   home_team_fouls
away_team_fouls  \
0                    0.0       1.89   ...                11
8
1                    0.0       1.32   ...                11
12
2                    0.0       1.53   ...                11
9
3                    0.0       1.11   ...                 9
11
4                    0.0       0.47   ...                 9
8

    home_team_possession   away_team_possession   Home Team Pre-Match xG
\
0                      46                     54                      0.0

1                      40                     60                      0.0

2                      62                     38                      0.0

3                      66                     34                      0.0

4                      37                     63                      0.0


    Away Team Pre-Match xG   team_a_xg  team_b_xg  \
0                      0.0        1.25       1.46
1                      0.0        1.18       1.58
2                      0.0        1.50       0.89
3                      0.0        1.67       1.81
```

```
4                          0.0        0.90        1.17

  average_goals_per_match_pre_match  \
0                                0.0
1                                0.0
2                                0.0
3                                0.0
4                                0.0

                                      stadium_name
0                          Old Trafford (Manchester)
1                 St. James' Park (Newcastle upon Tyne)
2              Vitality Stadium (Bournemouth- Dorset)
3                           Craven Cottage (London)
4  John Smith's Stadium (Huddersfield- West Yorks...

[5 rows x 43 columns]
```

Data Cleaning

I have narrowed down some of the columns we need within Excel, but there maybe a couple that we don't need plus some that need formatting

```python
player_stats.drop(columns=['league','season'],axis=1,inplace=True)

player_stats.head()
```

```
                          full_name  age birthday_GMT     position  \
0                        Aaron Rowe   22   2000-09-07      Forward
1                   Abdelhamid Sabiri   25   1996-11-28   Midfielder
2        Adalberto Peñaranda Maestre   25   1997-05-31      Forward
3    Adrián San Miguel del Castillo   35   1987-01-03   Goalkeeper
4   Adrien Sebastian Perruchet Silva   33   1989-03-15   Midfielder

        Current Club  minutes_played_overall  minutes_played_home  \
0  Huddersfield Town                      69                   14
1  Huddersfield Town                      49                    0
2            Watford                       0                    0
3    West Ham United                       0                    0
4      Leicester City                     88                    8

   minutes_played_away nationality  appearances_overall  ...  \
0                   55     England                    2  ...
1                   49     Morocco                    2  ...
2                    0   Venezuela                    0  ...
3                    0       Spain                    0  ...
```

```
4                         80      Portugal                       2  ...

   conceded_per_90_overall   min_per_conceded_overall   min_per_match  \
0                      1.30                         69              35
1                      5.51                         16              25
2                      0.00                          0               0
3                      0.00                          0               0
4                      1.02                         88              44

   min_per_card_overall   min_per_assist_overall   cards_per_90_overall
\
0                      0                        0                    0.0

1                      0                        0                    0.0

2                      0                        0                    0.0

3                      0                        0                    0.0

4                      0                        0                    0.0


   rank_in_league_top_attackers   rank_in_league_top_midfielders  \
0                            -1                               -1
1                            -1                               -1
2                            -1                               -1
3                            -1                               -1
4                            -1                               -1

   rank_in_league_top_defenders   rank_in_club_top_scorer
0                            -1                        31
1                            -1                        22
2                            -1                        -1
3                            -1                        -1
4                            -1                        15

[5 rows x 44 columns]

player_stats.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 570 entries, 0 to 569
Data columns (total 44 columns):
 #   Column                      Non-Null Count   Dtype
---  ------                      --------------   -----
 0   full_name                   570 non-null     object
 1   age                         570 non-null     int64
 2   birthday_GMT                570 non-null     datetime64[ns]
 3   position                    570 non-null     object
 4   Current Club                570 non-null     object
 5   minutes_played_overall      570 non-null     int64
```

```
 6    minutes_played_home              570 non-null    int64
 7    minutes_played_away              570 non-null    int64
 8    nationality                      570 non-null    object
 9    appearances_overall              570 non-null    int64
10    appearances_home                 570 non-null    int64
11    appearances_away                 570 non-null    int64
12    goals_overall                    570 non-null    int64
13    goals_home                       570 non-null    int64
14    goals_away                       570 non-null    int64
15    assists_overall                  570 non-null    int64
16    assists_home                     570 non-null    int64
17    assists_away                     570 non-null    int64
18    penalty_goals                    570 non-null    int64
19    penalty_misses                   570 non-null    int64
20    clean_sheets_overall             570 non-null    int64
21    clean_sheets_home                570 non-null    int64
22    clean_sheets_away                570 non-null    int64
23    conceded_overall                 570 non-null    int64
24    conceded_home                    570 non-null    int64
25    conceded_away                    570 non-null    int64
26    yellow_cards_overall             570 non-null    int64
27    red_cards_overall                570 non-null    int64
28    goals_involved_per_90_overall    570 non-null    float64
29    assists_per_90_overall           570 non-null    float64
30    goals_per_90_overall             570 non-null    float64
31    goals_per_90_home                570 non-null    float64
32    goals_per_90_away                570 non-null    float64
33    min_per_goal_overall             570 non-null    int64
34    conceded_per_90_overall          570 non-null    float64
35    min_per_conceded_overall         570 non-null    int64
36    min_per_match                    570 non-null    int64
37    min_per_card_overall             570 non-null    int64
38    min_per_assist_overall           570 non-null    int64
39    cards_per_90_overall             570 non-null    float64
40    rank_in_league_top_attackers     570 non-null    int64
41    rank_in_league_top_midfielders   570 non-null    int64
42    rank_in_league_top_defenders     570 non-null    int64
43    rank_in_club_top_scorer          570 non-null    int64
dtypes: datetime64[ns](1), float64(7), int64(32), object(4)
memory usage: 196.1+ KB
```

```python
(player_stats['rank_in_league_top_attackers']==-1).sum()
```

151

```python
#want to check the forward

subset_players_df=player_stats[player_stats.position=='Forward']

subset_players_df
```

```
                            full_name  age birthday_GMT position    \
0                          Aaron Rowe   22   2000-09-07  Forward
2             Adalberto PeÃ±aranda Maestre   25   1997-05-31  Forward
8                      Alexander SÃ¸rloth   26   1995-12-05  Forward
9       Alexandre Nascimento Costa Silva   25   1997-03-16  Forward
21                     Benjamin Johnson   22   2000-01-24  Forward
..                                  ...  ...          ...      ...
453                       Elias Kachunga   30   1992-04-22  Forward
481                      Laurent Depoitre   33   1988-12-07  Forward
482                       Adama Diakhaby   26   1996-07-05  Forward
553                  Alireza Jahanbakhsh   29   1993-08-11  Forward
565                          Oumar Niasse   32   1990-04-18  Forward

             Current Club  minutes_played_overall
minutes_played_home    \
0          Huddersfield Town                      69
14
2                    Watford                       0
0
8              Crystal Palace                     173
121
9            West Ham United                      17
0
21           West Ham United                      63
0
..                     ...                     ...                         .
..
453        Huddersfield Town                    1181
611
481        Huddersfield Town                    1059
446
482        Huddersfield Town                     551
345
553    Brighton & Hove Albion                    1022
494
565              Cardiff City                     998
435

      minutes_played_away nationality  appearances_overall   ...   \
0                       55     England                     2   ...
2                        0   Venezuela                     0   ...
8                       52      Norway                    12   ...
9                       17    Portugal                     1   ...
21                      63     England                     1   ...
..                     ...         ...                   ...   ...
453                    570    Congo DR                    20   ...
481                    613     Belgium                    23   ...
482                    206      France                    12   ...
553                    528        Iran                    19   ...
565                    563     Senegal                    18   ...
```

	conceded_per_90_overall	min_per_conceded_overall	min_per_match
0	1.30	69	35
2	0.00	0	0
8	0.52	173	14
9	0.00	0	17
21	1.43	63	63
..	...	...	...
453	2.13	42	59
481	1.87	48	46
482	2.61	34	46
553	1.41	64	54
565	1.08	83	55

	min_per_card_overall	min_per_assist_overall	cards_per_90_overall
0	0	0	0.00
2	0	0	0.00
8	0	0	0.00
9	0	0	0.00
21	0	0	0.00
..	...	...	..
453	591	0	0.15
481	1059	1059	0.08
482	551	0	0.16
553	511	0	0.18
565	0	0	0.00

```
      rank_in_league_top_attackers  rank_in_league_top_midfielders  \
0                               -1                              -1
2                               -1                              -1
8                               -1                              -1
9                               -1                              -1
21                              -1                              -1
..                             ...                             ...
453                            303                             259
481                            331                             169
482                            332                             359
553                            403                             270
565                            415                             388

      rank_in_league_top_defenders  rank_in_club_top_scorer
0                               -1                       31
2                               -1                       -1
8                               -1                       17
9                               -1                       16
21                              -1                       26
..                             ...                      ...
453                             -1                       12
481                             -1                       27
482                             -1                       26
553                             -1                       16
565                             -1                       13

[113 rows x 44 columns]
```

#113 players

```python
(subset_players_df['rank_in_league_top_attackers']==-1).sum()
```

```
30
```

Instead of adjusting -1 for the ranking per position, it makes sense to do it across all the relevant columns as there is no place for -1 value

```python
def adjusting_Ranks(rank):
    if rank == -1:
        return None
    else:
        return rank

player_stats.head()
```

```
                             full_name  age birthday_GMT     position  \
0                            Aaron Rowe   22   2000-09-07      Forward
1                      Abdelhamid Sabiri   25   1996-11-28   Midfielder
2          Adalberto Peñaranda Maestre   25   1997-05-31      Forward
3   Adrián San Miguel del Castillo   35   1987-01-03   Goalkeeper
```

```
4    Adrien Sebastian Perruchet Silva    33    1989-03-15    Midfielder

          Current Club  minutes_played_overall   minutes_played_home  \
0   Huddersfield Town                       69                    14
1   Huddersfield Town                       49                     0
2             Watford                        0                     0
3     West Ham United                        0                     0
4       Leicester City                      88                     8

    minutes_played_away  nationality   appearances_overall  ...  \
0                    55       England                     2  ...
1                    49       Morocco                     2  ...
2                     0     Venezuela                     0  ...
3                     0         Spain                     0  ...
4                    80      Portugal                     2  ...

    conceded_per_90_overall   min_per_conceded_overall   min_per_match  \
0                      1.30                          69              35
1                      5.51                          16              25
2                      0.00                           0               0
3                      0.00                           0               0
4                      1.02                          88              44

    min_per_card_overall   min_per_assist_overall   cards_per_90_overall
\
0                      0                        0                    0.0

1                      0                        0                    0.0

2                      0                        0                    0.0

3                      0                        0                    0.0

4                      0                        0                    0.0


    rank_in_league_top_attackers   rank_in_league_top_midfielders  \
0                             -1                               -1
1                             -1                               -1
2                             -1                               -1
3                             -1                               -1
4                             -1                               -1

    rank_in_league_top_defenders   rank_in_club_top_scorer
0                             -1                        31
1                             -1                        22
2                             -1                        -1
3                             -1                        -1
4                             -1                        15
```

```
[5 rows x 44 columns]

# we are doing the first column to check if it works

player_stats['rank_in_league_top_attackers']=player_stats['rank_in_lea
gue_top_attackers'].apply(adjusting_Ranks)

player_stats.head()

                          full_name  age birthday_GMT     position  \
0                        Aaron Rowe   22   2000-09-07      Forward
1                  Abdelhamid Sabiri   25   1996-11-28   Midfielder
2        Adalberto Peñaranda Maestre   25   1997-05-31      Forward
3     Adrián San Miguel del Castillo   35   1987-01-03   Goalkeeper
4  Adrien Sebastian Perruchet Silva   33   1989-03-15   Midfielder

       Current Club  minutes_played_overall  minutes_played_home  \
0  Huddersfield Town                      69                   14
1  Huddersfield Town                      49                    0
2            Watford                       0                    0
3    West Ham United                       0                    0
4      Leicester City                     88                    8

   minutes_played_away nationality  appearances_overall  ...  \
0                   55     England                    2  ...
1                   49     Morocco                    2  ...
2                    0   Venezuela                    0  ...
3                    0       Spain                    0  ...
4                   80    Portugal                    2  ...

   conceded_per_90_overall  min_per_conceded_overall  min_per_match  \
0                     1.30                        69             35
1                     5.51                        16             25
2                     0.00                         0              0
3                     0.00                         0              0
4                     1.02                        88             44

   min_per_card_overall  min_per_assist_overall  cards_per_90_overall
\
0                     0                       0                   0.0

1                     0                       0                   0.0

2                     0                       0                   0.0

3                     0                       0                   0.0

4                     0                       0                   0.0
```

```
    rank_in_league_top_attackers  rank_in_league_top_midfielders  \
0                            NaN                              -1
1                            NaN                              -1
2                            NaN                              -1
3                            NaN                              -1
4                            NaN                              -1

    rank_in_league_top_defenders  rank_in_club_top_scorer
0                             -1                       31
1                             -1                       22
2                             -1                       -1
3                             -1                       -1
4                             -1                       15

[5 rows x 44 columns]
```

Trying to do multiple colums

```
player_stats['rank_in_league_top_midfielders']=player_stats['rank_in_l
eague_top_midfielders'].apply(adjusting_Ranks)

player_stats['rank_in_league_top_defenders']=player_stats['rank_in_lea
gue_top_defenders'].apply(adjusting_Ranks)

player_stats.head()
```

```
                          full_name  age birthday_GMT    position  \
0                        Aaron Rowe   22   2000-09-07     Forward
1                   Abdelhamid Sabiri   25   1996-11-28  Midfielder
2        Adalberto PeÃ±aranda Maestre   25   1997-05-31     Forward
3   AdriÃ¡n San Miguel del Castillo   35   1987-01-03  Goalkeeper
4   Adrien Sebastian Perruchet Silva   33   1989-03-15  Midfielder

        Current Club  minutes_played_overall  minutes_played_home  \
0  Huddersfield Town                      69                   14
1  Huddersfield Town                      49                    0
2            Watford                       0                    0
3    West Ham United                       0                    0
4      Leicester City                     88                    8

    minutes_played_away nationality  appearances_overall  ...  \
0                    55     England                    2  ...
1                    49     Morocco                    2  ...
2                     0   Venezuela                    0  ...
3                     0       Spain                    0  ...
4                    80    Portugal                    2  ...

    conceded_per_90_overall  min_per_conceded_overall  min_per_match  \
0                      1.30                        69             35
1                      5.51                        16             25
2                      0.00                         0              0
```

```
3                         0.00                           0              0
4                         1.02                          88             44

   min_per_card_overall  min_per_assist_overall  cards_per_90_overall
\
0                      0                       0                   0.0

1                      0                       0                   0.0

2                      0                       0                   0.0

3                      0                       0                   0.0

4                      0                       0                   0.0


   rank_in_league_top_attackers  rank_in_league_top_midfielders  \
0                            NaN                             NaN
1                            NaN                             NaN
2                            NaN                             NaN
3                            NaN                             NaN
4                            NaN                             NaN

   rank_in_league_top_defenders  rank_in_club_top_scorer
0                            NaN                       31
1                            NaN                       22
2                            NaN                       -1
3                            NaN                       -1
4                            NaN                       15

[5 rows x 44 columns]
```

now we are dealing with the rank in club top scorer issue

```python
# by looking at the watford team, we try and investigate why there are
-1 in that column

cleaning_top_scorers=player_stats.loc[player_stats['Current Club'] ==
'Watford', 'full_name':'rank_in_club_top_scorer']

cleaning_top_scorers.head()
```

```
                        full_name  age birthday_GMT     position
Current Club  \
2     Adalberto Peñaranda Maestre   25   1997-05-31      Forward
Watford
20                   Ben Wilmot   23   1999-11-04     Defender
Watford
58        Heurelho da Silva Gomes   41   1981-02-15   Goalkeeper
Watford
97         Marc Navarro Ceciliano   27   1995-07-02     Defender
```

```
Watford
109  Miguel Ã)ngel Britos Cabrera   37   1985-07-17     Defender
Watford

     minutes_played_overall  minutes_played_home  minutes_played_away
\
2                          0                    0                    0

20                        26                    0                   26

58                         0                    0                    0

97                        84                    0                   84

109                      185                   13                  172


    nationality  appearances_overall  ...  conceded_per_90_overall  \
2      Venezuela                    0  ...                     0.00
20       England                    2  ...                     6.92
58        Brazil                    0  ...                     0.00
97         Spain                    2  ...                     2.14
109      Uruguay                    3  ...                     2.43

     min_per_conceded_overall  min_per_match  min_per_card_overall  \
2                           0              0                     0
20                         13             13                     0
58                          0              0                     0
97                         42             42                     0
109                        37             62                     0

     min_per_assist_overall  cards_per_90_overall  \
2                         0                   0.0
20                        0                   0.0
58                        0                   0.0
97                        0                   0.0
109                       0                   0.0

     rank_in_league_top_attackers  rank_in_league_top_midfielders  \
2                             NaN                             NaN
20                            NaN                             NaN
58                            NaN                             NaN
97                            NaN                             NaN
109                           NaN                             NaN

     rank_in_league_top_defenders  rank_in_club_top_scorer
2                             NaN                       -1
20                            NaN                       17
58                            NaN                       -1
97                            NaN                       15
```

```
109                         NaN                              25

[5 rows x 44 columns]

cleaning_top_scorers.info()

<class 'pandas.core.frame.DataFrame'>
Int64Index: 27 entries, 2 to 547
Data columns (total 44 columns):
 #   Column                        Non-Null Count  Dtype
---  ------                        --------------  -----
 0   full_name                     27 non-null     object
 1   age                           27 non-null     int64
 2   birthday_GMT                  27 non-null     datetime64[ns]
 3   position                      27 non-null     object
 4   Current Club                  27 non-null     object
 5   minutes_played_overall        27 non-null     int64
 6   minutes_played_home           27 non-null     int64
 7   minutes_played_away           27 non-null     int64
 8   nationality                   27 non-null     object
 9   appearances_overall           27 non-null     int64
 10  appearances_home              27 non-null     int64
 11  appearances_away              27 non-null     int64
 12  goals_overall                 27 non-null     int64
 13  goals_home                    27 non-null     int64
 14  goals_away                    27 non-null     int64
 15  assists_overall               27 non-null     int64
 16  assists_home                  27 non-null     int64
 17  assists_away                  27 non-null     int64
 18  penalty_goals                 27 non-null     int64
 19  penalty_misses                27 non-null     int64
 20  clean_sheets_overall          27 non-null     int64
 21  clean_sheets_home             27 non-null     int64
 22  clean_sheets_away             27 non-null     int64
 23  conceded_overall              27 non-null     int64
 24  conceded_home                 27 non-null     int64
 25  conceded_away                 27 non-null     int64
 26  yellow_cards_overall          27 non-null     int64
 27  red_cards_overall             27 non-null     int64
 28  goals_involved_per_90_overall 27 non-null     float64
 29  assists_per_90_overall        27 non-null     float64
 30  goals_per_90_overall          27 non-null     float64
 31  goals_per_90_home             27 non-null     float64
 32  goals_per_90_away             27 non-null     float64
 33  min_per_goal_overall          27 non-null     int64
 34  conceded_per_90_overall       27 non-null     float64
 35  min_per_conceded_overall      27 non-null     int64
 36  min_per_match                 27 non-null     int64
 37  min_per_card_overall          27 non-null     int64
 38  min_per_assist_overall        27 non-null     int64
```

```
 39   cards_per_90_overall              27 non-null      float64
 40   rank_in_league_top_attackers      19 non-null      float64
 41   rank_in_league_top_midfielders    19 non-null      float64
 42   rank_in_league_top_defenders      7 non-null       float64
 43   rank_in_club_top_scorer           27 non-null      int64
dtypes: datetime64[ns](1), float64(10), int64(29), object(4)
memory usage: 9.5+ KB
```

```
cleaning_top_scorers[["goals_overall","rank_in_club_top_scorer"]]
```

	goals_overall	rank_in_club_top_scorer
2	0	-1
20	0	17
58	0	-1
97	0	15
109	0	25
111	0	19
130	0	22
134	0	16
175	7	3
179	10	1
207	9	2
222	1	13
247	6	4
249	1	11
274	5	5
293	1	14
303	3	7
327	3	6
331	1	10
349	2	8
390	1	12
403	1	9
515	0	18
543	0	24
545	0	23
546	0	21
547	0	20

```
# so from this we see that the goals overall is not indicative of -1,
```

```
cleaning_top_scorers[["goals_overall","minutes_played_overall","rank_i
n_club_top_scorer"]]
```

	goals_overall	minutes_played_overall	rank_in_club_top_scorer
2	0	0	-1
20	0	26	17
58	0	0	-1
97	0	84	15
109	0	185	25

```
111             0           266              19
130             0            13              22
134             0            22              16
175             7          1358               3
179            10          2082               1
207             9          2548               2
222             1           335              13
247             6          2825               4
249             1           470              11
274             5          3062               5
293             1           755              14
303             3          2394               7
327             3          3099               6
331             1          1037              10
349             2          2436               8
390             1          2045              12
403             1          2947               9
515             0          3420              18
543             0          1587              24
545             0          1521              23
546             0          1921              21
547             0          1003              20
```

#it is more indicative to those who have zero minutes, receive a rating of -1

 # As we are not interested in those that have zero minutes we will drop them

```python
zero_minute_player=player_stats[player_stats['rank_in_club_top_scorer']== -1].index
```

```python
player_stats.drop(zero_minute_player,inplace=True)
```

```python
player_stats['full_name'].count()
```

496

#we dropped 74 players

 # we want to chnage the birthdate column-into a dateime columns

```python
player_stats['birthday_GMT']=pd.to_datetime(player_stats['birthday_GMT'])
```

```python
player_stats['birthday_GMT']
```

```
0      2000-09-07
1      1996-11-28
4      1989-03-15
5      1992-07-05
6      1993-08-15
```

```
        ...
565    1990-04-18
566    1998-06-30
567    1992-10-02
568    1994-03-11
569    1990-06-21
Name: birthday_GMT, Length: 496, dtype: datetime64[ns]
```

```python
#now we look at the prem_team_stats

# no nulls as of now

prem_team_stats_1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20 entries, 0 to 19
Columns: 169 entries, common_name to goals_conceded_min_81_to_90
dtypes: float64(30), int64(138), object(1)
memory usage: 26.5+ KB
```

```python
prem_team_stats_1.isna().sum()
```

```
common_name                     0
wins                            0
wins_home                       0
wins_away                       0
draws                           0
                               ..
goals_conceded_min_41_to_50     0
goals_conceded_min_51_to_60     0
goals_conceded_min_61_to_70     0
goals_conceded_min_71_to_80     0
goals_conceded_min_81_to_90     0
Length: 169, dtype: int64
```

```python
# it appears the datatypes are ok for now

prem_team_stats_2.head()
```

```
        common_name  average_attendance_overall
average_attendance_home  \
0            Arsenal                       49412
59899
1  Tottenham Hotspur                       45931
54216
2    Manchester City                       45914
54130
3     Leicester City                       34948
31851
4     Crystal Palace                       32193
```

25455

	average_attendance_away	minutes_per_goal_scored_overall \
0	38925	47
1	37645	51
2	37698	36
3	38046	67
4	38932	67

	minutes_per_goal_scored_home	minutes_per_goal_scored_away \
0	41	55
1	50	52
2	30	45
3	71	63
4	90	54

	minutes_per_goal_conceded_overall	minutes_per_goal_conceded_home \
0	67	107
1	87	107
2	148	143
3	71	86
4	65	74

	minutes_per_goal_conceded_away	...	wins_2h_percentage_home \
0	49	...	68
1	74	...	57
2	155	...	79
3	61	...	42
4	57	...	21

	wins_2h_percentage_away	draws_2h_home	draws_2h_away	losses_2h_home \
0	37	4	5	2
1	26	5	7	3
2	53	4	6	0
3	63	7	4	4
4	32	6	6	9

	losses_2h_away	clean_sheets_2h_home	clean_sheets_2h_away \

```
0                     7                        12                          6
1                     7                        11                          9
2                     3                        14                         13
3                     3                        13                          9
4                     7                         8                          6

    failed_to_score_2h_home   failed_to_score_2h_away
0                         4                        10
1                         5                         9
2                         2                         7
3                         9                         4
4                        10                         5

[5 rows x 96 columns]

prem_team_stats_2.isna().sum()

common_name                         0
average_attendance_overall          0
average_attendance_home             0
average_attendance_away             0
minutes_per_goal_scored_overall     0
                                   ..
losses_2h_away                      0
clean_sheets_2h_home                0
clean_sheets_2h_away                0
failed_to_score_2h_home             0
failed_to_score_2h_away             0
Length: 96, dtype: int64
```

#Now we look at the matches data

```
stats_from_matches.isna().sum()

timestamp                             0
date_GMT                              0
attendance                            0
home_team_name                        0
away_team_name                        0
referee                               0
Game Week                             0
Pre-Match PPG (Home)                  0
Pre-Match PPG (Away)                  0
home_ppg                              0
away_ppg                              0
home_team_goal_count                  0
away_team_goal_count                  0
total_goal_count                      0
total_goals_at_half_time              0
home_team_goal_count_half_time        0
away_team_goal_count_half_time        0
```

```
home_team_goal_timings                       88
away_team_goal_timings                      119
home_team_yellow_cards                        0
home_team_red_cards                           0
away_team_yellow_cards                        0
away_team_red_cards                           0
home_team_first_half_cards                    0
home_team_second_half_cards                   0
away_team_first_half_cards                    0
away_team_second_half_cards                   0
home_team_shots                               0
away_team_shots                               0
home_team_shots_on_target                     0
away_team_shots_on_target                     0
home_team_shots_off_target                    0
away_team_shots_off_target                    0
home_team_fouls                               0
away_team_fouls                               0
home_team_possession                          0
away_team_possession                          0
Home Team Pre-Match xG                        0
Away Team Pre-Match xG                        0
team_a_xg                                     0
team_b_xg                                     0
average_goals_per_match_pre_match             0
stadium_name                                  0
dtype: int64
```

#it seems the home goals and away goals timings have a lot of nulls

stats_from_matches.head()

```
    timestamp            date_GMT    attendance         home_team_name  \
0  1533927600    Aug 10 2018 - 7:00pm        74439     Manchester United
1  1533987000   Aug 11 2018 - 11:30am        51749       Newcastle United
2  1533996000    Aug 11 2018 - 2:00pm        10353        AFC Bournemouth
3  1533996000    Aug 11 2018 - 2:00pm        24821                 Fulham
4  1533996000    Aug 11 2018 - 2:00pm        24121        Huddersfield Town

      away_team_name            referee  Game Week  Pre-Match PPG (Home)
\
0     Leicester City     Andre Marriner          1                   0.0

1  Tottenham Hotspur    Martin Atkinson          1                   0.0

2        Cardiff City        Kevin Friend          1                   0.0

3      Crystal Palace          Mike Dean          1                   0.0

4             Chelsea     Chris Kavanagh          1                   0.0
```

```
    Pre-Match PPG (Away)   home_ppg   ...    home_team_fouls
away_team_fouls  \
0                    0.0       1.89   ...                 11
8
1                    0.0       1.32   ...                 11
12
2                    0.0       1.53   ...                 11
9
3                    0.0       1.11   ...                  9
11
4                    0.0       0.47   ...                  9
8

    home_team_possession   away_team_possession   Home Team Pre-Match xG
\
0                      46                     54                      0.0

1                      40                     60                      0.0

2                      62                     38                      0.0

3                      66                     34                      0.0

4                      37                     63                      0.0


    Away Team Pre-Match xG   team_a_xg  team_b_xg   \
0                      0.0        1.25       1.46
1                      0.0        1.18       1.58
2                      0.0        1.50       0.89
3                      0.0        1.67       1.81
4                      0.0        0.90       1.17

   average_goals_per_match_pre_match   \
0                                0.0
1                                0.0
2                                0.0
3                                0.0
4                                0.0


                                stadium_name
0                     Old Trafford (Manchester)
1           St. James' Park (Newcastle upon Tyne)
2          Vitality Stadium (Bournemouth- Dorset)
3                        Craven Cottage (London)
4  John Smith's Stadium (Huddersfield- West Yorks...

[5 rows x 43 columns]

# We are going to have a look at some_data and try and clean it
```

```
matches_test=stats_from_matches.loc[0:20,'home_team_goal_timings':'awa
y_team_goal_timings']

matches_test
```

	home_team_goal_timings	away_team_goal_timings
0	3,83	90'2
1	11	8,18
2	24,90'1	NaN
3	NaN	41,79
4	NaN	34,45,80
5	35,54	NaN
6	44,80	17,67
7	19,45'2,53,88	NaN
8	NaN	NaN
9	NaN	14,64
10	NaN	NaN
11	15,31	54
12	29,45	NaN
13	43,74,77	52
14	33	60,66
15	9,20,81	37,41
16	6	3,48,51
17	25,31,35,48,75,84	43
18	25,27,44	34,90'5
19	NaN	45,90'3
20	57	69

```
#okay so we see that there are some NaN's- as there were no goals
scored-thing is those matches are still useful sp we can keep that
data

#What we dp want to focus is splitting those values into commas into
different columns

stats_from_matches.home_team_goal_count.max()

6

#max of 6 goals

#so we want to split the home_goal_timings, first we use a test
dataset

matches_test_2=matches_test["home_team_goal_timings"].str.split(",",n=
6,expand=True)

matches_test_2
```

	0	1	2	3	4	5
0	3	83	None	None	None	None
1	NaN	NaN	NaN	NaN	NaN	NaN

```
2     24    90'1    None    None    None    None
3     NaN    NaN    NaN    NaN    NaN    NaN
4     NaN    NaN    NaN    NaN    NaN    NaN
5     35    54    None    None    None    None
6     44    80    None    None    None    None
7     19    45'2    53    88    None    None
8     NaN    NaN    NaN    NaN    NaN    NaN
9     NaN    NaN    NaN    NaN    NaN    NaN
10    NaN    NaN    NaN    NaN    NaN    NaN
11    15    31    None    None    None    None
12    29    45    None    None    None    None
13    43    74    77    None    None    None
14    NaN    NaN    NaN    NaN    NaN    NaN
15     9    20    81    None    None    None
16    NaN    NaN    NaN    NaN    NaN    NaN
17    25    31    35    48    75    84
18    25    27    44    None    None    None
19    NaN    NaN    NaN    NaN    NaN    NaN
20    NaN    NaN    NaN    NaN    NaN    NaN
```

```
#So deadling with the 45 and 90 bits

# we want to say if there's a 45'2 or 90'2 we want to strip, so we can
use the apostrophe

# So first we splt the goals into separate coluns

matches_test_2[0]=matches_test_2[0].str.split("'").str[0]

matches_test_2
```

```
       0     1     2     3     4     5
0      3    83    None    None    None    None
1     NaN    NaN    NaN    NaN    NaN    NaN
2     24    90'1    None    None    None    None
3     NaN    NaN    NaN    NaN    NaN    NaN
4     NaN    NaN    NaN    NaN    NaN    NaN
5     35    54    None    None    None    None
6     44    80    None    None    None    None
7     19    45'2    53    88    None    None
8     NaN    NaN    NaN    NaN    NaN    NaN
9     NaN    NaN    NaN    NaN    NaN    NaN
10    NaN    NaN    NaN    NaN    NaN    NaN
11    15    31    None    None    None    None
12    29    45    None    None    None    None
13    43    74    77    None    None    None
14    NaN    NaN    NaN    NaN    NaN    NaN
15     9    20    81    None    None    None
16    NaN    NaN    NaN    NaN    NaN    NaN
17    25    31    35    48    75    84
18    25    27    44    None    None    None
```

```
19   NaN    NaN    NaN    NaN    NaN    NaN
20   NaN    NaN    NaN    NaN    NaN    NaN
```

```python
# now we do it on the actual dataset

stats_matches_2=stats_from_matches["home_team_goal_timings"].str.split
(",",n=6,expand=True)

stats_matches_2[1]
```

```
0          83
1         NaN
2        90'1
3         NaN
4         NaN
          ...
375        81
376       NaN
377       NaN
378        75
379       NaN
Name: 1, Length: 380, dtype: object
```

```python
#first I'd like to rename the columns

dict={0:'Home_team_goal_timings_1',1:'Home_team_goal_timings_2',2:'Hom
e_team_goal_timings_3',

3:'Home_team_goal_timings_4',4:'Home_team_goal_timings_5',5:'Home_team
_goal_timings_6',
     }

stats_matches_2.rename(columns=dict,inplace=True)

stats_matches_2['Home_team_goal_timings_6']
```

```
0         None
1          NaN
2         None
3          NaN
4          NaN
          ...
375       None
376        NaN
377        NaN
378       None
379        NaN
Name: Home_team_goal_timings_6, Length: 380, dtype: object
```

```python
# Now we clean up the apostophes
```

```python
stats_matches_2['Home_team_goal_timings_1']=stats_matches_2['Home_team
_goal_timings_1'].str.split("'").str[0]

stats_matches_2['Home_team_goal_timings_2']=stats_matches_2['Home_team
_goal_timings_2'].str.split("'").str[0]

stats_matches_2['Home_team_goal_timings_3']=stats_matches_2['Home_team
_goal_timings_3'].str.split("'").str[0]

stats_matches_2['Home_team_goal_timings_4']=stats_matches_2['Home_team
_goal_timings_4'].str.split("'").str[0]

stats_matches_2['Home_team_goal_timings_5']=stats_matches_2['Home_team
_goal_timings_5'].str.split("'").str[0]

stats_matches_2['Home_team_goal_timings_6']=stats_matches_2['Home_team
_goal_timings_6'].str.split("'").str[0]
```

```python
#Now we need to do the away team goal_timigs, should be a very similar
process

stats_from_matches['away_team_goal_timings'].head()
```

```
0          90'2
1          8,18
2           NaN
3         41,79
4      34,45,80
Name: away_team_goal_timings, dtype: object
```

```python
stats_matches_3=stats_from_matches["away_team_goal_timings"].str.split
(",",n=6,expand=True)

stats_matches_3
```

```
          0       1      2      3      4      5
0      90'2    None   None   None   None   None
1         8      18   None   None   None   None
2       NaN     NaN    NaN    NaN    NaN    NaN
3        41      79   None   None   None   None
4        34      45     80   None   None   None
..      ...     ...    ...    ...    ...    ...
375     NaN     NaN    NaN    NaN    NaN    NaN
376      23      54   None   None   None   None
377     NaN     NaN    NaN    NaN    NaN    NaN
378      69      72   None   None   None   None
379      15      39     71     78   None   None

[380 rows x 6 columns]
```

```python
away_dict={0:'away_team_goal_timings_1',1:'away_team_goal_timings_2',2
:'away_team_goal_timings_3',
```

```
3:'away_team_goal_timings_4',4:'away_team_goal_timings_5',5:'away_team
_goal_timings_6',
      }

stats_matches_3.rename(columns=away_dict,inplace=True)

# Now we clean up the apostophes

stats_matches_3.head()

   away_team_goal_timings_1 away_team_goal_timings_2
away_team_goal_timings_3   \
0                     90'2                     None
None
1                        8                       18
None
2                      NaN                      NaN
NaN
3                       41                       79
None
4                       34                       45
80


   away_team_goal_timings_4 away_team_goal_timings_5
away_team_goal_timings_6
0                     None                     None
None
1                     None                     None
None
2                      NaN                      NaN
NaN
3                     None                     None
None
4                     None                     None
None

stats_matches_3['away_team_goal_timings_6'].count()

1

stats_matches_3['away_team_goal_timings_1']=stats_matches_3['away_team
_goal_timings_1'].str.split("'").str[0]

stats_matches_3['away_team_goal_timings_2']=stats_matches_3['away_team
_goal_timings_2'].str.split("'").str[0]

stats_matches_3['away_team_goal_timings_3']=stats_matches_3['away_team
_goal_timings_3'].str.split("'").str[0]

stats_matches_3['away_team_goal_timings_4']=stats_matches_3['away_team
_goal_timings_4'].str.split("'").str[0]
```

```python
stats_matches_3['away_team_goal_timings_5']=stats_matches_3['away_team
_goal_timings_5'].str.split("'").str[0]

stats_matches_3['away_team_goal_timings_6']=stats_matches_3['away_team
_goal_timings_6'].str.split("'").str[0]

stats_matches_3.head()
```

```
  away_team_goal_timings_1 away_team_goal_timings_2
away_team_goal_timings_3  \
0                       90                     None
None
1                        8                       18
None
2                      NaN                      NaN
NaN
3                       41                       79
None
4                       34                       45
80

  away_team_goal_timings_4 away_team_goal_timings_5
away_team_goal_timings_6
0                     None                     None
None
1                     None                     None
None
2                      NaN                      NaN
NaN
3                     None                     None
None
4                     None                     None
None
```

```python
#away_goal_timings are in stats_matches_3
```

```python
stats_matches_2.head()
```

```
  Home_team_goal_timings_1 Home_team_goal_timings_2
Home_team_goal_timings_3  \
0                        3                       83
None
1                      NaN                      NaN
NaN
2                       24                       90
None
3                      NaN                      NaN
NaN
4                      NaN                      NaN
NaN
```

```
   Home_team_goal_timings_4 Home_team_goal_timings_5
Home_team_goal_timings_6
0                         None                         None
None
1                          NaN                          NaN
NaN
2                         None                         None
None
3                          NaN                          NaN
NaN
4                          NaN                          NaN
NaN
```

```
stats_matches_3.head()
```

```
   away_team_goal_timings_1 away_team_goal_timings_2
away_team_goal_timings_3  \
0                         90                         None
None
1                          8                           18
None
2                        NaN                          NaN
NaN
3                         41                           79
None
4                         34                           45
80
```

```
   away_team_goal_timings_4 away_team_goal_timings_5
away_team_goal_timings_6
0                         None                         None
None
1                         None                         None
None
2                          NaN                          NaN
NaN
3                         None                         None
None
4                         None                         None
None
```

```
stats_from_matches['away_team_goal_timings']
```

```
0              90'2
1              8,18
2               NaN
3             41,79
4          34,45,80
           ...
375             NaN
```

```
376        23,54
377           55
378        69,72
379    15,39,71,78
Name: away_team_goal_timings, Length: 380, dtype: object

all_stats=[stats_from_matches,stats_matches_2,stats_matches_3]

completed_stats=pd.concat(all_stats,axis=1)

completed_stats
```

	timestamp	date_GMT	attendance	home_team_name
0	1533927600	Aug 10 2018 - 7:00pm	74439	Manchester United
1	1533987000	Aug 11 2018 - 11:30am	51749	Newcastle United
2	1533996000	Aug 11 2018 - 2:00pm	10353	AFC Bournemouth
3	1533996000	Aug 11 2018 - 2:00pm	24821	Fulham
4	1533996000	Aug 11 2018 - 2:00pm	24121	Huddersfield Town
..	...	...	...	...
375	1557669600	May 12 2019 - 2:00pm	53331	Liverpool
376	1557669600	May 12 2019 - 2:00pm	74457	Manchester United
377	1557669600	May 12 2019 - 2:00pm	30367	Southampton
378	1557669600	May 12 2019 - 2:00pm	60124	Tottenham Hotspur
379	1557669600	May 12 2019 - 2:00pm	20067	Watford

	away_team_name	referee	Game Week
0	Leicester City	Andre Marriner	1
1	Tottenham Hotspur	Martin Atkinson	1
2	Cardiff City	Kevin Friend	1
3	Crystal Palace	Mike Dean	1
4	Chelsea	Chris Kavanagh	1
..	...	...	...
375	Wolverhampton Wanderers	Martin Atkinson	38
376	Cardiff City	Jonathan Moss	38
377	Huddersfield Town	Lee Probert	38
378	Everton	Andre Marriner	38
379	West Ham United	Chris Kavanagh	38

```
     Pre-Match PPG (Home)  Pre-Match PPG (Away)  home_ppg  ...  \
```

```
0                   0.00               0.00   1.89   ...
1                   0.00               0.00   1.32   ...
2                   0.00               0.00   1.53   ...
3                   0.00               0.00   1.11   ...
4                   0.00               0.00   0.47   ...
..                   ...                ...    ...   ...
375                 2.78               1.28   2.79   ...
376                 2.00               0.61   1.89   ...
377                 1.22               0.33   1.21   ...
378                 2.06               1.06   2.00   ...
379                 1.50               1.00   1.42   ...

     Home_team_goal_timings_3  Home_team_goal_timings_4  \
0                        None                      None
1                        None                      None
2                        None                      None
3                        None                      None
4                        None                      None
..                        ...                       ...
375                      None                      None
376                      None                      None
377                      None                      None
378                      None                      None
379                      None                      None

     Home_team_goal_timings_5  Home_team_goal_timings_6  \
0                        None                      None
1                        None                      None
2                        None                      None
3                        None                      None
4                        None                      None
..                        ...                       ...
375                      None                      None
376                      None                      None
377                      None                      None
378                      None                      None
379                      None                      None

     away_team_goal_timings_1  away_team_goal_timings_2  \
0                          90                      None
1                           8                        18
2                         NaN                       NaN
3                          41                        79
4                          34                        45
..                        ...                       ...
375                       NaN                       NaN
376                        23                        54
377                       NaN                       NaN
378                        69                        72
379                        15                        39
```

```
      away_team_goal_timings_3 away_team_goal_timings_4  \
0                         None                       None
1                         None                       None
2                          NaN                       None
3                         None                       None
4                           80                       None
..                         ...                        ...
375                        NaN                        NaN
376                       None                       None
377                        NaN                        NaN
378                       None                       None
379                         71                         78

      away_team_goal_timings_5 away_team_goal_timings_6
0                         None                       None
1                         None                       None
2                         None                       None
3                         None                       None
4                         None                       None
..                         ...                        ...
375                        NaN                        NaN
376                        NaN                        NaN
377                        NaN                        NaN
378                        NaN                        NaN
379                        NaN                        NaN

[380 rows x 55 columns]
```

completed_stats['home_team_goal_timings']

```
0          3,83
1            11
2       24,90'1
3           NaN
4           NaN
         ...
375       17,81
376         NaN
377          41
378        3,75
379          46
Name: home_team_goal_timings, Length: 380, dtype: object
```

completed_stats.drop(['home_team_goal_timings','away_team_goal_timings'],axis=1,inplace=True)

Now the data is clean, we move onto the EDA

Exploratory Data Analysis

Let's look at the teams datasets

```
overall_teams=pd.concat([prem_team_stats_1,prem_team_stats_2],axis=1)

overall_teams.head()
```

```
        common_name  wins  wins_home  wins_away  draws  draws_home  \
0            Arsenal    21         14          7      7           3
1  Tottenham Hotspur    23         12         11      2           2
2    Manchester City    32         18         14      2           0
3     Leicester City    15          8          7      7           3
4     Crystal Palace    14          5          9      7           5

   draws_away  losses  losses_home  losses_away  ...  wins_2h_percentage_home  \
0           4      10            2            8  ...                       68
1           0      13            5            8  ...                       57
2           2       4            1            3  ...                       79
3           4      16            8            8  ...                       42
4           2      17            9            8  ...                       21

   wins_2h_percentage_away  draws_2h_home  draws_2h_away  losses_2h_home  \
0                       37              4              5               2
1                       26              5              7               3
2                       53              4              6               0
3                       63              7              4               4
4                       32              6              6               9

   losses_2h_away  clean_sheets_2h_home  clean_sheets_2h_away  \
0               7                    12                     6
1               7                    11                     9
2               3                    14                    13
3               3                    13                     9
```

| 4 | 7 | 8 | 6 |

	failed_to_score_2h_home	failed_to_score_2h_away
0	4	10
1	5	9
2	2	7
3	9	4
4	10	5

[5 rows x 265 columns]

overall_teams['common_name']

	common_name	common_name
0	Arsenal	Arsenal
1	Tottenham Hotspur	Tottenham Hotspur
2	Manchester City	Manchester City
3	Leicester City	Leicester City
4	Crystal Palace	Crystal Palace
5	Everton	Everton
6	Burnley	Burnley
7	Southampton	Southampton
8	AFC Bournemouth	AFC Bournemouth
9	Manchester United	Manchester United
10	Liverpool	Liverpool
11	Chelsea	Chelsea
12	West Ham United	West Ham United
13	Watford	Watford
14	Newcastle United	Newcastle United
15	Cardiff City	Cardiff City
16	Fulham	Fulham
17	Brighton & Hove Albion	Brighton & Hove Albion
18	Huddersfield Town	Huddersfield Town
19	Wolverhampton Wanderers	Wolverhampton Wanderers

overall_teams_2=overall_teams.T.drop_duplicates().T

overall_teams_2['common_name']

0	Arsenal
1	Tottenham Hotspur
2	Manchester City
3	Leicester City
4	Crystal Palace
5	Everton
6	Burnley
7	Southampton
8	AFC Bournemouth
9	Manchester United
10	Liverpool
11	Chelsea

```
12          West Ham United
13                  Watford
14        Newcastle United
15             Cardiff City
16                   Fulham
17    Brighton & Hove Albion
18        Huddersfield Town
19    Wolverhampton Wanderers
Name: common_name, dtype: object
```

Let's see who scored the most goals overall:

```
overall_teams_2.groupby('common_name')
['goals_scored'].sum().sort_values(ascending=False)
```

```
common_name
Manchester City             95
Liverpool                   89
Arsenal                     73
Tottenham Hotspur           67
Manchester United           65
Chelsea                     63
AFC Bournemouth             56
Everton                     54
West Ham United             52
Watford                     52
Leicester City              51
Crystal Palace              51
Wolverhampton Wanderers     47
Southampton                 45
Burnley                     45
Newcastle United            42
Brighton & Hove Albion      35
Fulham                      34
Cardiff City                34
Huddersfield Town           22
Name: goals_scored, dtype: object
```

```
# We can see that man city and Liverpool by far the best goal scoring
teams, with huderssfield Town doing very poor

# compare against xg
```

Who is the most wasteful?

```
diff_g_vs_xg=overall_teams_2['goals_scored']/38 -
overall_teams_2['xg_for_avg_overall']

diff_g_vs_xg
```

```
0      0.491053
1      0.183158
2          0.32
3     -0.217895
4     -0.017895
5     -0.028947
6      0.284211
7     -0.165789
8      0.123684
9      0.080526
10     0.442105
11    -0.052105
12     0.018421
13     0.208421
14     0.005263
15    -0.175263
16    -0.245263
17     0.021053
18    -0.361053
19    -0.123158
dtype: object
```

```python
overall_teams_2['diff_g_vs_xg']=overall_teams_2['goals_scored']/38 -
overall_teams_2['xg_for_avg_overall']

overall_teams_2[['common_name','diff_g_vs_xg']].sort_values(by=['diff_
g_vs_xg'],ascending=False)
```

	common_name	diff_g_vs_xg
0	Arsenal	0.491053
10	Liverpool	0.442105
2	Manchester City	0.32
6	Burnley	0.284211
13	Watford	0.208421
1	Tottenham Hotspur	0.183158
8	AFC Bournemouth	0.123684
9	Manchester United	0.080526
17	Brighton & Hove Albion	0.021053
12	West Ham United	0.018421
14	Newcastle United	0.005263
4	Crystal Palace	-0.017895
5	Everton	-0.028947
11	Chelsea	-0.052105
19	Wolverhampton Wanderers	-0.123158
7	Southampton	-0.165789
15	Cardiff City	-0.175263
3	Leicester City	-0.217895
16	Fulham	-0.245263
18	Huddersfield Town	-0.361053

```python
overall_teams_2['extra_goals_scored']=overall_teams_2['diff_g_vs_xg']*
38

overall_teams_2.loc[:,
['common_name','extra_goals_scored']].sort_values(by='extra_goals_scor
ed',ascending=False)
```

```
                common_name extra_goals_scored
0                   Arsenal              18.66
10                Liverpool               16.8
2           Manchester City              12.16
6                   Burnley               10.8
13                  Watford               7.92
1         Tottenham Hotspur               6.96
8           AFC Bournemouth                4.7
9         Manchester United               3.06
17    Brighton & Hove Albion                0.8
12          West Ham United                0.7
14          Newcastle United                0.2
4            Crystal Palace              -0.68
5                   Everton               -1.1
11                  Chelsea              -1.98
19  Wolverhampton Wanderers              -4.68
7               Southampton               -6.3
15              Cardiff City              -6.66
3            Leicester City              -8.28
16                   Fulham              -9.32
18         Huddersfield Town             -13.72
```

```python
# We can see that Arenal and Liverpool overscored as did Burnely



# So what this shows, is the teams difference between the actual goals
and the expected- i.e showing the team who were most clinical

#Can see that arsenal and liverpool overscored intersting, arguably
showing that they did better than they were supposed to

# teams who scored the least, were also the least clinical, which is
unsurprising- but intesting to note chelsea's position, who were 6th
in terms of goals scored but 1th in this ranking

#Effectively those who were postive overacheived and those with
negative rankin g under acheived

# Now I want to see breakdown of goals by position grouped by the
teams

player_stats.head()
```

```
                             full_name  age birthday_GMT     position   \
0                           Aaron Rowe   22   2000-09-07      Forward
1                     Abdelhamid Sabiri   25   1996-11-28   Midfielder
4   Adrien Sebastian Perruchet Silva    33   1989-03-15   Midfielder
5                       Alberto Moreno   30   1992-07-05     Defender
6                Alex Oxlade-Chamberlain   29   1993-08-15   Midfielder

         Current Club  minutes_played_overall  minutes_played_home   \
0   Huddersfield Town                      69                   14
1   Huddersfield Town                      49                    0
4       Leicester City                      88                    8
5            Liverpool                     155                   90
6            Liverpool                      19                   19

    minutes_played_away nationality  appearances_overall   ...   \
0                    55     England                    2   ...
1                    49     Morocco                    2   ...
4                    80    Portugal                    2   ...
5                    65       Spain                    2   ...
6                     0     England                    2   ...

    conceded_per_90_overall  min_per_conceded_overall  min_per_match  \
0                      1.30                        69             35
1                      5.51                        16             25
4                      1.02                        88             44
5                      1.16                        78             78
6                      0.00                         0             10

    min_per_card_overall  min_per_assist_overall  cards_per_90_overall
\
0                      0                       0                   0.0

1                      0                       0                   0.0

4                      0                       0                   0.0

5                      0                       0                   0.0

6                      0                       0                   0.0


    rank_in_league_top_attackers  rank_in_league_top_midfielders  \
0                            NaN                             NaN
1                            NaN                             NaN
4                            NaN                             NaN
5                            NaN                             NaN
6                            NaN                             NaN

    rank_in_league_top_defenders  rank_in_club_top_scorer
0                            NaN                       31
1                            NaN                       22
```

```
4                                        NaN                          15
5                                        NaN                          17
6                                        NaN                          21

[5 rows x 44 columns]

test_3=player_stats.loc[:,['Current Club','position','goals_overall']]

grouped_test=test_3.groupby(['Current
Club','position'],as_index=False)['goals_overall'].sum()

print(grouped_test)

              Current Club      position   goals_overall
0           AFC Bournemouth     Defender               7
1           AFC Bournemouth      Forward              34
2           AFC Bournemouth   Goalkeeper               0
3           AFC Bournemouth   Midfielder              14
4                   Arsenal     Defender               7
..                      ...          ...             ...
75          West Ham United   Midfielder              26
76   Wolverhampton Wanderers    Defender              10
77   Wolverhampton Wanderers     Forward              16
78   Wolverhampton Wanderers  Goalkeeper               0
79   Wolverhampton Wanderers  Midfielder              20

[80 rows x 3 columns]
```

THe data could still be minimised , as it is hard to see insights

```
grouped_test.loc[0:40,:].sort_values(by=['goals_overall'],ascending=False)

              Current Club      position   goals_overall
5                   Arsenal      Forward              40
23                  Chelsea   Midfielder              39
1           AFC Bournemouth      Forward              34
27            Crystal Palace  Midfielder              32
13                  Burnley      Forward              29
29                  Everton      Forward              27
7                   Arsenal   Midfielder              22
33                   Fulham      Forward              20
9      Brighton & Hove Albion     Forward              18
19              Cardiff City  Midfielder              16
31                  Everton   Midfielder              15
21                  Chelsea      Forward              15
3           AFC Bournemouth   Midfielder              14
28                  Everton     Defender              11
15                  Burnley   Midfielder              10
16              Cardiff City    Defender               9
40             Leicester City    Defender               9
```

```
11    Brighton & Hove Albion   Midfielder               9
35                    Fulham   Midfielder               9
24            Crystal Palace     Defender               9
17              Cardiff City      Forward               8
8     Brighton & Hove Albion     Defender               8
39          Huddersfield Town   Midfielder               8
25            Crystal Palace      Forward               7
37          Huddersfield Town      Forward               7
0            AFC Bournemouth     Defender               7
4                    Arsenal     Defender               7
20                   Chelsea     Defender               7
36          Huddersfield Town     Defender               6
12                   Burnley     Defender               4
32                    Fulham     Defender               4
26            Crystal Palace   Goalkeeper               0
22                   Chelsea   Goalkeeper               0
18              Cardiff City   Goalkeeper               0
14                   Burnley   Goalkeeper               0
30                   Everton   Goalkeeper               0
10    Brighton & Hove Albion   Goalkeeper               0
34                    Fulham   Goalkeeper               0
6                    Arsenal   Goalkeeper               0
38          Huddersfield Town   Goalkeeper               0
2            AFC Bournemouth   Goalkeeper               0
```

#Goalkeeper seems like an unnecessary position for goals we need to investigate, so let's drop it

```python
grouped_test=test_3.groupby(['Current
Club','position'],as_index=False).apply(lambda x: x[x['position'] !=
'Goalkeeper'])
```

```python
grouped_test.groupby(['Current Club','position'],as_index=False)
['goals_overall'].sum()
```

```
            Current Club     position  goals_overall
0          AFC Bournemouth     Defender              7
1          AFC Bournemouth      Forward             34
2          AFC Bournemouth   Midfielder             14
3                  Arsenal     Defender              7
4                  Arsenal      Forward             40
5                  Arsenal   Midfielder             22
6    Brighton & Hove Albion     Defender              8
7    Brighton & Hove Albion      Forward             18
8    Brighton & Hove Albion   Midfielder              9
9                  Burnley     Defender              4
10                 Burnley      Forward             29
11                 Burnley   Midfielder             10
```

```
12              Cardiff City    Defender         9
13              Cardiff City     Forward         8
14              Cardiff City   Midfielder        16
15                   Chelsea    Defender         7
16                   Chelsea     Forward        15
17                   Chelsea   Midfielder        39
18            Crystal Palace    Defender         9
19            Crystal Palace     Forward         7
20            Crystal Palace   Midfielder        32
21                   Everton    Defender        11
22                   Everton     Forward        27
23                   Everton   Midfielder        15
24                    Fulham    Defender         4
25                    Fulham     Forward        20
26                    Fulham   Midfielder         9
27          Huddersfield Town   Defender         6
28          Huddersfield Town    Forward         7
29          Huddersfield Town  Midfielder         8
30            Leicester City    Defender         9
31            Leicester City     Forward        20
32            Leicester City   Midfielder        19
33                 Liverpool    Defender         7
34                 Liverpool     Forward        61
35                 Liverpool   Midfielder        18
36           Manchester City    Defender         6
37           Manchester City     Forward        45
38           Manchester City   Midfielder        40
39         Manchester United    Defender         3
40         Manchester United     Forward        33
41         Manchester United   Midfielder        29
42          Newcastle United    Defender         8
43          Newcastle United     Forward        26
44          Newcastle United   Midfielder         7
45               Southampton    Defender         6
46               Southampton     Forward        16
47               Southampton   Midfielder        22
48         Tottenham Hotspur    Defender         4
49         Tottenham Hotspur     Forward        30
50         Tottenham Hotspur   Midfielder        32
51                   Watford    Defender         6
52                   Watford     Forward        33
53                   Watford   Midfielder        12
54           West Ham United    Defender         3
55           West Ham United     Forward        22
56           West Ham United   Midfielder        26
57    Wolverhampton Wanderers   Defender        10
58    Wolverhampton Wanderers    Forward        16
59    Wolverhampton Wanderers  Midfielder        20

# we cam see the distribution of goals amongst positions
```

```
testing_dist_goal_pg=completed_stats.groupby(['home_team_goal_count','
home_team_name'], as_index=False)['home_team_goal_count'].count()

home_team_goal_dist=testing_dist_goal_pg.groupby('home_team_goal_count
')['home_team_goal_count'].sum()

home_team_goal_dist

home_team_goal_count
1      18
2      22
3      54
4      36
5      70
6      42
7      49
8      40
9      18
10     20
11     11
Name: home_team_goal_count, dtype: int64


print(testing_dist_goal_pg)

             home_team_name  home_team_goal_count
0            AFC Bournemouth                     4
1                    Arsenal                     1
2      Brighton & Hove Albion                    5
3                    Burnley                     3
4                Cardiff City                    7
..                      ...                   ...
89             Crystal Palace                    1
90                  Liverpool                    3
91            Manchester City                    1
92          Tottenham Hotspur                    1
93            Manchester City                    3

[94 rows x 2 columns]
```

So we know the number of goals scored by the home team per game week

```
testing_dist_goal_pg=completed_stats.groupby(['home_team_goal_count','
home_team_name'], as_index=False)
['home_team_goal_count'].value_counts()

testing_dist_goal_pg
```

```
     home_team_goal_count         home_team_name   count
0                       0        AFC Bournemouth       4
1                       0                Arsenal       1
2                       0  Brighton & Hove Albion     5
3                       0                Burnley       3
4                       0            Cardiff City     7
..                    ...                    ...     ...
89                      5          Crystal Palace     1
90                      5              Liverpool     3
91                      5         Manchester City     1
92                      5       Tottenham Hotspur     1
93                      6         Manchester City     3

[94 rows x 3 columns]
```

```
testing_dist_goal_pg_2=completed_stats.groupby('home_team_name',
as_index=False)['home_team_goal_count'].value_counts()

testing_dist_goal_pg_2
```

```
            home_team_name  home_team_goal_count   count
0           AFC Bournemouth                     2       7
1           AFC Bournemouth                     1       5
2           AFC Bournemouth                     0       4
3           AFC Bournemouth                     4       2
4           AFC Bournemouth                     3       1
..                      ...                   ...     ...
89  Wolverhampton Wanderers                     2       7
90  Wolverhampton Wanderers                     0       5
91  Wolverhampton Wanderers                     1       4
92  Wolverhampton Wanderers                     3       2
93  Wolverhampton Wanderers                     4       1

[94 rows x 3 columns]
```

```
#testing_dist_goal_pg_2 is the most useful variable, counts frequency
of times they scored a unique number of home goals



zero_goals_Scored=testing_dist_goal_pg_2.loc[testing_dist_goal_pg_2['h
ome_team_goal_count']==0]

zero_goals_Scored.sort_values(by='count',ascending=False)
```

```
            home_team_name  home_team_goal_count   count
44          Huddersfield Town                    0       9
29             Crystal Palace                    0       8
19               Cardiff City                    0       7
39                     Fulham                    0       6
84            West Ham United                    0       6
```

```
65        Newcastle United                        0    6
45          Leicester City                        0    6
61       Manchester United                        0    5
90  Wolverhampton Wanderers                       0    5
12   Brighton & Hove Albion                       0    5
80                 Watford                        0    4
2          AFC Bournemouth                        0    4
35                 Everton                        0    3
25                 Chelsea                        0    3
17                 Burnley                        0    3
72             Southampton                        0    3
76       Tottenham Hotspur                        0    3
9                  Arsenal                        0    1
54               Liverpool                        0    1
```

#So of the teams witht the most amount of games withj 0 goals scoredd, Hudderfield,Palace andCArdiff are highest

#Intersting to note that city scored in every single game they played in as a home team

Remeber this is just home team

we knoe 6 is the max scored, but let's go from 4 scored

```
four_and_over_goals_Scored=testing_dist_goal_pg_2.loc[testing_dist_goa
l_pg_2['home_team_goal_count']>=4]
```

```
four_and_over_goals_Scored.sort_values(by='count', ascending=False)
```

```
          home_team_name  home_team_goal_count  count
49              Liverpool                     4      5
57        Manchester City                     6      3
51              Liverpool                     5      3
40                 Fulham                     4      2
88        West Ham United                     4      2
63      Manchester United                     4      2
8                 Arsenal                     4      2
3         AFC Bournemouth                     4      2
37                Everton                     4      1
32         Crystal Palace                     5      1
28                Chelsea                     4      1
27                Chelsea                     5      1
22            Cardiff City                     4      1
59        Manchester City                     5      1
18                Burnley                     4      1
77      Tottenham Hotspur                     5      1
78      Tottenham Hotspur                     4      1
83                Watford                     4      1
10                Arsenal                     5      1
93  Wolverhampton Wanderers                    4      1
```

```python
# City scored 6 goals 3 times, but not 4 or 5 once

#Liverpool scored 4 goals 5 times and 5 goals 3 times

#Fulham is interesting scored 4 goals 2 times and got relegated

## so we get the leagie [osition, it is useful but maybe place it
lzter in the analysis

# Let'see the difference when we do 3 goals and over

three_and_over_goals_Scored=testing_dist_goal_pg_2.loc[testing_dist_go
al_pg_2['home_team_goal_count']>=3]

three_and_over_goals_Scored.sort_values(by='count', ascending=False)
```

	home_team_name	home_team_goal_count	count
55	Manchester City	3	7
85	West Ham United	3	5
49	Liverpool	4	5
24	Chelsea	3	4
74	Tottenham Hotspur	3	4
68	Newcastle United	3	3
52	Liverpool	3	3
57	Manchester City	6	3
62	Manchester United	3	3
51	Liverpool	5	3
71	Southampton	3	3
7	Arsenal	3	3
48	Leicester City	3	3
63	Manchester United	4	2
82	Watford	3	2
88	West Ham United	4	2
92	Wolverhampton Wanderers	3	2
3	AFC Bournemouth	4	2
40	Fulham	4	2
8	Arsenal	4	2
36	Everton	3	2
13	Brighton & Hove Albion	3	2
4	AFC Bournemouth	3	1
77	Tottenham Hotspur	5	1
10	Arsenal	5	1
83	Watford	4	1
18	Burnley	4	1
78	Tottenham Hotspur	4	1
22	Cardiff City	4	1
42	Fulham	3	1
27	Chelsea	5	1
28	Chelsea	4	1
32	Crystal Palace	5	1
59	Manchester City	5	1

```
37                   Everton                    4        1
93   Wolverhampton Wanderers                    4        1
```

If we compare league position at home, and those who scoreded at home

```python
home_goals=overall_teams_2.groupby('common_name', as_index=False)
['goals_scored_home'].sum()

home_goals.sort_values(by='goals_scored_home',ascending=False)
```

```
            common_name  goals_scored_home
12       Manchester City                 57
11             Liverpool                 55
1                Arsenal                 42
5                Chelsea                 39
16     Tottenham Hotspur                 34
13     Manchester United                 33
18       West Ham United                 32
0        AFC Bournemouth                 30
7                Everton                 30
19 Wolverhampton Wanderers               28
15           Southampton                 27
17               Watford                 26
14       Newcastle United                24
3                Burnley                 24
10        Leicester City                 24
8                 Fulham                 22
4           Cardiff City                 21
6         Crystal Palace                 19
2   Brighton & Hove Albion               19
9       Huddersfield Town                10
```

```python
overall_teams_2[['common_name','league_position_home']].sort_values(by
='league_position_home',ascending=True)
```

```
            common_name  league_position_home
2        Manchester City                     1
10             Liverpool                     2
0                Arsenal                     3
11               Chelsea                     4
1      Tottenham Hotspur                     5
9      Manchester United                     6
5                Everton                     7
19 Wolverhampton Wanderers                   8
12       West Ham United                     9
8        AFC Bournemouth                    10
3         Leicester City                    11
13               Watford                    12
14       Newcastle United                   13
7            Southampton                    14
```

```
6                   Burnley                    15
17   Brighton & Hove Albion                    16
16                    Fulham                    17
4             Crystal Palace                    18
15              Cardiff City                    19
18          Huddersfield Town                   20
```

```python
# We dan already tell there is a strong correlationn betweem
goals_scored at home- let's tes it:

# perhaps the datatype of the object is wrong

overall_teams_2[['goals_scored_home','league_position_home']].info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20 entries, 0 to 19
Data columns (total 2 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   goals_scored_home      20 non-null     object
 1   league_position_home   20 non-null     object
dtypes: object(2)
memory usage: 448.0+ bytes
```

```python
#Thus we need to convert it

overall_teams_2[['goals_scored_home','league_position_home']]=overall_
teams_2[['goals_scored_home','league_position_home']].astype(str).asty
pe(int)

overall_teams_2[['goals_scored_home','league_position_home']].info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20 entries, 0 to 19
Data columns (total 2 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   goals_scored_home      20 non-null     int32
 1   league_position_home   20 non-null     int32
dtypes: int32(2)
memory usage: 288.0 bytes
```

```python
print(overall_teams_2['goals_scored_home'].corr(overall_teams_2['leagu
e_position_home']))
```

```
-0.9048863631679042
```

```python
#Strong correlation betweem goals_Scored at home and league_position
at home

# After changing the datatype, we see it has a stong correlation, the
negative is also expected as ranking is 1 as highest
```

```python
# I do want to see the correlation in overall league position and
goals_scored_home

overall_teams_2['league_position'].info()
```

```
<class 'pandas.core.series.Series'>
RangeIndex: 20 entries, 0 to 19
Series name: league_position
Non-Null Count  Dtype
--------------  -----
20 non-null     object
dtypes: object(1)
memory usage: 288.0+ bytes
```

```python
#we also need to convert it

overall_teams_2['league_position']=overall_teams['league_position'].as
type(str).astype(int)

print(overall_teams_2['goals_scored_home'].corr(overall_teams_2['leagu
e_position']))
```

```
-0.8486200049811436
```

```python
# Still a strong correlation, but unsurprsingly less- teams often are
talked about how strong home advantage is

# Let's look at overall goals vs league position

overall_teams_2['goals_scored']
```

```
0     73
1     67
2     95
3     51
4     51
5     54
6     45
7     45
8     56
9     65
10    89
11    63
12    52
13    52
14    42
15    34
16    34
17    35
18    22
```

```
19     47
Name: goals_scored, dtype: object
```

```python
#need to convert it
```

```python
overall_teams_2['goals_scored']=overall_teams_2['goals_scored'].astype
(str).astype(int)
```

```python
print(overall_teams_2['goals_scored'].corr(overall_teams_2['league_pos
ition']))
```

```
-0.9004072621949807
```

```python
#very stong correlation between overall goals scored and
league_postion
```

Now we need to find the value counts for away_goals

```python
overall_teams_2['goals_scored_away']
```

```
0      31
1      33
2      38
3      27
4      32
5      24
6      21
7      18
8      26
9      32
10     34
11     24
12     20
13     26
14     18
15     13
16     12
17     16
18     12
19     19
Name: goals_scored_away, dtype: object
```

```python
away_goals=overall_teams_2.groupby('common_name', as_index=False)
['goals_scored_away'].sum()
```

```python
away_goals.sort_values(by='goals_scored_away',ascending=False)
```

```
            common_name  goals_scored_away
12       Manchester City                 38
11             Liverpool                 34
16     Tottenham Hotspur                 33
```

```
13          Manchester United                    32
6             Crystal Palace                    32
1                    Arsenal                    31
10            Leicester City                    27
17                   Watford                    26
0            AFC Bournemouth                    26
7                    Everton                    24
5                    Chelsea                    24
3                    Burnley                    21
18          West Ham United                    20
19  Wolverhampton Wanderers                    19
14          Newcastle United                    18
15               Southampton                    18
2      Brighton & Hove Albion                   16
4               Cardiff City                    13
9          Huddersfield Town                    12
8                      Fulham                   12
```

#again city and liverpool at the top, but interstomg to see Crystal Place and Tottenham Hotspur


```
testing_dist_goal_pg_away=completed_stats.groupby('away_team_name',
as_index=False)['away_team_goal_count'].value_counts()

testing_dist_goal_pg_away
```

```
            away_team_name  away_team_goal_count   count
0            AFC Bournemouth                    0       8
1            AFC Bournemouth                    1       4
2            AFC Bournemouth                    3       3
3            AFC Bournemouth                    2       2
4            AFC Bournemouth                    4       1
..                      ...                   ...     ...
86          West Ham United                    4       1
87  Wolverhampton Wanderers                    1       9
88  Wolverhampton Wanderers                    0       6
89  Wolverhampton Wanderers                    2       2
90  Wolverhampton Wanderers                    3       2

[91 rows x 3 columns]
```

```
zero_goals_Scored_away=testing_dist_goal_pg_away.loc[testing_dist_goal
_pg_away['away_team_goal_count']==0]

zero_goals_Scored_away.sort_values(by='count',ascending=False)
```

```
            away_team_name  away_team_goal_count   count
18               Cardiff City                    0      10
39          Huddersfield Town                    0      10
```

```
14            Burnley                    0    9
36             Fulham                    0    9
0       AFC Bournemouth                  0    8
11  Brighton & Hove Albion               0    8
82       West Ham United                 0    8
65      Newcastle United                 0    7
21             Chelsea                   0    7
31             Everton                   0    7
68          Southampton                  0    7
77             Watford                   0    6
88  Wolverhampton Wanderers              0    6
74      Tottenham Hotspur                0    3
8              Arsenal                   0    3
28        Crystal Palace                 0    3
61      Manchester United                0    2
57        Manchester City                0    2
51           Liverpool                   0    2
45        Leicester City                 0    2
```

 # not suprised by the teams, who failed to score away- teams who were lowest in the table

```
overall_teams_2[['common_name','league_position_away']].sort_values(by='league_position_away',ascending=True)
```

```
            common_name league_position_away
2        Manchester City                    1
10             Liverpool                    2
1      Tottenham Hotspur                    3
9      Manchester United                    4
11               Chelsea                    5
4         Crystal Palace                    6
3         Leicester City                    7
0               Arsenal                     8
13               Watford                    9
19  Wolverhampton Wanderers               10
12       West Ham United                   11
5              Everton                     12
14      Newcastle United                   13
6              Burnley                     14
7          Southampton                     15
8       AFC Bournemouth                    16
15          Cardiff City                   17
17  Brighton & Hove Albion                18
18      Huddersfield Town                 19
16             Fulham                      20
```

Make sure we use away_league_table

```
completed_stats['away_team_goal_count'].max()
```

```
6
```

The most a team scored away from home is 6, so we will start with a
similar process with the home goals analysis

```
over_three_goals_Scored_away=testing_dist_goal_pg_away.loc[testing_dis
t_goal_pg_away['away_team_goal_count']>=3]
```

```
over_three_goals_Scored_away.sort_values(by='away_team_goal_count',asc
ending=False)
```

	away_team_name	away_team_goal_count	count
76	Tottenham Hotspur	6	1
81	Watford	5	1
35	Everton	5	1
63	Manchester United	5	1
58	Manchester City	5	1
10	Arsenal	5	1
5	AFC Bournemouth	5	1
25	Chelsea	4	1
30	Crystal Palace	4	1
86	West Ham United	4	1
47	Leicester City	4	1
4	AFC Bournemouth	4	1
52	Liverpool	4	1
67	Newcastle United	4	1
56	Manchester City	4	2
84	West Ham United	3	2
80	Watford	3	2
75	Tottenham Hotspur	3	3
71	Southampton	3	1
2	AFC Bournemouth	3	3
50	Liverpool	3	4
62	Manchester United	3	2
55	Manchester City	3	3
46	Leicester City	3	1
42	Huddersfield Town	3	1
34	Everton	3	1
27	Crystal Palace	3	5
24	Chelsea	3	2
16	Burnley	3	2
9	Arsenal	3	2
90	Wolverhampton Wanderers	3	2

#Didn;t expect to see bournemouth everton,watford get this many goals-
but it is a one off, as with most teams

```
over_three_goals_Scored_away.sort_values(by='count',ascending=False)
```

	away_team_name	away_team_goal_count	count
27	Crystal Palace	3	5

50	Liverpool	3	4
75	Tottenham Hotspur	3	3
55	Manchester City	3	3
2	AFC Bournemouth	3	3
16	Burnley	3	2
9	Arsenal	3	2
84	West Ham United	3	2
80	Watford	3	2
62	Manchester United	3	2
24	Chelsea	3	2
56	Manchester City	4	2
90	Wolverhampton Wanderers	3	2
67	Newcastle United	4	1
86	West Ham United	4	1
25	Chelsea	4	1
81	Watford	5	1
10	Arsenal	5	1
76	Tottenham Hotspur	6	1
5	AFC Bournemouth	5	1
71	Southampton	3	1
42	Huddersfield Town	3	1
46	Leicester City	3	1
30	Crystal Palace	4	1
58	Manchester City	5	1
34	Everton	3	1
35	Everton	5	1
52	Liverpool	4	1
4	AFC Bournemouth	4	1
47	Leicester City	4	1
63	Manchester United	5	1

```python
# As it is of a much less freuency compared to home_games, it is hard
to pick trends, even the so-called 'best teams' did not high numbers
frequenly

#the highest was across all games was 36 by man city- maybe best to
look at 3 an under and see the data there

three_or_less_away_goals=testing_dist_goal_pg_away.query('away_team_go
al_count <=3 & away_team_goal_count >0')

three_or_less_away_goals
```

	away_team_name	away_team_goal_count	count
1	AFC Bournemouth	1	4
2	AFC Bournemouth	3	3
3	AFC Bournemouth	2	2
6	Arsenal	2	7
7	Arsenal	1	6
9	Arsenal	3	2

12	Brighton & Hove Albion	1	6
13	Brighton & Hove Albion	2	5
15	Burnley	2	7
16	Burnley	3	2
17	Burnley	1	1
19	Cardiff City	1	5
20	Cardiff City	2	4
22	Chelsea	2	5
23	Chelsea	1	4
24	Chelsea	3	2
26	Crystal Palace	1	7
27	Crystal Palace	3	5
29	Crystal Palace	2	3
32	Everton	2	6
33	Everton	1	4
34	Everton	3	1
37	Fulham	1	8
38	Fulham	2	2
40	Huddersfield Town	1	7
41	Huddersfield Town	2	1
42	Huddersfield Town	3	1
43	Leicester City	1	10
44	Leicester City	2	5
46	Leicester City	3	1
48	Liverpool	2	6
49	Liverpool	1	6
50	Liverpool	3	4
53	Manchester City	1	6
54	Manchester City	2	5
55	Manchester City	3	3
59	Manchester United	1	7
60	Manchester United	2	7
62	Manchester United	3	2
64	Newcastle United	1	8
66	Newcastle United	2	3
69	Southampton	1	7
70	Southampton	2	4
71	Southampton	3	1
72	Tottenham Hotspur	2	6
73	Tottenham Hotspur	1	6
75	Tottenham Hotspur	3	3
78	Watford	2	5
79	Watford	1	5
80	Watford	3	2
83	West Ham United	1	6
84	West Ham United	3	2
85	West Ham United	2	2
87	Wolverhampton Wanderers	1	9

```
89  Wolverhampton Wanderers                    2      2
90  Wolverhampton Wanderers                    3      2
```

#We see that very few teams reach 3 goals and the majority of teams score betwen 1 and 2 goals

```
three_or_less_away_goals.groupby('away_team_goal_count')
['count'].sum()

away_team_goal_count
1    122
2     87
3     36
Name: count, dtype: int64
```

```
dist_away_goal_count=
testing_dist_goal_pg_away.groupby('away_team_goal_count')
['count'].sum()

dist_away_goal_count
```

```
away_team_goal_count
0    119
1    122
2     87
3     36
4      9
5      6
6      1
Name: count, dtype: int64
```

Majority of teams score either none or 1

```
new_dist_away_goal_counts=dist_away_goal_count.reset_index(level=0)
```

```
new_dist_away_goal_counts
```

	away_team_goal_count	count
0	0	119
1	1	122
2	2	87
3	3	36
4	4	9
5	5	6
6	6	1

```
new_dist_away_goal_counts.plot(x='away_team_goal_count',y='count',kind
='bar')
```

```
<AxesSubplot:xlabel='away_team_goal_count'>
```
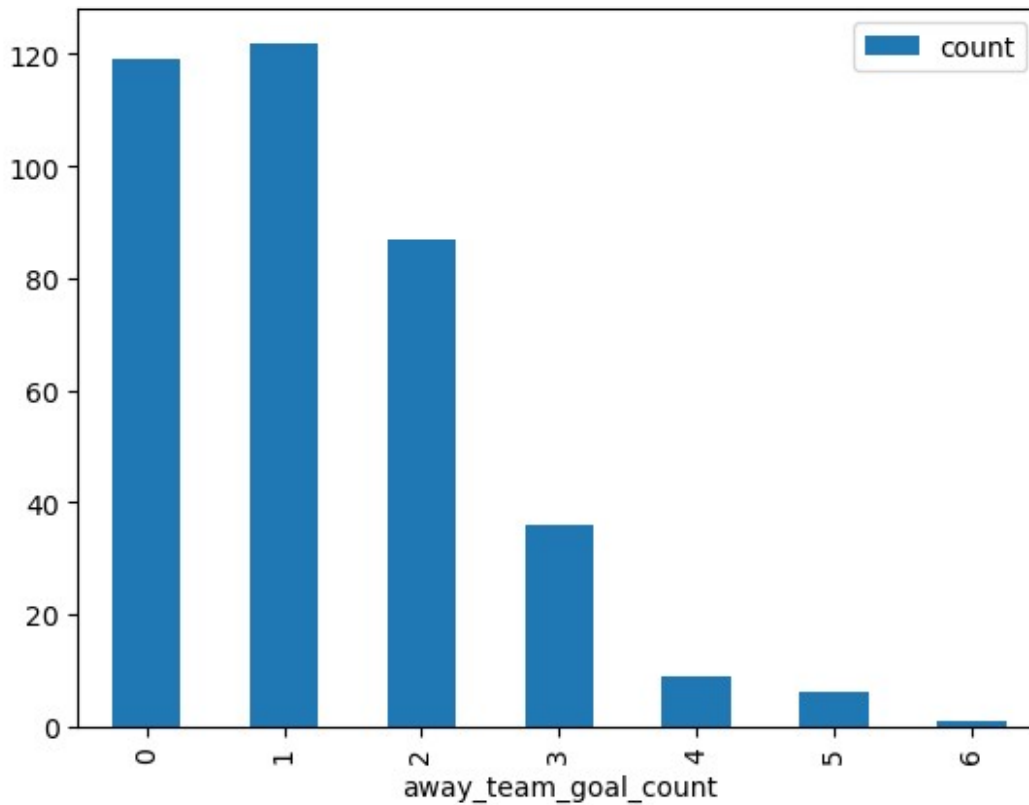
```
home_dist_away_goal_count=
testing_dist_goal_pg_2.groupby('home_team_goal_count')['count'].sum()

home_dist_away_goal_count

home_team_goal_count
0       88
1      116
2       95
3       48
4       22
5        8
6        3
Name: count, dtype: int64

home_dist_goal_counts=home_dist_away_goal_count.reset_index(level=0)

home_dist_goal_counts

    home_team_goal_count   count
0                      0      88
1                      1     116
2                      2      95
3                      3      48
4                      4      22
```

```
5                      5     8
6                      6     3
```

```
home_dist_goal_counts.plot(x='home_team_goal_count',y='count',kind='ba
r')
```

```
<AxesSubplot:xlabel='home_team_goal_count'>
```
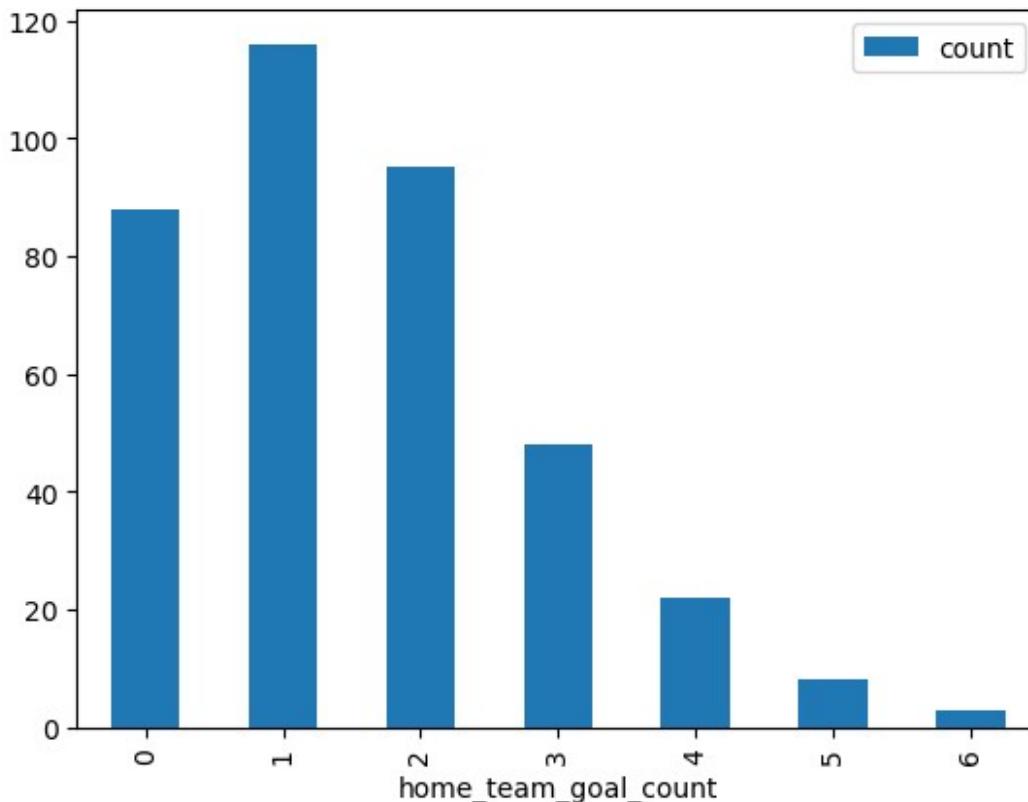


```
# we can see that teams tend to score much more awayy
```

As this in the context of scoring goals at home, we need how well do teams score goals away as well

But it should be noted, that the distribution to scoring goals at home is equivalent the goals conceded away

```
completed_stats['home_team_goal_count'].describe()
```

```
count    380.000000
mean       1.568421
std        1.312836
min        0.000000
25%        1.000000
```

```
50%        1.000000
75%        2.000000
max        6.000000
Name: home_team_goal_count, dtype: float64
```

```
completed_stats['away_team_goal_count'].describe()
```

```
count    380.000000
mean       1.252632
std        1.180031
min        0.000000
25%        0.000000
50%        1.000000
75%        2.000000
max        6.000000
Name: away_team_goal_count, dtype: float64
```

Now we do goals_conceded

```
## WHhile scoring goals is important, defence wins you titles, so now
we look at the defence of teams in the premier leagie
```

```
overall_teams_2[['common_name','league_position']].sort_values(by='lea
gue_position',ascending=True)
```

```
              common_name  league_position
2            Manchester City                1
10                 Liverpool                2
11                   Chelsea                3
1         Tottenham Hotspur                4
0                   Arsenal                5
9         Manchester United                6
19   Wolverhampton Wanderers                7
5                   Everton                8
3            Leicester City                9
12          West Ham United               10
13                   Watford               11
4            Crystal Palace               12
14          Newcastle United               13
8           AFC Bournemouth               14
6                   Burnley               15
7               Southampton               16
17    Brighton & Hove Albion               17
15              Cardiff City               18
16                    Fulham               19
18         Huddersfield Town               20
```

```
overall_teams_2.groupby('common_name')
['goals_conceded'].sum().sort_values(ascending=True)
```

```
common_name
Liverpool                      22
Manchester City                23
Tottenham Hotspur              39
Chelsea                        39
Everton                        46
Wolverhampton Wanderers        46
Leicester City                 48
Newcastle United               48
Arsenal                        51
Crystal Palace                 53
Manchester United              54
West Ham United                55
Watford                        59
Brighton & Hove Albion         60
Southampton                    65
Burnley                        68
Cardiff City                   69
AFC Bournemouth                70
Huddersfield Town              76
Fulham                         81
Name: goals_conceded, dtype: object

# So the teams with the best defences- unsuprsingly liverpool and
city, close correlation to the league table on sight
```

Who has the luckiest defence?

```
overall_teams_2[['xg_against_avg_overall','common_name']].sort_values(
by='xg_against_avg_overall',ascending=False)
```

```
    xg_against_avg_overall                 common_name
16                     1.7                      Fulham
6                     1.67                     Burnley
15                    1.63                 Cardiff City
17                    1.58       Brighton & Hove Albion
12                    1.55              West Ham United
4                     1.53                Crystal Palace
13                     1.5                     Watford
18                    1.48             Huddersfield Town
8                     1.48               AFC Bournemouth
7                     1.47                  Southampton
0                     1.44                      Arsenal
9                     1.44            Manchester United
14                    1.42              Newcastle United
1                     1.35            Tottenham Hotspur
19                    1.32      Wolverhampton Wanderers
3                     1.32               Leicester City
5                     1.25                      Everton
```

```
11                     1.14                        Chelsea
10                     0.96                      Liverpool
2                      0.79               Manchester City
```

#In terms of expected goals against,we see the teams who should've
conceded this amount of goals per game on average

```
overall_teams_2['g_conc_vs_xga']=overall_teams_2['goals_conceded']/38-
overall_teams_2['xg_against_avg_overall']
```

```
overall_teams_2[['g_conc_vs_xga','common_name']].sort_values(by=['g_co
nc_vs_xga'],ascending=False)
```

```
    g_conc_vs_xga                        common_name
18          0.52               Huddersfield Town
16      0.431579                          Fulham
8       0.362105                 AFC Bournemouth
7       0.240526                     Southampton
15      0.185789                     Cardiff City
6       0.119474                          Burnley
13      0.052632                          Watford
17     -0.001053       Brighton & Hove Albion
9      -0.018947               Manchester United
5      -0.039474                          Everton
3      -0.056842                 Leicester City
0      -0.097895                          Arsenal
12     -0.102632                 West Ham United
19     -0.109474       Wolverhampton Wanderers
11     -0.113684                          Chelsea
4      -0.135263                 Crystal Palace
14     -0.156842                 Newcastle United
2      -0.184737                 Manchester City
1      -0.323684               Tottenham Hotspur
10     -0.381053                      Liverpool
```

 # So those wih a postive values, conceded more goals than they were
expected to

 # and subsequenty negative values, suggest that they conceded less
goals than they were supposed to

#Does this mean over the course of the season, huddersfield conceded
c.19 more goals than they were supposed to

```
overall_teams_2['goals_conceded_per_match']=overall_teams_2['goals_con
ceded']/38
```

```
overall_teams_2[['common_name','xg_against_avg_overall','goals_concede
d_per_match']].sort_values(by=['xg_against_avg_overall'],ascending=
False)
```

```
                common_name xg_against_avg_overall
goals_conceded_per_match
16                     Fulham                    1.7
2.131579
6                     Burnley                   1.67
1.789474
15             Cardiff City                   1.63
1.815789
17    Brighton & Hove Albion                1.58
1.578947
12          West Ham United               1.55
1.447368
4             Crystal Palace                1.53
1.394737
13                    Watford                    1.5
1.552632
18          Huddersfield Town             1.48
2.0
8             AFC Bournemouth              1.48
1.842105
7                 Southampton               1.47
1.710526
0                     Arsenal                   1.44
1.342105
9           Manchester United            1.44
1.421053
14             Newcastle United            1.42
1.263158
1             Tottenham Hotspur           1.35
1.026316
19   Wolverhampton Wanderers         1.32
1.210526
3               Leicester City              1.32
1.263158
5                     Everton                   1.25
1.210526
11                    Chelsea                   1.14
1.026316
10                    Liverpool                 0.96
0.578947
2             Manchester City             0.79
0.605263

#Here we look at Fulham

0.43*38

16.34

# so fulham conceded an extra 16 goals than they should have
```

```
overall_teams_2['xtra_goals_conceded_overall']=overall_teams_2['g_conc
_vs_xga']*38

overall_teams_2[['common_name','xtra_goals_conceded_overall']].sort_va
lues(by='xtra_goals_conceded_overall',ascending=False)

                   common_name xtra_goals_conceded_overall
18          Huddersfield Town                        19.76
16                     Fulham                         16.4
8              AFC Bournemouth                        13.76
7                  Southampton                         9.14
15                Cardiff City                         7.06
6                      Burnley                         4.54
13                     Watford                          2.0
17       Brighton & Hove Albion                       -0.04
9            Manchester United                        -0.72
5                      Everton                         -1.5
3               Leicester City                        -2.16
0                      Arsenal                        -3.72
12             West Ham United                         -3.9
19      Wolverhampton Wanderers                       -4.16
11                     Chelsea                        -4.32
4               Crystal Palace                        -5.14
14             Newcastle United                       -5.96
2              Manchester City                        -7.02
1            Tottenham Hotspur                        -12.3
10                   Liverpool                       -14.48

# We should compare those the graphs between the actual goals conceded
and xtra goals conded overall

# Here we can see the extremes of the dataset, HUddersfiled conceded
much more goals than expected- 25% of the goals that they conceded

# whle liverpool overperformed defensively and so did spurs

#Man utd and brighton's defence was as expected
```

We don't really need to group goals_conceded as that is the same as goals_scored

```
# So what's next, # correlation maybe

# So we know expected goals, expected goals against

# CNa we combine them?

overall_teams_2['g_conc_vs_xga']=overall_teams_2['g_conc_vs_xga'].asty
pe(str).astype(float)

# Okay let;s deice what correlations we want to dO?
```

```python
#First, we'll do correlation between

print(overall_teams_2['goals_conceded'].corr(overall_teams_2['league_p
osition']))
```

```
---------------------------------------------------------------------
-----
AttributeError                            Traceback (most recent call
last)
~\AppData\Local\Temp\ipykernel_3780\959703263.py in <module>
----> 1
print(overall_teams_2['goals_conceded'].corr(overall_teams_2['league_p
osition']))

~\anaconda3\lib\site-packages\pandas\core\series.py in corr(self,
other, method, min_periods)
   2555
   2556         if method in ["pearson", "spearman", "kendall"] or
callable(method):
-> 2557             return nanops.nancorr(
   2558                 this.values, other.values, method=method,
min_periods=min_periods
   2559             )

~\anaconda3\lib\site-packages\pandas\core\nanops.py in _f(*args,
**kwargs)
     91             try:
     92                 with np.errstate(invalid="ignore"):
---> 93                     return f(*args, **kwargs)
     94             except ValueError as e:
     95                 # we want to transform an object array

~\anaconda3\lib\site-packages\pandas\core\nanops.py in nancorr(a, b,
method, min_periods)
   1544
   1545     f = get_corr_func(method)
-> 1546     return f(a, b)
   1547
   1548

~\anaconda3\lib\site-packages\pandas\core\nanops.py in func(a, b)
   1565
   1566         def func(a, b):
-> 1567             return np.corrcoef(a, b)[0, 1]
   1568
   1569         return func

<__array_function__ internals> in corrcoef(*args, **kwargs)

~\anaconda3\lib\site-packages\numpy\lib\function_base.py in
```

```
corrcoef(x, y, rowvar, bias, ddof, dtype)
   2681        warnings.warn('bias and ddof have no effect and are
deprecated',
   2682                        DeprecationWarning, stacklevel=3)
-> 2683        c = cov(x, y, rowvar, dtype=dtype)
   2684        try:
   2685            d = diag(c)

<__array_function__ internals> in cov(*args, **kwargs)

~\anaconda3\lib\site-packages\numpy\lib\function_base.py in cov(m, y,
rowvar, bias, ddof, fweights, aweights, dtype)
   2516                w *= aweights
   2517
-> 2518        avg, w_sum = average(X, axis=1, weights=w, returned=True)
   2519        w_sum = w_sum[0]
   2520

<__array_function__ internals> in average(*args, **kwargs)

~\anaconda3\lib\site-packages\numpy\lib\function_base.py in average(a,
axis, weights, returned)
    413
    414        if returned:
--> 415            if scl.shape != avg.shape:
    416                scl = np.broadcast_to(scl, avg.shape).copy()
    417            return avg, scl

AttributeError: 'float' object has no attribute 'shape'
```

```
overall_teams_2['goals_conceded'].info()
```

```
<class 'pandas.core.series.Series'>
RangeIndex: 20 entries, 0 to 19
Series name: goals_conceded
Non-Null Count  Dtype
--------------  -----
20 non-null     object
dtypes: object(1)
memory usage: 288.0+ bytes
```

```
overall_teams_2['goals_conceded']=overall_teams_2['goals_conceded'].as
type(str).astype(int)
```

```
print(overall_teams_2['goals_conceded'].corr(overall_teams_2['league_p
osition']))
```

```
0.9064264639383145
```

```
#as expected a strong correlation
```

```
overall_teams_2['xg_against_avg_overall']=overall_teams_2['xg_against_
avg_overall'].astype(str).astype(float)

print(overall_teams_2['xg_against_avg_overall'].corr(overall_teams_2['
league_position']))

0.7985790732125752

# a weaker correlation between xg against and league_postion

# indication that xg is not an exact indicator to goals conceded

print(overall_teams_2['xg_against_avg_overall'].corr(overall_teams_2['
goals_conceded_per_match'].astype(float)))

0.8850763914729524

# But still a strong correlation between xg and goals conceded per
game, so still a very good indicator

## Breakdownn of grouping of overall goals conceded- we don;t have
that figure, we have to compare away and home side-by side and build
new field

testing_dist_goal_conc_ov=completed_stats.groupby('home_team_name',
as_index=False)['home_team_goal_count'].value_counts()
```

Break down of away and home goals conceded

Home

```
overall_teams_2['goals_conceded_home']

0      16
1      16
2      12
3      20
4      23
5      21
6      32
7      30
8      25
9      25
10     10
11     12
12     27
13     28
14     25
15     38
16     36
17     28
```

```
18      31
19      21
Name: goals_conceded_home, dtype: object

home_goals_conceded=overall_teams_2.groupby('common_name',as_index=Fal
se)['goals_conceded_home'].sum()

home_goals_conceded.sort_values(by='goals_conceded_home',ascending=Fal
se)
```

```
              common_name  goals_conceded_home
4             Cardiff City                   38
8                   Fulham                   36
3                  Burnley                   32
9         Huddersfield Town                  31
15               Southampton                 30
2     Brighton & Hove Albion                 28
17                  Watford                  28
18          West Ham United                  27
0           AFC Bournemouth                  25
13         Manchester United                 25
14          Newcastle United                 25
6             Crystal Palace                 23
19   Wolverhampton Wanderers                 21
7                   Everton                  21
10            Leicester City                 20
1                   Arsenal                  16
16         Tottenham Hotspur                 16
12           Manchester City                 12
5                   Chelsea                  12
11                Liverpool                  10
```

```
# so we see cardiff,fulham,burnleu huddersfiled conceded the most
goals at home

# liverpool,chelsea and city have the best defensive recrods
```

Look at xg conceded at home

```
overall_teams_2['goals_conceded_per_h_game']=overall_teams_2['goals_co
nceded_home']/19

overall_teams_2.loc[:,
['common_name','xg_against_avg_home','goals_conceded_per_h_game']].sor
t_values(by='goals_conceded_per_h_game')
```

```
              common_name xg_against_avg_home
goals_conceded_per_h_game
10                Liverpool                 0.85
0.526316
2           Manchester City                 0.75
```

```
                                        0.631579
11                 Chelsea                   1.03
0.631579
0                  Arsenal                   1.29
0.842105
1        Tottenham Hotspur                   1.28
0.842105
3            Leicester City                  1.19
1.052632
19  Wolverhampton Wanderers                  1.28
1.105263
5                  Everton                   1.19
1.105263
4            Crystal Palace                  1.31
1.210526
8            AFC Bournemouth                 1.47
1.315789
14          Newcastle United                1.21
1.315789
9          Manchester United                1.34
1.315789
12          West Ham United                 1.31
1.421053
13                 Watford                    1.4
1.473684
17    Brighton & Hove Albion                1.32
1.473684
7               Southampton                  1.36
1.578947
18         Huddersfield Town                1.34
1.631579
6                  Burnley                   1.46
1.684211
16                  Fulham                    1.6
1.894737
15             Cardiff City                  1.43
2.0
```

```python
overall_teams_2['g_conc_vs_xga_home']=overall_teams_2['goals_conceded_
per_h_game']-overall_teams_2['xg_against_avg_home']

overall_teams_2[['common_name','g_conc_vs_xga_home']].sort_values(by=[
'g_conc_vs_xga_home'],ascending=False)
```

```
              common_name g_conc_vs_xga_home
15             Cardiff City                0.57
16                  Fulham            0.294737
18         Huddersfield Town          0.291579
6                  Burnley            0.224211
7               Southampton          0.218947
```

```
17      Brighton & Hove Albion          0.153684
12             West Ham United          0.111053
14           Newcastle United          0.105789
13                    Watford          0.073684
9            Manchester United         -0.024211
5                     Everton         -0.084737
4              Crystal Palace         -0.099474
2              Manchester City        -0.118421
3               Leicester City        -0.137368
8              AFC Bournemouth        -0.154211
19  Wolverhampton Wanderers           -0.174737
10                  Liverpool         -0.323684
11                    Chelsea         -0.398421
1           Tottenham Hotspur         -0.437895
0                     Arsenal         -0.447895
```

#Cardiff conceded more goals at home they were supposed to, considerably

Arsenal conceded less goals at home, than they were supposed to- i.e they're defence appears better than it actually is

City is most impressivec,not only did they concede the lear but it was expected

clean_stats_from_matches=completed_stats

So similat to above with the value_counts, let's find out how many clean sheets, how mny times they concede,1,2,3,4,5,6 uniquely, have to put it as away

testing_dist_goal_conc_home=completed_stats.groupby('home_team_name', as_index=False)['away_team_goal_count'].value_counts()

testing_dist_goal_conc_home

```
            home_team_name  away_team_goal_count   count
0           AFC Bournemouth                    0       6
1           AFC Bournemouth                    2       5
2           AFC Bournemouth                    1       5
3           AFC Bournemouth                    3       2
4           AFC Bournemouth                    4       1
..                      ...                  ...     ...
85          West Ham United                    3       1
86  Wolverhampton Wanderers                    0       7
87  Wolverhampton Wanderers                    1       5
88  Wolverhampton Wanderers                    2       5
89  Wolverhampton Wanderers                    3       2

[90 rows x 3 columns]
```

```
#Let's see frequency of clean sheet

#let's test the setting with copy warning method

zero_goals_conceded_home=testing_dist_goal_conc_home.loc[testing_dist_
goal_conc_home['away_team_goal_count']==0].copy()

zero_goals_conceded_home.sort_values(by='count',ascending=False)
```

	home_team_name	away_team_goal_count	count
53	Liverpool	0	12
26	Chelsea	0	10
57	Manchester City	0	9
33	Everton	0	9
86	Wolverhampton Wanderers	0	7
6	Arsenal	0	7
73	Tottenham Hotspur	0	7
29	Crystal Palace	0	7
0	AFC Bournemouth	0	6
50	Leicester City	0	6
21	Cardiff City	0	5
15	Burnley	0	5
66	Newcastle United	0	5
11	Brighton & Hove Albion	0	5
78	Watford	0	4
83	West Ham United	0	4
47	Huddersfield Town	0	3
40	Fulham	0	3
71	Southampton	0	3
62	Manchester United	0	2

```
#Utd have the least, which is very poor, liverpool appear to have a a
very strong defensive record but we know they should have let in more
goals than they actually did

zero_goals_conceded_home['clean_sheet_perc']=zero_goals_conceded_home[
'count']/19 * 100

zero_goals_conceded_home.head()
```

	home_team_name	away_team_goal_count	count	clean_sheet_perc
0	AFC Bournemouth	0	6	31.578947
6	Arsenal	0	7	36.842105
11	Brighton & Hove Albion	0	5	26.315789
15	Burnley	0	5	26.315789

```
21              Cardiff City                              0         5
26.315789
```

```
zero_goals_conceded_home.rename(columns={'count':'no.of_clean_sheets'}
,inplace= True)
```

```
zero_goals_conceded_home.head()
```

```
            home_team_name  away_team_goal_count
no.of_clean_sheets  \
0          AFC Bournemouth                     0                    6

6                  Arsenal                     0                    7

11   Brighton & Hove Albion                    0                    5

15                 Burnley                     0                    5

21             Cardiff City                    0                    5


     clean_sheet_perc
0          31.578947
6          36.842105
11         26.315789
15         26.315789
21         26.315789
```

```
zero_goals_conceded_home
```

	home_team_name	away_team_goal_count	no.of_clean_sheets
0	AFC Bournemouth	0	6
6	Arsenal	0	7
11	Brighton & Hove Albion	0	5
15	Burnley	0	5
21	Cardiff City	0	5
26	Chelsea	0	10
29	Crystal Palace	0	7
33	Everton	0	9
40	Fulham	0	3
47	Huddersfield Town	0	3

50	Leicester City	0	6
53	Liverpool	0	12
57	Manchester City	0	9
62	Manchester United	0	2
66	Newcastle United	0	5
71	Southampton	0	3
73	Tottenham Hotspur	0	7
78	Watford	0	4
83	West Ham United	0	4
86	Wolverhampton Wanderers	0	7

```
    clean_sheet_perc
0         31.578947
6         36.842105
11        26.315789
15        26.315789
21        26.315789
26        52.631579
29        36.842105
33        47.368421
40        15.789474
47        15.789474
50        31.578947
53        63.157895
57        47.368421
62        10.526316
66        26.315789
71        15.789474
73        36.842105
78        21.052632
83        21.052632
86        36.842105
```

zero_goals_conceded_home.head()

	home_team_name	away_team_goal_count	no.of_clean_sheets \
0	AFC Bournemouth	0	6
6	Arsenal	0	7

	home_team_name	away_team_goal_count	count
11	Brighton & Hove Albion	0	5
15	Burnley	0	5
21	Cardiff City	0	5

	clean_sheet_perc
0	31.578947
6	36.842105
11	26.315789
15	26.315789
21	26.315789

```
less_than_three_goals_conceded_home=testing_dist_goal_conc_home.loc[(testing_dist_goal_conc_home['away_team_goal_count']>0) & (testing_dist_goal_conc_home['away_team_goal_count']<3)].copy()

less_than_three_goals_conceded_home
```

	home_team_name	away_team_goal_count	count
1	AFC Bournemouth	2	5
2	AFC Bournemouth	1	5
5	Arsenal	1	9
7	Arsenal	2	2
9	Brighton & Hove Albion	1	6
10	Brighton & Hove Albion	2	5
16	Burnley	1	5
18	Burnley	2	3
23	Cardiff City	2	4
25	Cardiff City	1	3
27	Chelsea	1	6
28	Chelsea	2	3
30	Crystal Palace	2	5
31	Crystal Palace	1	4
34	Everton	1	5
35	Everton	2	2
38	Fulham	2	9
39	Fulham	1	3
44	Huddersfield Town	2	6
45	Huddersfield Town	1	6
49	Leicester City	1	8
51	Leicester City	2	4
54	Liverpool	1	5
55	Liverpool	2	1
58	Manchester City	1	9
60	Manchester United	1	10
61	Manchester United	2	6
64	Newcastle United	2	7
65	Newcastle United	1	5

```
68              Southampton                          1        6
69              Southampton                          2        6
72          Tottenham Hotspur                        1        9
74          Tottenham Hotspur                        2        2
76                  Watford                          1        7
77                  Watford                          2        5
81          West Ham United                          2        7
82          West Ham United                          1        6
87  Wolverhampton Wanderers                          1        5
88  Wolverhampton Wanderers                          2        5
```

```
dist_home_goal_conc_count=
testing_dist_goal_conc_home.groupby('away_team_goal_count')
['count'].sum()
```

```
dist_home_goal_conc_count
```

```
away_team_goal_count
0    119
1    122
2     87
3     36
4      9
5      6
6      1
Name: count, dtype: int64
```

```
# I guess it would be the same as the away goal scoring, is equivalent
to the home_goaL_conceding account and vice versa

# this the away_goal_conc_coubnt is equivalent to the hometeam scoring

# i.e we have those graphs matplot lib to check that
```

We want to find expected win/loss and also how many points they should've ended up with

We can use this as an example to test how the xg model was wrong

This is where we start the xg properly – things before this should be gotten rid of

```
overall_teams_2[['common_name','points_per_game_home','points_per_game
_away']]
```

```
              common_name points_per_game_home points_per_game_away
0                 Arsenal                 2.37                 1.32
1       Tottenham Hotspur                  2.0                 1.74
2         Manchester City                 2.84                 2.32
3          Leicester City                 1.42                 1.32
4           Crystal Palace                1.05                 1.53
5                 Everton                 1.79                 1.05
6                 Burnley                 1.21                 0.89
7              Southampton                1.21                 0.84
8          AFC Bournemouth               1.53                 0.84
9       Manchester United                 1.89                 1.58
10              Liverpool                 2.79                 2.32
11                Chelsea                 2.21                 1.58
12        West Ham United                 1.63                 1.11
13                Watford                 1.42                 1.21
14       Newcastle United                 1.32                 1.05
15            Cardiff City                1.05                 0.74
16                 Fulham                 1.11                 0.26
17   Brighton & Hove Albion               1.21                 0.68
18        Huddersfield Town              0.47                 0.37
19  Wolverhampton Wanderers               1.79                 1.21
```

```
overall_teams_2
```

	common_name	wins	wins_home	wins_away	draws	draws_home
0	Arsenal	21	14	7	7	3
1	Tottenham Hotspur	23	12	11	2	2
2	Manchester City	32	18	14	2	0

3	Leicester City	15	8	7	7	3
4	Crystal Palace	14	5	9	7	5
5	Everton	15	10	5	9	4
6	Burnley	11	7	4	7	2
7	Southampton	9	5	4	12	8
8	AFC Bournemouth	13	8	5	6	5
9	Manchester United	19	10	9	9	6
10	Liverpool	30	17	13	7	2
11	Chelsea	21	12	9	9	6
12	West Ham United	15	9	6	7	4
13	Watford	14	8	6	8	3
14	Newcastle United	12	8	4	9	1
15	Cardiff City	10	6	4	4	2
16	Fulham	7	6	1	5	3
17	Brighton & Hove Albion	9	6	3	9	5
18	Huddersfield Town	3	2	1	7	3
19	Wolverhampton Wanderers	16	10	6	9	4

	draws_away	losses	losses_home	losses_away	...	clean_sheets_2h_home
0	4	10	2	8	...	12
1	0	13	5	8	...	11
2	2	4	1	3	...	14
3	4	16	8	8	...	13
4	2	17	9	8	...	8
5	5	14	5	9	...	12
6	5	20	10	10	...	8
7	4	17	6	11	...	8

8	1	19	6	13	...	8
9	3	10	3	7	...	5
10	5	1	0	1	...	15
11	3	8	1	7	...	14
12	3	16	6	10	...	8
13	5	16	8	8	...	9
14	8	17	10	7	...	10
15	2	24	11	13	...	9
16	2	26	10	16	...	9
17	4	20	8	12	...	8
18	4	28	14	14	...	9
19	5	13	5	8	...	10

```
    clean_sheets_2h_away  failed_to_score_2h_home  failed_to_score_2h_away  \
0                      6                        4
10
1                      9                        5
9
2                     13                        2
7
3                      9                        9
4
4                      6                       10
5
5                      7                       10
10
6                      8                       11
11
7                      7                        9
10
8                      6                       10
12
9                      9                        8
10
10                    12                        4
5
11                    10                        5
```

```
11
12                     7                    9
12
13                     8                    5
10
14                     9                   10
15
15                     8                    8
11
16                     5                   10
13
17                     7                   11
12
18                     6                   13
14
19                     8                    7
9
```

	diff_g_vs_xg	extra_goals_scored	g_conc_vs_xga	\\
0	0.491053	18.66	-0.097895	
1	0.183158	6.96	-0.323684	
2	0.32	12.16	-0.184737	
3	-0.217895	-8.28	-0.056842	
4	-0.017895	-0.68	-0.135263	
5	-0.028947	-1.1	-0.039474	
6	0.284211	10.8	0.119474	
7	-0.165789	-6.3	0.240526	
8	0.123684	4.7	0.362105	
9	0.080526	3.06	-0.018947	
10	0.442105	16.8	-0.381053	
11	-0.052105	-1.98	-0.113684	
12	0.018421	0.7	-0.102632	
13	0.208421	7.92	0.052632	
14	0.005263	0.2	-0.156842	
15	-0.175263	-6.66	0.185789	
16	-0.245263	-9.32	0.431579	
17	0.021053	0.8	-0.001053	
18	-0.361053	-13.72	0.520000	
19	-0.123158	-4.68	-0.109474	

	xtra_goals_conceded_overall	goals_conceded_per_h_game	g_conc_vs_xga_home
0	-3.72	0.842105	-0.447895
1	-12.3	0.842105	-0.437895
2	-7.02	0.631579	-0.118421
3	-2.16	1.052632	-

			0.137368
4	-5.14	1.210526	-0.099474
5	-1.5	1.105263	-0.084737
6	4.54	1.684211	0.224211
7	9.14	1.578947	0.218947
8	13.76	1.315789	-0.154211
9	-0.72	1.315789	-0.024211
10	-14.48	0.526316	-0.323684
11	-4.32	0.631579	-0.398421
12	-3.9	1.421053	0.111053
13	2.0	1.473684	0.073684
14	-5.96	1.315789	0.105789
15	7.06	2.0	0.57
16	16.4	1.894737	0.294737
17	-0.04	1.473684	0.153684
18	19.76	1.631579	0.291579
19	-4.16	1.105263	-0.174737

[20 rows x 254 columns]

```
# the data already has points per game away and home,to be fair we did
do wins,home wins, awya, draws,draws_away ish
```

e merge the overall_teams and stats_from_matches, to get xw & xl in the eyes of the home_team

```
overall_teams_2
```

	common_name	wins	wins_home	wins_away	draws	draws_home
0	Arsenal	21	14	7	7	3
1	Tottenham Hotspur	23	12	11	2	2
2	Manchester City	32	18	14	2	0
3	Leicester City	15	8	7	7	3
4	Crystal Palace	14	5	9	7	5
5	Everton	15	10	5	9	4
6	Burnley	11	7	4	7	2
7	Southampton	9	5	4	12	8
8	AFC Bournemouth	13	8	5	6	5
9	Manchester United	19	10	9	9	6
10	Liverpool	30	17	13	7	2
11	Chelsea	21	12	9	9	6
12	West Ham United	15	9	6	7	4
13	Watford	14	8	6	8	3
14	Newcastle United	12	8	4	9	1
15	Cardiff City	10	6	4	4	2
16	Fulham	7	6	1	5	3
17	Brighton & Hove Albion	9	6	3	9	5
18	Huddersfield Town	3	2	1	7	3
19	Wolverhampton Wanderers	16	10	6	9	4

	draws_away	losses	losses_home	losses_away	...	clean_sheets_2h_home
0	4	10	2	8	...	12
1	0	13	5	8	...	11
2	2	4	1	3	...	14
3	4	16	8	8	...	13

4	2	17	9	8	...	8
5	5	14	5	9	...	12
6	5	20	10	10	...	8
7	4	17	6	11	...	8
8	1	19	6	13	...	8
9	3	10	3	7	...	5
10	5	1	0	1	...	15
11	3	8	1	7	...	14
12	3	16	6	10	...	8
13	5	16	8	8	...	9
14	8	17	10	7	...	10
15	2	24	11	13	...	9
16	2	26	10	16	...	9
17	4	20	8	12	...	8
18	4	28	14	14	...	9
19	5	13	5	8	...	10

	clean_sheets_2h_away	failed_to_score_2h_home	failed_to_score_2h_away \
0	6	4	10
1	9	5	9
2	13	2	7
3	9	9	4
4	6	10	5
5	7	10	10
6	8	11	11
7	7	9	

```
10
8                              6                            10
12
9                              9                             8
10
10                            12                             4
5
11                            10                             5
11
12                             7                             9
12
13                             8                             5
10
14                             9                            10
15
15                             8                             8
11
16                             5                            10
13
17                             7                            11
12
18                             6                            13
14
19                             8                             7
9
```

```
    diff_g_vs_xg  extra_goals_scored  g_conc_vs_xga  \
0       0.491053               18.66      -0.097895
1       0.183158                6.96      -0.323684
2       0.32                   12.16      -0.184737
3      -0.217895               -8.28      -0.056842
4      -0.017895               -0.68      -0.135263
5      -0.028947               -1.1       -0.039474
6       0.284211               10.8        0.119474
7      -0.165789               -6.3        0.240526
8       0.123684                4.7        0.362105
9       0.080526                3.06      -0.018947
10      0.442105               16.8       -0.381053
11     -0.052105               -1.98      -0.113684
12      0.018421                0.7       -0.102632
13      0.208421                7.92       0.052632
14      0.005263                0.2       -0.156842
15     -0.175263               -6.66       0.185789
16     -0.245263               -9.32       0.431579
17      0.021053                0.8       -0.001053
18     -0.361053              -13.72       0.520000
19     -0.123158               -4.68      -0.109474

    xtra_goals_conceded_overall  goals_conceded_per_h_game
g_conc_vs_xga_home
```

0	-3.72	0.842105	-0.447895
1	-12.3	0.842105	-0.437895
2	-7.02	0.631579	-0.118421
3	-2.16	1.052632	-0.137368
4	-5.14	1.210526	-0.099474
5	-1.5	1.105263	-0.084737
6	4.54	1.684211	0.224211
7	9.14	1.578947	0.218947
8	13.76	1.315789	-0.154211
9	-0.72	1.315789	-0.024211
10	-14.48	0.526316	-0.323684
11	-4.32	0.631579	-0.398421
12	-3.9	1.421053	0.111053
13	2.0	1.473684	0.073684
14	-5.96	1.315789	0.105789
15	7.06	2.0	0.57
16	16.4	1.894737	0.294737
17	-0.04	1.473684	0.153684
18	19.76	1.631579	0.291579
19	-4.16	1.105263	-0.174737

[20 rows x 254 columns]

```python
overall_teams_merge=overall_teams_2[['common_name','xg_against_avg_home','xg_for_avg_home']].copy()

overall_teams_merge['home_team_name']=overall_teams_2['common_name']

overall_teams_merge.drop(columns='common_name',inplace=True)
```

```python
test_merge=pd.merge(overall_teams_merge,stats_from_matches,how='outer'
,on='home_team_name')

# We are going to test this model on Arsenal

arsenal=test_merge[['Game
Week','home_team_name','home_team_goal_count','xg_for_avg_home','team_
a_xg','team_b_xg','away_team_goal_count']].head(19)

# don;t need this cell below

arsenal[['home_team_name','team_a_xg','team_b_xg']]
```

```
    home_team_name  team_a_xg  team_b_xg
0          Arsenal       1.11       1.91
1          Arsenal       2.01       1.44
2          Arsenal       1.30       1.46
3          Arsenal       1.22       1.53
4          Arsenal       2.13       0.89
5          Arsenal       1.48       1.29
6          Arsenal       1.39       1.40
7          Arsenal       2.15       1.40
8          Arsenal       1.30       0.66
9          Arsenal       1.31       0.93
10         Arsenal       2.14       1.17
11         Arsenal       1.64       1.52
12         Arsenal       1.50       1.29
13         Arsenal       1.40       1.28
14         Arsenal       1.82       1.25
15         Arsenal       1.36       1.46
16         Arsenal       1.15       0.58
17         Arsenal       1.83       1.67
18         Arsenal       2.25       1.34
```

```python
# could find correlation to home_team xg and actual goals

arsenal['team_a_xg'].mean()
```

```
1.6047368421052632
```

```python
arsenal['xg_goal_difference']=arsenal['team_a_xg']-
arsenal['team_b_xg']

arsenal['result']= arsenal['home_team_goal_count']-
arsenal['away_team_goal_count']

arsenal
```

```
    Game Week home_team_name  home_team_goal_count  xg_for_avg_home
team_a_xg  \
0           1        Arsenal                     0              1.6
1.11
```

```
1         3         Arsenal               3              1.6
2.01
2         6         Arsenal               2              1.6
1.30
3         7         Arsenal               2              1.6
1.22
4         9         Arsenal               3              1.6
2.13
5         11        Arsenal               1              1.6
1.48
6         12        Arsenal               1              1.6
1.39
7         14        Arsenal               4              1.6
2.15
8         16        Arsenal               1              1.6
1.30
9         18        Arsenal               3              1.6
1.31
10        21        Arsenal               4              1.6
2.14
11        23        Arsenal               2              1.6
1.64
12        24        Arsenal               2              1.6
1.50
13        27        Arsenal               2              1.6
1.40
14        28        Arsenal               5              1.6
1.82
15        30        Arsenal               2              1.6
1.36
16        32        Arsenal               2              1.6
1.15
17        35        Arsenal               2              1.6
1.83
18        37        Arsenal               1              1.6
2.25
```

	team_b_xg	away_team_goal_count	xg_goal_difference	result
0	1.91	2	-0.80	-2
1	1.44	1	0.57	2
2	1.46	0	-0.16	2
3	1.53	0	-0.31	2
4	0.89	1	1.24	2
5	1.29	1	0.19	0
6	1.40	1	-0.01	0
7	1.40	2	0.75	2
8	0.66	0	0.64	1
9	0.93	1	0.38	2
10	1.17	1	0.97	3

```
11      1.52                    0               0.12      2
12      1.29                    1               0.21      1
13      1.28                    0               0.12      2
14      1.25                    1               0.57      4
15      1.46                    0              -0.10      2
16      0.58                    0               0.57      2
17      1.67                    3               0.16     -1
18      1.34                    1               0.91      0
```

```python
def arsenal_result(x):
    if x>0:
        return 3
    if x ==0:
        return 1
    if x<0:
        return 0
```

```python
arsenal['points_at_home']=arsenal['result'].apply(lambda
x:arsenal_result(x))

arsenal.head()
```

```
   Game Week home_team_name  home_team_goal_count xg_for_avg_home
team_a_xg  \
0          1         Arsenal                     0             1.6
1.11
1          3         Arsenal                     3             1.6
2.01
2          6         Arsenal                     2             1.6
1.30
3          7         Arsenal                     2             1.6
1.22
4          9         Arsenal                     3             1.6
2.13

   team_b_xg  away_team_goal_count  xg_goal_difference  result
points_at_home
0       1.91                     2               -0.80      -2
0
1       1.44                     1                0.57       2
3
2       1.46                     0               -0.16       2
3
3       1.53                     0               -0.31       2
3
4       0.89                     1                1.24       2
3
```

```python
arsenal['points_at_home'].sum()
```

```
45
```

```python
# we know from checking the home_league_table that this is correct

def arsenal_xg_Results(x):
    if x>0.5:
        return 3
    if x>-0.5 and x<0.5:
        return 1
    if x<0.5:
        return 0

arsenal['x_points']=arsenal['xg_goal_difference'].apply(lambda
x:arsenal_xg_Results(x))

arsenal.head()
```

```
   Game Week home_team_name  home_team_goal_count xg_for_avg_home
team_a_xg  \
0         1         Arsenal                     0             1.6
1.11
1         3         Arsenal                     3             1.6
2.01
2         6         Arsenal                     2             1.6
1.30
3         7         Arsenal                     2             1.6
1.22
4         9         Arsenal                     3             1.6
2.13

   team_b_xg  away_team_goal_count  xg_goal_difference  result  \
0       1.91                     2               -0.80      -2
1       1.44                     1                0.57       2
2       1.46                     0               -0.16       2
3       1.53                     0               -0.31       2
4       0.89                     1                1.24       2

   points_at_home  x_points
0               0         0
1               3         3
2               3         1
3               3         1
4               3         3
```

```python
arsenal[['points_at_home','x_points']].sum()
```

```
points_at_home    45
x_points          34
dtype: int64
```

```python
# So that's where xg is betwen -0.5 and 0.5

arsenal['xg_goal_difference'].corr(arsenal['result'])
```
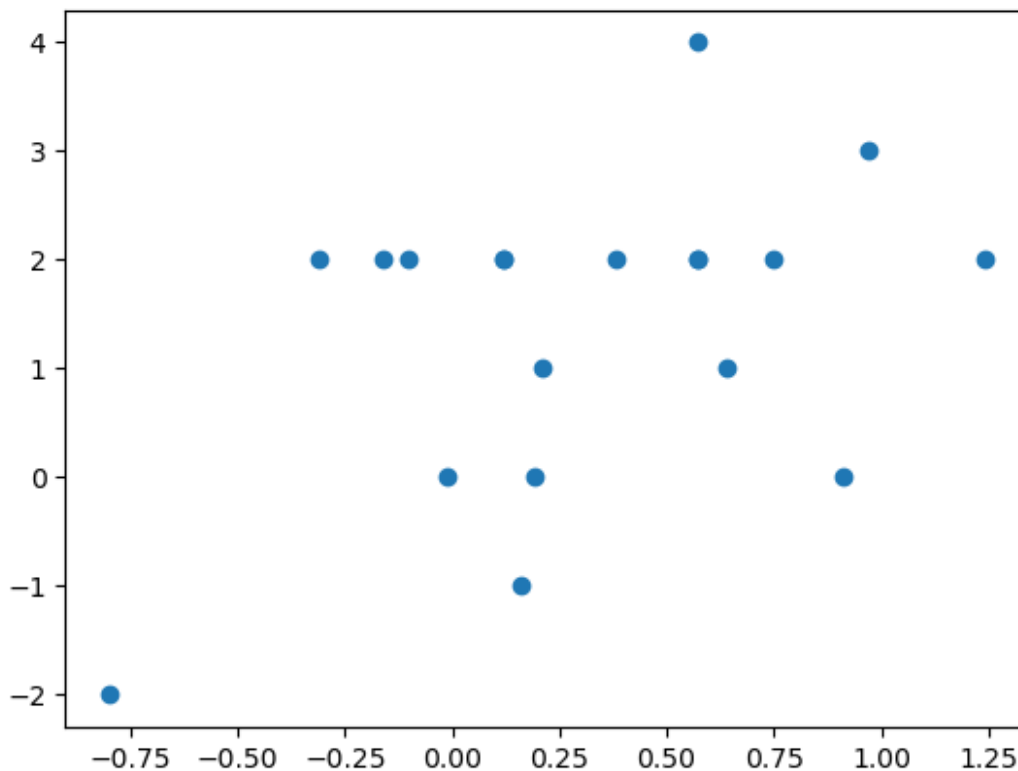
```
0.4426965554165693
```

```
# But we want to  see the impact across the whole leagie, this sample
may be biased
```

```python
import matplotlib.pyplot as plt

x=arsenal['xg_goal_difference']
y=arsenal['result']

plt.scatter(x,y)
plt.show()
```



```python
# The correlation between xg_gaol_difference and results

# we have xg_results, we need to compare to points per game home

#is 0.5 the best way?

#I'd say do that end

all_teams_home=test_merge[['Game
Week','home_team_name','home_team_goal_count','xg_for_avg_home','team_
a_xg','team_b_xg','away_team_goal_count']].copy()

all_teams_home.head()
```

```
    Game Week home_team_name  home_team_goal_count xg_for_avg_home
team_a_xg  \
0           1        Arsenal                     0            1.6
1.11
1           3        Arsenal                     3            1.6
2.01
2           6        Arsenal                     2            1.6
1.30
3           7        Arsenal                     2            1.6
1.22
4           9        Arsenal                     3            1.6
2.13

   team_b_xg  away_team_goal_count
0       1.91                     2
1       1.44                     1
2       1.46                     0
3       1.53                     0
4       0.89                     1
```

```python
all_teams_home['home_result']=all_teams_home['home_team_goal_count']-
all_teams_home['away_team_goal_count']

all_teams_home['x_goal_diff']=all_teams_home['team_a_xg']-
all_teams_home['team_b_xg']

all_teams_home.head()
```

```
    Game Week home_team_name  home_team_goal_count xg_for_avg_home
team_a_xg  \
0           1        Arsenal                     0            1.6
1.11
1           3        Arsenal                     3            1.6
2.01
2           6        Arsenal                     2            1.6
1.30
3           7        Arsenal                     2            1.6
1.22
4           9        Arsenal                     3            1.6
2.13

   team_b_xg  away_team_goal_count  home_result  x_goal_diff
0       1.91                     2           -2        -0.80
1       1.44                     1            2         0.57
2       1.46                     0            2        -0.16
3       1.53                     0            2        -0.31
4       0.89                     1            2         1.24
```

```python
# now we want to find home_points
```

```
def home_points(x):
    if x>0:
        return 3
    if x==0:
        return 1
    if x<0:
        return 0

all_teams_home['home_points']=all_teams_home['home_result'].apply(lamb
da x:home_points(x))

def home_xg_Results(x):
    if x>0.5:
        return 3
    if x>-0.5 and x<0.5:
        return 1
    if x<0.5:
        return 0

all_teams_home['xg_home_points']=all_teams_home['x_goal_diff'].apply(l
ambda x:home_xg_Results(x))

all_teams_home.head()
```

```
   Game Week home_team_name  home_team_goal_count xg_for_avg_home
team_a_xg  \
0          1        Arsenal                     0             1.6
1.11
1          3        Arsenal                     3             1.6
2.01
2          6        Arsenal                     2             1.6
1.30
3          7        Arsenal                     2             1.6
1.22
4          9        Arsenal                     3             1.6
2.13

   team_b_xg  away_team_goal_count  home_result  x_goal_diff
home_points  \
0       1.91                     2           -2        -0.80
0
1       1.44                     1            2         0.57
3
2       1.46                     0            2        -0.16
3
3       1.53                     0            2        -0.31
3
4       0.89                     1            2         1.24
3

   xg_home_points
```

```
0          0.0
1          3.0
2          1.0
3          1.0
4          3.0
```

```
all_teams_home['x_goal_diff'].corr(all_teams_home['xg_home_points'])
```

```
0.8769749749094116
```

When we comparre the x_goal_diff to xg_home_points, there is a strong correlation- thus it appears that is model is fair

```
checking_xg_model=all_teams_home.groupby('home_team_name',as_index=False)[['xg_home_points','home_points']].sum()
```

```
checking_xg_model.head()
```

	home_team_name	xg_home_points	home_points
0	AFC Bournemouth	22.0	29
1	Arsenal	34.0	45
2	Brighton & Hove Albion	21.0	23
3	Burnley	21.0	23
4	Cardiff City	23.0	20

```
overall_teams_2[['common_name','points_per_game_home']]
```

	common_name	points_per_game_home
0	Arsenal	2.37
1	Tottenham Hotspur	2.0
2	Manchester City	2.84
3	Leicester City	1.42
4	Crystal Palace	1.05
5	Everton	1.79
6	Burnley	1.21
7	Southampton	1.21
8	AFC Bournemouth	1.53
9	Manchester United	1.89
10	Liverpool	2.79
11	Chelsea	2.21
12	West Ham United	1.63
13	Watford	1.42
14	Newcastle United	1.32
15	Cardiff City	1.05
16	Fulham	1.11

```
17     Brighton & Hove Albion                    1.21
18          Huddersfield Town                    0.47
19  Wolverhampton Wanderers                      1.79

checking_xg_model

              home_team_name  xg_home_points  home_points
0           AFC Bournemouth            22.0           29
1                   Arsenal            34.0           45
2     Brighton & Hove Albion          21.0           23
3                   Burnley            21.0           23
4              Cardiff City            23.0           20
5                   Chelsea            49.0           42
6            Crystal Palace            33.0           20
7                   Everton            32.0           34
8                    Fulham            21.0           21
9          Huddersfield Town          23.0            9
10            Leicester City          33.0           27
11                Liverpool            51.0           53
12           Manchester City          55.0           54
13         Manchester United          36.0           36
14           Newcastle United         25.0           25
15               Southampton          22.0           23
16          Tottenham Hotspur         35.0           38
17                   Watford          19.0           27
18           West Ham United          28.0           31
19   Wolverhampton Wanderers          25.0           34
```

after ehcking league_table, we know that the home_table is correct

```python
xx=checking_xg_model['xg_home_points']
yy=checking_xg_model['home_points']

plt.scatter(xx,yy)

plt.xlabel('exp_home_points')
plt.ylabel('home_points')

plt.show()
```
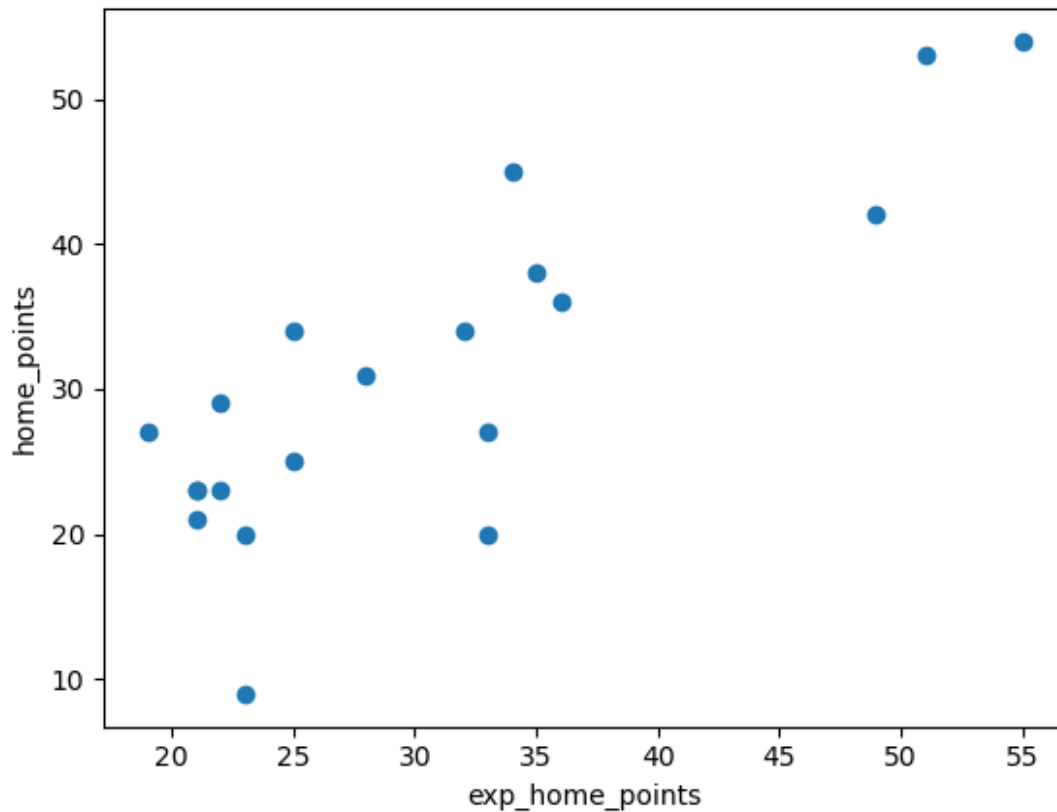
```
print(xx.corr(yy))
```

```
0.8294356683055515
```

#A strong correlation between expected points. so we will use this model

 # nnow we create a home_league_table_exp

```python
league_position_home=overall_teams_2[['common_name','league_position_home']].sort_values(by='common_name',ascending= True)
```

```
league_position_home
```

```
              common_name  league_position_home
8          AFC Bournemouth                    10
0                  Arsenal                     3
17   Brighton & Hove Albion                   16
6                  Burnley                    15
15            Cardiff City                    19
11                 Chelsea                     4
4            Crystal Palace                   18
5                  Everton                     7
16                  Fulham                    17
18        Huddersfield Town                    20
```

```
3             Leicester City                    11
10                 Liverpool                     2
2            Manchester City                     1
9          Manchester United                     6
14         Newcastle United                     13
7                Southampton                    14
1          Tottenham Hotspur                     5
13                   Watford                    12
12           West Ham United                     9
19  Wolverhampton Wanderers                     8

checking_xg_model.head()

            home_team_name  xg_home_points  home_points
0          AFC Bournemouth            22.0           29
1                  Arsenal            34.0           45
2    Brighton & Hove Albion           21.0           23
3                  Burnley            21.0           23
4              Cardiff City           23.0           20
```

We want to get the rank of the x_home_table against the actual home_table

```
checking_xg_model['exp_home_position']=checking_xg_model['xg_home_poin
ts'].rank( method='dense',ascending=False)

checking_xg_model.sort_values(by='exp_home_position',ascending=False)

            home_team_name  xg_home_points  home_points
exp_home_position
17                   Watford            19.0           27
14.0
2    Brighton & Hove Albion           21.0           23
13.0
3                  Burnley            21.0           23
13.0
8                    Fulham            21.0           21
13.0
0          AFC Bournemouth            22.0           29
12.0
15               Southampton           22.0           23
12.0
9          Huddersfield Town           23.0            9
11.0
4              Cardiff City            23.0           20
11.0
14         Newcastle United            25.0           25
```

```
10.0
19  Wolverhampton Wanderers                  25.0                34
10.0
18              West Ham United              28.0                31
9.0
7                       Everton              32.0                34
8.0
6                Crystal Palace              33.0                20
7.0
10               Leicester City              33.0                27
7.0
1                       Arsenal              34.0                45
6.0
16            Tottenham Hotspur              35.0                38
5.0
13            Manchester United              36.0                36
4.0
5                       Chelsea              49.0                42
3.0
11                    Liverpool              51.0                53
2.0
12              Manchester City              55.0                54
1.0
```

```
checking_xg_model['exp_league_position']=checking_xg_model.groupby('ho
me_team_name',as_index=False)['xg_home_points'].rank(ascending=False)

checking_xg_model.head()
```

```
            home_team_name  xg_home_points  home_points
exp_home_position  \
0          AFC Bournemouth             22.0                29
12.0
1                  Arsenal             34.0                45
6.0
2      Brighton & Hove Albion         21.0                23
13.0
3                  Burnley             21.0                23
13.0
4              Cardiff City             23.0                20
11.0

   exp_league_position
0                  1.0
1                  1.0
2                  1.0
3                  1.0
4                  1.0
```

```
checking_xg_model.drop(columns=['exp_home_position','exp_league_positi
on'],inplace=True)

checking_xg_model.head()
```

	home_team_name	xg_home_points	home_points
0	AFC Bournemouth	22.0	29
1	Arsenal	34.0	45
2	Brighton & Hove Albion	21.0	23
3	Burnley	21.0	23
4	Cardiff City	23.0	20

```
all_teams_home.head()
```

	Game Week	home_team_name	home_team_goal_count	xg_for_avg_home	team_a_xg
0	1	Arsenal	0	1.6	1.11
1	3	Arsenal	3	1.6	2.01
2	6	Arsenal	2	1.6	1.30
3	7	Arsenal	2	1.6	1.22
4	9	Arsenal	3	1.6	2.13

	team_b_xg	away_team_goal_count	home_result	x_goal_diff	home_points
0	1.91	2	-2	-0.80	0
1	1.44	1	2	0.57	3
2	1.46	0	2	-0.16	3
3	1.53	0	2	-0.31	3
4	0.89	1	2	1.24	3

	xg_home_points
0	0.0
1	3.0
2	1.0
3	1.0
4	3.0

We need to use goal difference to separate rankings

```python
x_goal_diff=all_teams_home.groupby('home_team_name',as_index=False)
['x_goal_diff'].sum()

x_goal_diff.head()
```

```
        home_team_name  x_goal_diff
0       AFC Bournemouth        -2.46
1               Arsenal         6.02
2   Brighton & Hove Albion      -1.91
3               Burnley        -3.82
4           Cardiff City        -2.89
```

```python
checking_xg_model.head()
```

```
        home_team_name  xg_home_points  home_points
0       AFC Bournemouth           22.0           29
1               Arsenal           34.0           45
2   Brighton & Hove Albion         21.0           23
3               Burnley           21.0           23
4           Cardiff City          23.0           20
```

```python
# We don't want the home_team_name as index


index=list(range(0,20))

index_2=pd.Index(index)

checking_xg_model=checking_xg_model.set_index(index_2)

checking_xg_model.head()
```

```
        home_team_name  xg_home_points  home_points
0       AFC Bournemouth           22.0           29
1               Arsenal           34.0           45
2   Brighton & Hove Albion         21.0           23
3               Burnley           21.0           23
4           Cardiff City          23.0           20
```

```python
x_goal_diff.head()
```

```
        home_team_name  x_goal_diff
0       AFC Bournemouth        -2.46
1               Arsenal         6.02
2   Brighton & Hove Albion      -1.91
3               Burnley        -3.82
4           Cardiff City        -2.89
```

```
new_leag_tab_h=x_goal_diff.merge(checking_xg_model,
on='home_team_name',how='inner')

new_leag_tab_h.head()
```

	home_team_name	x_goal_diff	xg_home_points	home_points
0	AFC Bournemouth	-2.46	22.0	29
1	Arsenal	6.02	34.0	45
2	Brighton & Hove Albion	-1.91	21.0	23
3	Burnley	-3.82	21.0	23
4	Cardiff City	-2.89	23.0	20

```
new_leag_tab_h['exp_rank']=new_leag_tab_h[['x_goal_diff','xg_home_poin
ts']].apply(tuple,axis=1).rank(method='dense',ascending=False).astype(
int)

new_leag_tab_h=new_leag_tab_h.sort_values("exp_rank")

new_leag_tab_h
```

	home_team_name	x_goal_diff	xg_home_points	home_points
12	Manchester City	28.20	55.0	54
11	Liverpool	20.58	51.0	53
5	Chelsea	14.79	49.0	42
10	Leicester City	8.88	33.0	27
16	Tottenham Hotspur	7.81	35.0	38
13	Manchester United	7.33	36.0	36
7	Everton	6.81	32.0	34
1	Arsenal	6.02	34.0	45
14	Newcastle United	3.90	25.0	25
6	Crystal Palace	3.66	33.0	20
19	Wolverhampton Wanderers	3.48	25.0	34
18	West Ham United	3.08	28.0	31
15	Southampton	1.51	22.0	23
17	Watford	-1.33	19.0	27
2	Brighton & Hove Albion	-1.91	21.0	23

9	Huddersfield Town	-2.06	23.0	9
0	AFC Bournemouth	-2.46	22.0	29
8	Fulham	-2.57	21.0	21
4	Cardiff City	-2.89	23.0	20
3	Burnley	-3.82	21.0	23

```
    exp_rank
12         1
11         2
5          3
10         4
16         5
13         6
7          7
1          8
14         9
6         10
19        11
18        12
15        13
17        14
2         15
9         16
0         17
8         18
4         19
3         20
```

```
# now we compare to the actual league_Table
```

we want to merge on the home_Team_name with the league _position table

```
league_position_home=overall_teams_2[['common_name','league_position_h
ome']].sort_values( by='league_position_home', ascending=True)

league_position_home
```

```
              common_name  league_position_home
2          Manchester City                     1
10                Liverpool                    2
0                  Arsenal                     3
11                 Chelsea                     4
```

```
1          Tottenham Hotspur              5
9          Manchester United             6
5               Everton                  7
19  Wolverhampton Wanderers             8
12         West Ham United               9
8          AFC Bournemouth              10
3          Leicester City               11
13              Watford                 12
14         Newcastle United             13
7            Southampton                14
6              Burnley                  15
17    Brighton & Hove Albion            16
16              Fulham                  17
4          Crystal Palace               18
15           Cardiff City               19
18        Huddersfield Town             20
```

```python
league_position_home['home_team_name']=league_position_home['common_name']

ov_new_league_tab_h=pd.merge(league_position_home,new_leag_tab_h,on='home_team_name',how='inner')

ov_new_league_tab_h
```

```
                common_name  league_position_home
home_team_name  \
0          Manchester City                      1          Manchester
City
1               Liverpool                       2
Liverpool
2                Arsenal                        3
Arsenal
3                Chelsea                        4
Chelsea
4          Tottenham Hotspur                    5          Tottenham
Hotspur
5          Manchester United                    6          Manchester
United
6               Everton                         7
Everton
7    Wolverhampton Wanderers                    8  Wolverhampton
Wanderers
8          West Ham United                      9          West Ham
United
9          AFC Bournemouth                     10          AFC
Bournemouth
10         Leicester City                      11          Leicester
City
11              Watford                        12
```

```
            Watford
12              Newcastle United                13              Newcastle
United
13                Southampton                   14
Southampton
14                  Burnley                     15
Burnley
15     Brighton & Hove Albion                   16     Brighton & Hove
Albion
16                   Fulham                     17
Fulham
17               Crystal Palace                 18                Crystal
Palace
18                Cardiff City                  19                 Cardiff
City
19              Huddersfield Town               20             Huddersfield
Town

    x_goal_diff  xg_home_points  home_points  exp_rank
0         28.20            55.0           54         1
1         20.58            51.0           53         2
2          6.02            34.0           45         8
3         14.79            49.0           42         3
4          7.81            35.0           38         5
5          7.33            36.0           36         6
6          6.81            32.0           34         7
7          3.48            25.0           34        11
8          3.08            28.0           31        12
9         -2.46            22.0           29        17
10         8.88            33.0           27         4
11        -1.33            19.0           27        14
12         3.90            25.0           25         9
13         1.51            22.0           23        13
14        -3.82            21.0           23        20
15        -1.91            21.0           23        15
16        -2.57            21.0           21        18
17         3.66            33.0           20        10
18        -2.89            23.0           20        19
19        -2.06            23.0            9        16
```

ov_new_league_tab_h[['common_name','league_position_home','exp_rank']]

```
           common_name  league_position_home  exp_rank
0       Manchester City                     1         1
1             Liverpool                     2         2
2               Arsenal                     3         8
3               Chelsea                     4         3
4     Tottenham Hotspur                     5         5
5     Manchester United                     6         6
6               Everton                     7         7
```

```
7     Wolverhampton Wanderers                           8       11
8             West Ham United                           9       12
9            AFC Bournemouth                           10       17
10            Leicester City                           11        4
11                   Watford                           12       14
12           Newcastle United                          13        9
13               Southampton                           14       13
14                   Burnley                           15       20
15    Brighton & Hove Albion                           16       15
16                    Fulham                           17       18
17             Crystal Palace                          18       10
18               Cardiff City                          19       19
19          Huddersfield Town                          20       16

#Initl insights
```

Inisghts from the home_table

```
# Arsneal overperfomed, as did bournemouth, burnely

# Leicester underperfomed, as did Newcastle and Crystal Palace

print(ov_new_league_tab_h['league_position_home'].corr(ov_new_league_t
ab_h['exp_rank']))

0.7969924812030074
```

Now we want to replicate this for the away_table

```
away_teams_merge=overall_teams_2[['common_name','xg_against_avg_home',
'xg_for_avg_home']].copy()

away_teams_merge['away_team_name']=overall_teams_2['common_name']

away_teams_merge.drop(columns='common_name',inplace=True)

away_merge=pd.merge(away_teams_merge,stats_from_matches,how='outer',on
='away_team_name')

all_teams_away=away_merge[['Game
Week','away_team_name','home_team_goal_count','xg_for_avg_home','team_
a_xg','team_b_xg','away_team_goal_count']].copy()

all_teams_away.head()

   Game Week away_team_name  home_team_goal_count xg_for_avg_home
team_a_xg  \
```

```
0          2          Arsenal                    3                1.6
2.47
1          4          Arsenal                    2                1.6
1.49
2          5          Arsenal                    1                1.6
0.70
3          8          Arsenal                    1                1.6
1.53
4         10          Arsenal                    2                1.6
1.62

   team_b_xg   away_team_goal_count
0       1.61                      2
1       2.35                      3
2       1.09                      2
3       1.56                      5
4       0.90                      2
```

```
all_teams_away['away_result']=all_teams_away['away_team_goal_count']-
all_teams_away['home_team_goal_count']

all_teams_away['x_goal_diff']=all_teams_away['team_b_xg']-
all_teams_away['team_a_xg']

all_teams_away.head()
```

```
   Game Week away_team_name  home_team_goal_count xg_for_avg_home
team_a_xg  \
0          2          Arsenal                    3                1.6
2.47
1          4          Arsenal                    2                1.6
1.49
2          5          Arsenal                    1                1.6
0.70
3          8          Arsenal                    1                1.6
1.53
4         10          Arsenal                    2                1.6
1.62

   team_b_xg   away_team_goal_count   away_result   x_goal_diff
0       1.61                      2            -1         -0.86
1       2.35                      3             1          0.86
2       1.09                      2             1          0.39
3       1.56                      5             4          0.03
4       0.90                      2             0         -0.72
```

```
# now we create a function

# athough it is a replica of the away function, have labelled it
differently to avoid confusion
```

```
def away_points(x):
    if x>0:
        return 3
    if x==0:
        return 1
    if x<0:
        return 0

all_teams_away['away_points']=all_teams_away['away_result'].apply( lam
bda x:away_points(x))

def awa_xg_results(x):
    if x>0.5:
        return 3
    if x>0.5 and x<0.5:
        return 1
    if x<0.5:
        return 0

all_teams_away['xg_away_points']=all_teams_away['x_goal_diff'].apply(l
ambda x: home_xg_Results(x))

all_teams_away.head()
```

```
   Game Week away_team_name  home_team_goal_count xg_for_avg_home
team_a_xg  \
0          2         Arsenal                     3             1.6
2.47
1          4         Arsenal                     2             1.6
1.49
2          5         Arsenal                     1             1.6
0.70
3          8         Arsenal                     1             1.6
1.53
4         10         Arsenal                     2             1.6
1.62

   team_b_xg  away_team_goal_count  away_result  x_goal_diff
away_points  \
0       1.61                     2           -1        -0.86
0
1       2.35                     3            1         0.86
3
2       1.09                     2            1         0.39
3
3       1.56                     5            4         0.03
3
4       0.90                     2            0        -0.72
1

   xg_away_points
```

```
0              0.0
1              3.0
2              1.0
3              1.0
4              0.0
```

```
all_teams_away['x_goal_diff'].corr(all_teams_away['xg_away_points'])
```

0.8424460921155279

```
away_xg_model=all_teams_away.groupby('away_team_name',as_index=False)
[['xg_away_points','away_points']].sum()
```

```
away_xg_model
```

```
         away_team_name  xg_away_points  away_points
0         AFC Bournemouth            20.0           16
1                 Arsenal            18.0           25
2     Brighton & Hove Albion          8.0           13
3                 Burnley             8.0           17
4             Cardiff City           10.0           14
5                 Chelsea            28.0           30
6          Crystal Palace           14.0           29
7                 Everton            24.0           20
8                  Fulham             8.0            5
9        Huddersfield Town           16.0            7
10          Leicester City           21.0           25
11               Liverpool           36.0           44
12          Manchester City           49.0           44
13        Manchester United           22.0           30
14         Newcastle United            7.0           20
15              Southampton           14.0           16
16        Tottenham Hotspur           22.0           33
17                 Watford           13.0           23
18         West Ham United             8.0           21
19  Wolverhampton Wanderers           19.0           23
```

```
overall_teams_2[['common_name','points_per_game_away']]
```

```
            common_name  points_per_game_away
0                 Arsenal                  1.32
1        Tottenham Hotspur                 1.74
2          Manchester City                 2.32
3           Leicester City                 1.32
4           Crystal Palace                 1.53
5                 Everton                  1.05
6                 Burnley                  0.89
7              Southampton                 0.84
8          AFC Bournemouth                 0.84
9        Manchester United                 1.58
10               Liverpool                 2.32
```

```
11                  Chelsea                          1.58
12          West Ham United                          1.11
13                 Watford                           1.21
14         Newcastle United                          1.05
15            Cardiff City                           0.74
16                  Fulham                           0.26
17   Brighton & Hove Albion                          0.68
18        Huddersfield Town                          0.37
19  Wolverhampton Wanderers                          1.21
```

```python
# the aaway_points are accurate

xa=away_xg_model['xg_away_points']
ya=away_xg_model['away_points']

plt.scatter(xa,ya)
plt.xlabel('exp_away_points')
plt.ylabel('away_points')
plt.show()
```
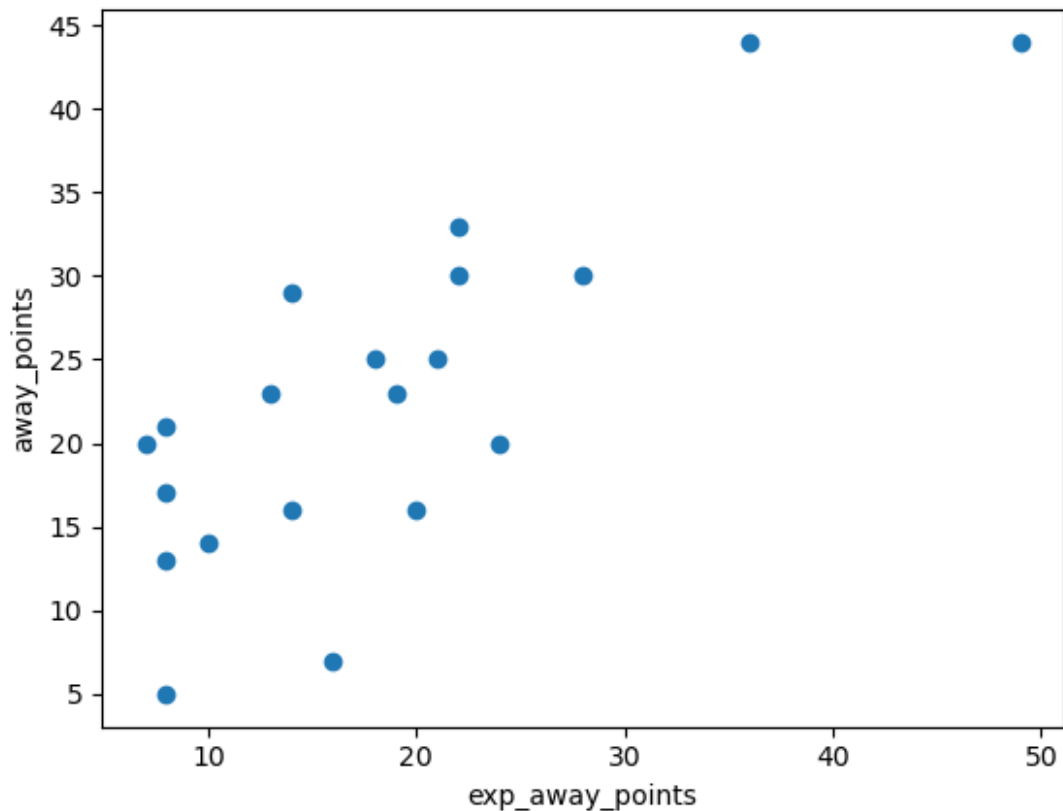


```python
print(xa.corr(ya))
```

```
0.7917606588072454
```

```python
# a slighlty weaker correaltion thatn the home model, but still strong

# Now we find the rank of the exp table and compare it to the actual
table

away_xg_model.head()
```

```
            away_team_name  xg_away_points  away_points
0           AFC Bournemouth            20.0           16
1                   Arsenal            18.0           25
2   Brighton & Hove Albion             8.0           13
3                   Burnley             8.0           17
4               Cardiff City            10.0           14
```

```python
all_teams_away.head()
```

```
   Game Week away_team_name  home_team_goal_count xg_for_avg_home
team_a_xg  \
0          2         Arsenal                     3             1.6
2.47
1          4         Arsenal                     2             1.6
1.49
2          5         Arsenal                     1             1.6
0.70
3          8         Arsenal                     1             1.6
1.53
4         10         Arsenal                     2             1.6
1.62

   team_b_xg  away_team_goal_count  away_result  x_goal_diff
away_points  \
0       1.61                     2           -1        -0.86
0
1       2.35                     3            1         0.86
3
2       1.09                     2            1         0.39
3
3       1.56                     5            4         0.03
3
4       0.90                     2            0        -0.72
1

   xg_away_points
0             0.0
1             3.0
2             1.0
3             1.0
4             0.0
```

```python
x_goal_diff_away=all_teams_away.groupby('away_team_name',as_index=False)['x_goal_diff'].sum()
```

```
x_goal_diff.head()

           home_team_name   x_goal_diff
0          AFC Bournemouth         -2.46
1                  Arsenal          6.02
2    Brighton & Hove Albion        -1.91
3                  Burnley         -3.82
4              Cardiff City        -2.89

new_league_table_away=x_goal_diff_away.merge(away_xg_model,on='away_te
am_name',how='inner')

new_league_table_away['exp_rank']=new_league_table_away[['x_goal_diff'
,'xg_away_points']].apply(tuple,axis=1).rank(method='dense',ascending=
False).astype(int)

new_league_table_away=new_league_table_away.sort_values('exp_rank')

new_league_table_away
```

	away_team_name	x_goal_diff	xg_away_points	away_points
12	Manchester City	17.09	49.0	44
11	Liverpool	8.67	36.0	44
5	Chelsea	5.13	28.0	30
7	Everton	0.65	24.0	20
16	Tottenham Hotspur	-0.78	22.0	33
10	Leicester City	-1.78	21.0	25
13	Manchester United	-1.89	22.0	30
19	Wolverhampton Wanderers	-1.99	19.0	23
0	AFC Bournemouth	-2.59	20.0	16
15	Southampton	-6.00	14.0	16
1	Arsenal	-6.53	18.0	25
17	Watford	-6.62	13.0	23
9	Huddersfield Town	-9.76	16.0	7
6	Crystal Palace	-10.36	14.0	29
18	West Ham United	-10.72	8.0	21

14	Newcastle United	-11.62	7.0	20
8	Fulham	-13.45	8.0	5
4	Cardiff City	-13.71	10.0	14
2	Brighton & Hove Albion	-15.55	8.0	13
3	Burnley	-17.20	8.0	17

```
    exp_rank
12         1
11         2
5          3
7          4
16         5
10         6
13         7
19         8
0          9
15        10
1         11
17        12
9         13
6         14
18        15
14        16
8         17
4         18
2         19
3         20
```

```python
league_position_away=overall_teams_2[['common_name','league_position_a
way']].sort_values( by='league_position_away', ascending=True)

league_position_away.head()
```

```
        common_name league_position_away
2      Manchester City                    1
10            Liverpool                    2
1    Tottenham Hotspur                    3
9    Manchester United                    4
11             Chelsea                    5
```

```python
league_position_away['away_team_name']=league_position_away['common_na
me']

ov_new_leageu_tab_aw=pd.merge(league_position_away,new_league_table_aw
ay,on='away_team_name',how='inner')
```

```
ov_new_leageu_tab_aw

                  common_name  league_position_away  \
away_team_name
0                Manchester City                    1              Manchester
City
1                      Liverpool                    2
Liverpool
2              Tottenham Hotspur                    3              Tottenham
Hotspur
3              Manchester United                    4              Manchester
United
4                        Chelsea                    5
Chelsea
5                  Crystal Palace                   6                 Crystal
Palace
6                  Leicester City                   7               Leicester
City
7                        Arsenal                    8
Arsenal
8                        Watford                    9
Watford
9         Wolverhampton Wanderers                  10   Wolverhampton
Wanderers
10              West Ham United                    11               West Ham
United
11                       Everton                   12
Everton
12              Newcastle United                   13               Newcastle
United
13                       Burnley                   14
Burnley
14                   Southampton                   15
Southampton
15               AFC Bournemouth                   16                     AFC
Bournemouth
16                  Cardiff City                   17                 Cardiff
City
17       Brighton & Hove Albion                   18       Brighton & Hove
Albion
18             Huddersfield Town                   19            Huddersfield
Town
19                         Fulham                   20
Fulham

    x_goal_diff  xg_away_points  away_points  exp_rank
0         17.09            49.0           44         1
1          8.67            36.0           44         2
2         -0.78            22.0           33         5
3         -1.89            22.0           30         7
```

```
4          5.13              28.0              30               3
5         -10.36             14.0              29              14
6          -1.78             21.0              25               6
7          -6.53             18.0              25              11
8          -6.62             13.0              23              12
9          -1.99             19.0              23               8
10        -10.72              8.0              21              15
11          0.65             24.0              20               4
12        -11.62              7.0              20              16
13        -17.20              8.0              17              20
14         -6.00             14.0              16              10
15         -2.59             20.0              16               9
16        -13.71             10.0              14              18
17        -15.55              8.0              13              19
18         -9.76             16.0               7              13
19        -13.45              8.0               5              17
```

```python
ov_new_leageu_tab_aw[['away_team_name','league_position_away','exp_ran
k']]
```

```
            away_team_name  league_position_away  exp_rank
0            Manchester City                     1         1
1                  Liverpool                     2         2
2          Tottenham Hotspur                     3         5
3          Manchester United                     4         7
4                    Chelsea                     5         3
5             Crystal Palace                     6        14
6             Leicester City                     7         6
7                    Arsenal                     8        11
8                    Watford                     9        12
9     Wolverhampton Wanderers                    10         8
10           West Ham United                    11        15
11                   Everton                    12         4
12          Newcastle United                    13        16
13                   Burnley                    14        20
14               Southampton                    15        10
15           AFC Bournemouth                    16         9
16               Cardiff City                    17        18
17     Brighton & Hove Albion                    18        19
18          Huddersfield Town                    19        13
19                     Fulham                    20        17
```

```python
# In terms of overperformers, Palace,Watford,West ham,Burnley

# Underperformers, AFC bournemouth,SOuthamption,Everton
```

Now we do the overall league

```
ov_new_league_tab_aw=ov_new_leageu_tab_aw.sort_values(by='exp_rank',as
cending=True)

ov_new_leageu_tab_aw=ov_new_leageu_tab_aw.sort_values(by='exp_rank',as
cending=True)

ov_new_leageu_tab_aw.head()
```

```
         common_name league_position_away     away_team_name
x_goal_diff  \
0      Manchester City                    1     Manchester City
17.09
1            Liverpool                    2           Liverpool
8.67
4              Chelsea                    5             Chelsea
5.13
11             Everton                   12             Everton
0.65
2    Tottenham Hotspur                    3   Tottenham Hotspur          -
0.78

    xg_away_points  away_points  exp_rank
0            49.0           44         1
1            36.0           44         2
4            28.0           30         3
11           24.0           20         4
2            22.0           33         5
```

```
ov_new_league_tab_h
```

```
              common_name  league_position_home
home_team_name  \
0         Manchester City                     1         Manchester
City
1               Liverpool                     2
Liverpool
2                 Arsenal                     3
Arsenal
3                 Chelsea                     4
Chelsea
4       Tottenham Hotspur                     5         Tottenham
Hotspur
5       Manchester United                     6         Manchester
United
6                 Everton                     7
Everton
7   Wolverhampton Wanderers                   8   Wolverhampton
Wanderers
```

8	West Ham United	9	West Ham United
9	AFC Bournemouth	10	AFC Bournemouth
10	Leicester City	11	Leicester City
11	Watford	12	
12	Newcastle United	13	Newcastle United
13	Southampton	14	
14	Burnley	15	
15	Brighton & Hove Albion	16	Brighton & Hove Albion
16	Fulham	17	
17	Crystal Palace	18	Crystal Palace
18	Cardiff City	19	Cardiff City
19	Huddersfield Town	20	Huddersfield Town

	x_goal_diff	xg_home_points	home_points	exp_rank
0	28.20	55.0	54	1
1	20.58	51.0	53	2
2	6.02	34.0	45	8
3	14.79	49.0	42	3
4	7.81	35.0	38	5
5	7.33	36.0	36	6
6	6.81	32.0	34	7
7	3.48	25.0	34	11
8	3.08	28.0	31	12
9	-2.46	22.0	29	17
10	8.88	33.0	27	4
11	-1.33	19.0	27	14
12	3.90	25.0	25	9
13	1.51	22.0	23	13
14	-3.82	21.0	23	20
15	-1.91	21.0	23	15
16	-2.57	21.0	21	18
17	3.66	33.0	20	10
18	-2.89	23.0	20	19
19	-2.06	23.0	9	16

```python
# we want to only merge certain columsn
```

```
ov_new_leageu_tab_spec_cols_aw=ov_new_leageu_tab_aw.loc[:,
['common_name','x_goal_diff','xg_away_points','away_points']]

ov_new_league_tab_h_spec_cols_ho=ov_new_league_tab_h.loc[:,
['common_name','x_goal_diff','xg_home_points','home_points']]

overall_comb_h_away_exp_tab=pd.merge(ov_new_leageu_tab_spec_cols_aw,ov
_new_league_tab_h_spec_cols_ho,on='common_name',how="outer")

overall_comb_h_away_exp_tab
```

	common_name	x_goal_diff_x	xg_away_points	away_points
0	Manchester City	17.09	49.0	44
1	Liverpool	8.67	36.0	44
2	Chelsea	5.13	28.0	30
3	Everton	0.65	24.0	20
4	Tottenham Hotspur	-0.78	22.0	33
5	Leicester City	-1.78	21.0	25
6	Manchester United	-1.89	22.0	30
7	Wolverhampton Wanderers	-1.99	19.0	23
8	AFC Bournemouth	-2.59	20.0	16
9	Southampton	-6.00	14.0	16
10	Arsenal	-6.53	18.0	25
11	Watford	-6.62	13.0	23
12	Huddersfield Town	-9.76	16.0	7
13	Crystal Palace	-10.36	14.0	29
14	West Ham United	-10.72	8.0	21
15	Newcastle United	-11.62	7.0	20
16	Fulham	-13.45	8.0	5
17	Cardiff City	-13.71	10.0	14
18	Brighton & Hove Albion	-15.55	8.0	

```
13
19                  Burnley          -17.20              8.0
17

    x_goal_diff_y  xg_home_points  home_points
0          28.20            55.0           54
1          20.58            51.0           53
2          14.79            49.0           42
3           6.81            32.0           34
4           7.81            35.0           38
5           8.88            33.0           27
6           7.33            36.0           36
7           3.48            25.0           34
8          -2.46            22.0           29
9           1.51            22.0           23
10          6.02            34.0           45
11         -1.33            19.0           27
12         -2.06            23.0            9
13          3.66            33.0           20
14          3.08            28.0           31
15          3.90            25.0           25
16         -2.57            21.0           21
17         -2.89            23.0           20
18         -1.91            21.0           23
19         -3.82            21.0           23
```

overall_comb_h_away_exp_tab['total_exp_points']=overall_comb_h_away_ex
p_tab['xg_away_points']+ overall_comb_h_away_exp_tab['xg_home_points']

overall_comb_h_away_exp_tab['total_exp_g_diff']=overall_comb_h_away_ex
p_tab['x_goal_diff_x']+overall_comb_h_away_exp_tab['x_goal_diff_y']

overall_comb_h_away_exp_tab.head()

```
         common_name  x_goal_diff_x  xg_away_points  away_points  \
0      Manchester City          17.09            49.0           44
1           Liverpool           8.67            36.0           44
2             Chelsea           5.13            28.0           30
3             Everton           0.65            24.0           20
4   Tottenham Hotspur          -0.78            22.0           33

    x_goal_diff_y  xg_home_points  home_points  total_exp_points  \
0          28.20            55.0           54             104.0
1          20.58            51.0           53              87.0
2          14.79            49.0           42              77.0
3           6.81            32.0           34              56.0
4           7.81            35.0           38              57.0

    total_exp_g_diff
0             45.29
```

```
1              29.25
2              19.92
3               7.46
4               7.03
```

```python
overall_comb_h_away_exp_tab['actual_points']=overall_comb_h_away_exp_t
ab['away_points']+ overall_comb_h_away_exp_tab['home_points']
```

```python
overall_comb_h_away_exp_tab.head()
```

```
        common_name  x_goal_diff_x  xg_away_points  away_points  \
0     Manchester City          17.09            49.0           44
1           Liverpool           8.67            36.0           44
2             Chelsea           5.13            28.0           30
3             Everton           0.65            24.0           20
4   Tottenham Hotspur          -0.78            22.0           33

   x_goal_diff_y  xg_home_points  home_points  total_exp_points  \
0          28.20            55.0           54             104.0
1          20.58            51.0           53              87.0
2          14.79            49.0           42              77.0
3           6.81            32.0           34              56.0
4           7.81            35.0           38              57.0

   total_exp_g_diff  actual_points
0             45.29             98
1             29.25             97
2             19.92             72
3              7.46             54
4              7.03             71
```

```python
overall_comb_h_away_exp_tab['exp_rank']=overall_comb_h_away_exp_tab[['
total_exp_points','total_exp_g_diff',]].apply(tuple,axis=1).rank(metho
d='dense',ascending=False).astype(int)
```

```python
overall_comb_h_away_exp_tab.sort_values(by
='total_exp_points',ascending=False)
```

```
               common_name  x_goal_diff_x  xg_away_points
away_points  \
0          Manchester City          17.09            49.0
44
1                Liverpool           8.67            36.0
44
2                  Chelsea           5.13            28.0
30
6        Manchester United          -1.89            22.0
30
4        Tottenham Hotspur          -0.78            22.0
33
3                  Everton           0.65            24.0
```

```
20
5                Leicester City          -1.78         21.0
25
10                      Arsenal          -6.53         18.0
25
13                Crystal Palace         -10.36        14.0
29
7     Wolverhampton Wanderers            -1.99         19.0
23
8                AFC Bournemouth          -2.59        20.0
16
12             Huddersfield Town          -9.76        16.0
7
9                    Southampton          -6.00        14.0
16
14              West Ham United          -10.72         8.0
21
17                  Cardiff City         -13.71        10.0
14
11                      Watford          -6.62         13.0
23
15             Newcastle United          -11.62         7.0
20
16                       Fulham          -13.45         8.0
5
18      Brighton & Hove Albion           -15.55         8.0
13
19                      Burnley          -17.20         8.0
17
```

```
    x_goal_diff_y  xg_home_points  home_points  total_exp_points  \
0          28.20            55.0           54             104.0
1          20.58            51.0           53              87.0
2          14.79            49.0           42              77.0
6           7.33            36.0           36              58.0
4           7.81            35.0           38              57.0
3           6.81            32.0           34              56.0
5           8.88            33.0           27              54.0
10          6.02            34.0           45              52.0
13          3.66            33.0           20              47.0
7           3.48            25.0           34              44.0
8          -2.46            22.0           29              42.0
12         -2.06            23.0            9              39.0
9           1.51            22.0           23              36.0
14          3.08            28.0           31              36.0
17         -2.89            23.0           20              33.0
11         -1.33            19.0           27              32.0
15          3.90            25.0           25              32.0
16         -2.57            21.0           21              29.0
```

```
18            -1.91          21.0          23          29.0
19            -3.82          21.0          23          29.0

    total_exp_g_diff  actual_points  exp_rank
0              45.29             98         1
1              29.25             97         2
2              19.92             72         3
6               5.44             66         4
4               7.03             71         5
3               7.46             54         6
5               7.10             52         7
10             -0.51             70         8
13             -6.70             49         9
7               1.49             57        10
8              -5.05             45        11
12            -11.82             16        12
9              -4.49             39        13
14             -7.64             52        14
17            -16.60             34        15
11             -7.95             50        17
15             -7.72             45        16
16            -16.02             26        18
18            -17.46             36        19
19            -21.02             40        20
```

Check the line with the tuple is the right away round

```
overall_comb_h_away_exp_tab_fin=overall_comb_h_away_exp_tab.loc[:,
['common_name','total_exp_points','total_exp_g_diff','exp_rank','actua
l_points']].copy()

overall_comb_h_away_exp_tab_fin
```

	common_name	total_exp_points	total_exp_g_diff
exp_rank			
0 1	Manchester City	104.0	45.29
1 2	Liverpool	87.0	29.25
2 3	Chelsea	77.0	19.92
3 6	Everton	56.0	7.46
4 5	Tottenham Hotspur	57.0	7.03
5	Leicester City	54.0	7.10

7			
6	Manchester United	58.0	5.44
4			
7	Wolverhampton Wanderers	44.0	1.49
10			
8	AFC Bournemouth	42.0	-5.05
11			
9	Southampton	36.0	-4.49
13			
10	Arsenal	52.0	-0.51
8			
11	Watford	32.0	-7.95
17			
12	Huddersfield Town	39.0	-11.82
12			
13	Crystal Palace	47.0	-6.70
9			
14	West Ham United	36.0	-7.64
14			
15	Newcastle United	32.0	-7.72
16			
16	Fulham	29.0	-16.02
18			
17	Cardiff City	33.0	-16.60
15			
18	Brighton & Hove Albion	29.0	-17.46
19			
19	Burnley	29.0	-21.02
20			

	actual_points
0	98
1	97
2	72
3	54
4	71
5	52
6	66
7	57
8	45
9	39
10	70
11	50
12	16
13	49
14	52
15	45
16	26
17	34

```
18          36
19          40
```

```
add_fin_tab=overall_teams_2[['common_name','goal_difference','league_p
osition']]
```

```
add_fin_tab
```

	common_name	goal_difference	league_position
0	Arsenal	22	5
1	Tottenham Hotspur	28	4
2	Manchester City	72	1
3	Leicester City	3	9
4	Crystal Palace	-2	12
5	Everton	8	8
6	Burnley	-23	15
7	Southampton	-20	16
8	AFC Bournemouth	-14	14
9	Manchester United	11	6
10	Liverpool	67	2
11	Chelsea	24	3
12	West Ham United	-3	10
13	Watford	-7	11
14	Newcastle United	-6	13
15	Cardiff City	-35	18
16	Fulham	-47	19
17	Brighton & Hove Albion	-25	17
18	Huddersfield Town	-54	20
19	Wolverhampton Wanderers	1	7

```
overall_comb_h_away_exp_tab_fin=pd.merge(overall_comb_h_away_exp_tab_f
in,add_fin_tab,on='common_name',how='outer')
```

```
overall_comb_h_away_exp_tab_fin
```

	common_name	total_exp_points	total_exp_g_diff	exp_rank
0	Manchester City	104.0	45.29	1
1	Liverpool	87.0	29.25	2
2	Chelsea	77.0	19.92	3
3	Everton	56.0	7.46	6
4	Tottenham Hotspur	57.0	7.03	5
5	Leicester City	54.0	7.10	7
6	Manchester United	58.0	5.44	4

7	Wolverhampton Wanderers	44.0	1.49
10			
8	AFC Bournemouth	42.0	-5.05
11			
9	Southampton	36.0	-4.49
13			
10	Arsenal	52.0	-0.51
8			
11	Watford	32.0	-7.95
17			
12	Huddersfield Town	39.0	-11.82
12			
13	Crystal Palace	47.0	-6.70
9			
14	West Ham United	36.0	-7.64
14			
15	Newcastle United	32.0	-7.72
16			
16	Fulham	29.0	-16.02
18			
17	Cardiff City	33.0	-16.60
15			
18	Brighton & Hove Albion	29.0	-17.46
19			
19	Burnley	29.0	-21.02
20			

	actual_points	goal_difference	league_position
0	98	72	1
1	97	67	2
2	72	24	3
3	54	8	8
4	71	28	4
5	52	3	9
6	66	11	6
7	57	1	7
8	45	-14	14
9	39	-20	16
10	70	22	5
11	50	-7	11
12	16	-54	20
13	49	-2	12
14	52	-3	10
15	45	-6	13
16	26	-47	19
17	34	-35	18
18	36	-25	17
19	40	-23	15

```python
print(overall_comb_h_away_exp_tab_fin['exp_rank'].corr(overall_comb_h_away_exp_tab_fin['league_position']))
```

```
0.8330827067669173
```

```python
# Correlation between expected table and final table
```

```python
overall_comb_h_away_exp_tab_fin['diff_rank']=pd.Series.abs(overall_comb_h_away_exp_tab_fin['exp_rank']-
overall_comb_h_away_exp_tab_fin['league_position'])
```

```python
variance=overall_comb_h_away_exp_tab_fin['diff_rank'].var()
```

```python
variance
```

```
4.010526315789474
```

```python
from math import sqrt
```

```python
sqrt(variance)
```

```
2.0026298499197184
```

```python
# tells us how far on avaergae the results are close tothe mean
```

```python
overall_comb_h_away_exp_tab_fin
```

```
                    common_name  total_exp_points  total_exp_g_diff
exp_rank  \
0              Manchester City             104.0             45.29
1
1                    Liverpool              87.0             29.25
2
2                      Chelsea              77.0             19.92
3
3                      Everton              56.0              7.46
6
4            Tottenham Hotspur              57.0              7.03
5
5                Leicester City              54.0              7.10
7
6            Manchester United              58.0              5.44
4
7     Wolverhampton Wanderers              44.0              1.49
10
8              AFC Bournemouth              42.0             -5.05
11
9                  Southampton              36.0             -4.49
13
10                     Arsenal              52.0             -0.51
8
11                     Watford              32.0             -7.95
```

```
17
12         Huddersfield Town                39.0              -11.82
12
13           Crystal Palace                 47.0               -6.70
9
14          West Ham United                 36.0               -7.64
14
15         Newcastle United                 32.0               -7.72
16
16                   Fulham                 29.0              -16.02
18
17             Cardiff City                 33.0              -16.60
15
18   Brighton & Hove Albion                 29.0              -17.46
19
19                  Burnley                 29.0              -21.02
20

     actual_points goal_difference  league_position  diff_rank
0               98              72                 1          0
1               97              67                 2          0
2               72              24                 3          0
3               54               8                 8          2
4               71              28                 4          1
5               52               3                 9          2
6               66              11                 6          2
7               57               1                 7          3
8               45             -14                14          3
9               39             -20                16          3
10              70              22                 5          3
11              50              -7                11          6
12              16             -54                20          8
13              49              -2                12          3
14              52              -3                10          4
15              45              -6                13          3
16              26             -47                19          1
17              34             -35                18          3
18              36             -25                17          2
19              40             -23                15          5
```

```python
import random
```

```python
overall_comb_h_away_exp_tab_fin['random']=pd.Series(range(0,20,1))
```

```python
new_var=overall_comb_h_away_exp_tab_fin['random'].var()
```

```python
new_var
```

```
35.0
```

```python
sqrt(new_var)
```

```
5.916079783099616
```

```
overall_comb_h_away_exp_tab_fin['diff_rank'].std()
```

```
2.0026298499197184
```

```
#coefficeint of varitation
```

```
CV=overall_comb_h_away_exp_tab_fin['diff_rank'].std()/
overall_comb_h_away_exp_tab_fin['diff_rank'].mean()
```

```
CV
```

```
0.7417147592295252
```

```
# maybe if we compared standard deviation to another type of model
```

```
x=overall_comb_h_away_exp_tab_fin['diff_rank']
```

```
plt.hist(x)
plt.show()
```
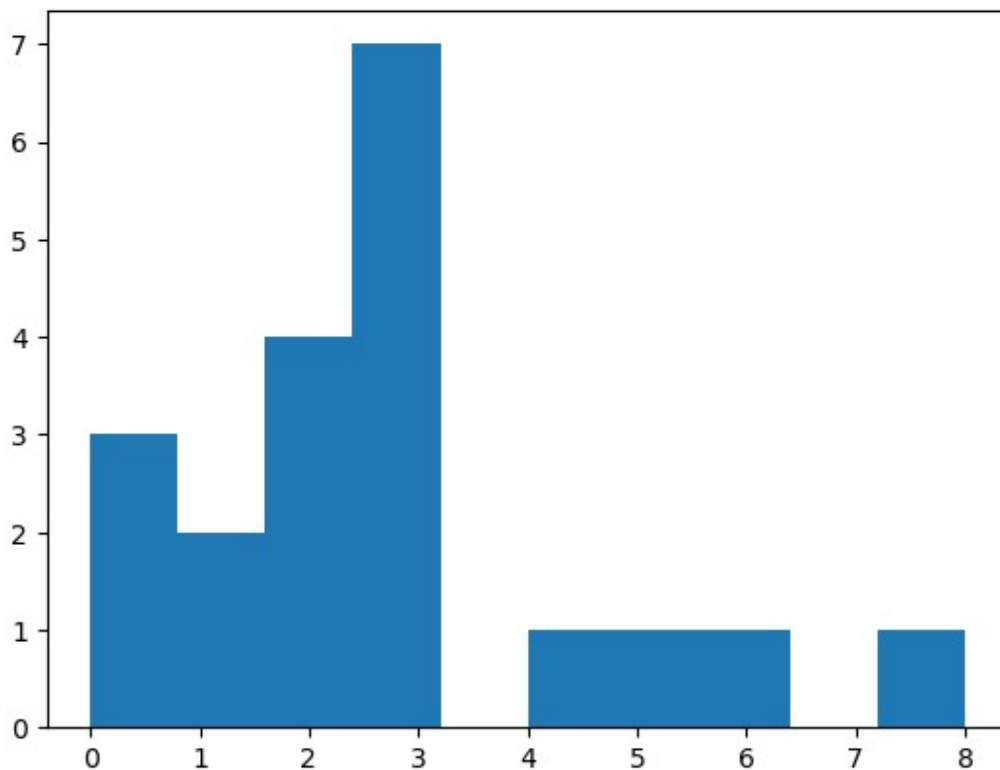


```
overall_comb_h_away_exp_tab_fin['diff_rank'].mean()
```

```
2.7
```

```
# We can see most of the dsitribution fits under the mean change in
rank, but some outliers skew the means

print(overall_comb_h_away_exp_tab_fin['diff_rank'].median())

3.0

#Let's mev
```