In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn import metrics
from sklearn.metrics import accuracy_score
from sklearn.feature_selection import SelectKBest, chi2, f_regression
from sklearn.neighbors import NearestNeighbors
from sklearn.neighbors import KNeighborsClassifier
from sklearn import svm
```

In [2]:
```
df_results = pd.read_csv("results-0.2.csv")
```

In [3]:
```
df_results.head(21)
```

Out[3]:

| | League | Year | position | team | matches | wins | draws | loses | scored | missed | pts |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | Bundesliga | 2014 | 1 | Bayern Munich | 34 | 25 | 4 | 5 | 80 | 18 | 79 |
| **1** | Bundesliga | 2014 | 2 | Wolfsburg | 34 | 20 | 9 | 5 | 72 | 38 | 69 |
| **2** | Bundesliga | 2014 | 3 | Borussia M.Gladbach | 34 | 19 | 9 | 6 | 53 | 26 | 66 |
| **3** | Bundesliga | 2014 | 4 | Bayer Leverkusen | 34 | 17 | 10 | 7 | 62 | 37 | 61 |
| **4** | Bundesliga | 2014 | 5 | Augsburg | 34 | 15 | 4 | 15 | 43 | 43 | 49 |
| **5** | Bundesliga | 2014 | 6 | Schalke 04 | 34 | 13 | 9 | 12 | 42 | 40 | 48 |
| **6** | Bundesliga | 2014 | 7 | Borussia Dortmund | 34 | 13 | 7 | 14 | 47 | 42 | 46 |
| **7** | Bundesliga | 2014 | 8 | Hoffenheim | 34 | 12 | 8 | 14 | 49 | 55 | 44 |
| **8** | Bundesliga | 2014 | 9 | Werder Bremen | 34 | 11 | 10 | 13 | 50 | 65 | 43 |
| **9** | Bundesliga | 2014 | 10 | Eintracht Frankfurt | 34 | 11 | 10 | 13 | 56 | 62 | 43 |
| **10** | Bundesliga | 2014 | 11 | FC Cologne | 34 | 9 | 13 | 12 | 34 | 40 | 40 |
| **11** | Bundesliga | 2014 | 12 | Mainz 05 | 34 | 9 | 13 | 12 | 45 | 47 | 40 |
| **12** | Bundesliga | 2014 | 13 | Hannover 96 | 34 | 9 | 10 | 15 | 40 | 56 | 37 |
| **13** | Bundesliga | 2014 | 14 | VfB Stuttgart | 34 | 9 | 9 | 16 | 42 | 60 | 36 |
| **14** | Bundesliga | 2014 | 15 | Hamburger SV | 34 | 9 | 8 | 17 | 25 | 50 | 35 |
| **15** | Bundesliga | 2014 | 16 | Hertha Berlin | 34 | 9 | 8 | 17 | 36 | 52 | 35 |
| **16** | Bundesliga | 2014 | 17 | Freiburg | 34 | 7 | 13 | 14 | 36 | 47 | 34 |
| **17** | Bundesliga | 2014 | 18 | Paderborn | 34 | 7 | 10 | 17 | 31 | 65 | 31 |
| **18** | Bundesliga | 2015 | 1 | Bayern Munich | 34 | 28 | 4 | 2 | 80 | 17 | 88 |
| **19** | Bundesliga | 2015 | 2 | Borussia Dortmund | 34 | 24 | 6 | 4 | 82 | 34 | 78 |
| **20** | Bundesliga | 2015 | 3 | Bayer Leverkusen | 34 | 18 | 6 | 10 | 56 | 40 | 60 |

In [4]:
```
df_results.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 570 entries, 0 to 569
Data columns (total 11 columns):
League      570 non-null object
Year        570 non-null int64
position    570 non-null int64
team        570 non-null object
matches     570 non-null int64
wins        570 non-null int64
draws       570 non-null int64
loses       570 non-null int64
scored      570 non-null int64
missed      570 non-null int64
pts         570 non-null int64
dtypes: int64(9), object(2)
memory usage: 49.1+ KB
```

In [5]:
```
df_results['League'].unique()
```

Out[5]:
```
array(['Bundesliga', 'EPL', 'La_liga', 'Ligue_1', 'RFPL', 'Serie_A'],
      dtype=object)
```

In [6]:
```
df_results.describe()
```

Out[6]:

| | Year | position | matches | wins | draws | loses | scored | |
|---|---|---|---|---|---|---|---|---|
| **count** | 570.000000 | 570.000000 | 570.000000 | 570.000000 | 570.000000 | 570.000000 | 570.000000 | 57 |
| **mean** | 2016.000000 | 10.061404 | 36.245614 | 13.531579 | 9.182456 | 13.531579 | 48.385965 | 4 |
| **std** | 1.415456 | 5.580982 | 2.906152 | 5.935200 | 2.927064 | 5.540700 | 17.634599 | 1 |
| **min** | 2014.000000 | 1.000000 | 30.000000 | 2.000000 | 2.000000 | 1.000000 | 13.000000 | 1 |
| **25%** | 2015.000000 | 5.000000 | 34.000000 | 9.000000 | 7.000000 | 10.000000 | 36.000000 | 3 |
| **50%** | 2016.000000 | 10.000000 | 38.000000 | 12.000000 | 9.000000 | 14.000000 | 45.000000 | 4 |
| **75%** | 2017.000000 | 15.000000 | 38.000000 | 17.000000 | 11.000000 | 17.000000 | 56.000000 | 5 |
| **max** | 2018.000000 | 20.000000 | 38.000000 | 32.000000 | 18.000000 | 29.000000 | 118.000000 | 9 |

In [7]:
```
df_results['LeagueFill'] = df_results['League'].map({'Bundesliga':0, 'EPL':1,
'La_liga':2, 'Ligue_1':3,
                                                     'RFPL': 4, 'Serie_A':5})
```

In [8]:
```
team_list = df_results['team'].unique()
numer_list = []

count = df_results['team'].nunique()

for i in range(count):
    numer_list.append(i)

mapped_frame = pd.DataFrame(team_list)
number_frame = pd.DataFrame(numer_list)

zipObj = zip(team_list, numer_list)
diction = dict(zipObj)
df_results['TeamFill'] = df_results['team'].map(diction)
df_results.head(30)
```

Out[8]:

| | League | Year | position | team | matches | wins | draws | loses | scored | missed | pts |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | Bundesliga | 2014 | 1 | Bayern Munich | 34 | 25 | 4 | 5 | 80 | 18 | 79 |
| **1** | Bundesliga | 2014 | 2 | Wolfsburg | 34 | 20 | 9 | 5 | 72 | 38 | 69 |
| **2** | Bundesliga | 2014 | 3 | Borussia M.Gladbach | 34 | 19 | 9 | 6 | 53 | 26 | 66 |
| **3** | Bundesliga | 2014 | 4 | Bayer Leverkusen | 34 | 17 | 10 | 7 | 62 | 37 | 61 |
| **4** | Bundesliga | 2014 | 5 | Augsburg | 34 | 15 | 4 | 15 | 43 | 43 | 49 |
| **5** | Bundesliga | 2014 | 6 | Schalke 04 | 34 | 13 | 9 | 12 | 42 | 40 | 48 |
| **6** | Bundesliga | 2014 | 7 | Borussia Dortmund | 34 | 13 | 7 | 14 | 47 | 42 | 46 |
| **7** | Bundesliga | 2014 | 8 | Hoffenheim | 34 | 12 | 8 | 14 | 49 | 55 | 44 |
| **8** | Bundesliga | 2014 | 9 | Werder Bremen | 34 | 11 | 10 | 13 | 50 | 65 | 43 |
| **9** | Bundesliga | 2014 | 10 | Eintracht Frankfurt | 34 | 11 | 10 | 13 | 56 | 62 | 43 |
| **10** | Bundesliga | 2014 | 11 | FC Cologne | 34 | 9 | 13 | 12 | 34 | 40 | 40 |
| **11** | Bundesliga | 2014 | 12 | Mainz 05 | 34 | 9 | 13 | 12 | 45 | 47 | 40 |
| **12** | Bundesliga | 2014 | 13 | Hannover 96 | 34 | 9 | 10 | 15 | 40 | 56 | 37 |
| **13** | Bundesliga | 2014 | 14 | VfB Stuttgart | 34 | 9 | 9 | 16 | 42 | 60 | 36 |
| **14** | Bundesliga | 2014 | 15 | Hamburger SV | 34 | 9 | 8 | 17 | 25 | 50 | 35 |
| **15** | Bundesliga | 2014 | 16 | Hertha Berlin | 34 | 9 | 8 | 17 | 36 | 52 | 35 |
| **16** | Bundesliga | 2014 | 17 | Freiburg | 34 | 7 | 13 | 14 | 36 | 47 | 34 |
| **17** | Bundesliga | 2014 | 18 | Paderborn | 34 | 7 | 10 | 17 | 31 | 65 | 31 |
| **18** | Bundesliga | 2015 | 1 | Bayern Munich | 34 | 28 | 4 | 2 | 80 | 17 | 88 |
| **19** | Bundesliga | 2015 | 2 | Borussia Dortmund | 34 | 24 | 6 | 4 | 82 | 34 | 78 |
| **20** | Bundesliga | 2015 | 3 | Bayer Leverkusen | 34 | 18 | 6 | 10 | 56 | 40 | 60 |
| **21** | Bundesliga | 2015 | 4 | Borussia M.Gladbach | 34 | 17 | 4 | 13 | 67 | 50 | 55 |
| **22** | Bundesliga | 2015 | 5 | Schalke 04 | 34 | 15 | 7 | 12 | 51 | 49 | 52 |
| **23** | Bundesliga | 2015 | 6 | Mainz 05 | 34 | 14 | 8 | 12 | 46 | 42 | 50 |
| **24** | Bundesliga | 2015 | 7 | Hertha Berlin | 34 | 14 | 8 | 12 | 42 | 42 | 50 |
| **25** | Bundesliga | 2015 | 8 | Wolfsburg | 34 | 12 | 9 | 13 | 47 | 49 | 45 |
| **26** | Bundesliga | 2015 | 9 | FC Cologne | 34 | 10 | 13 | 11 | 38 | 42 | 43 |

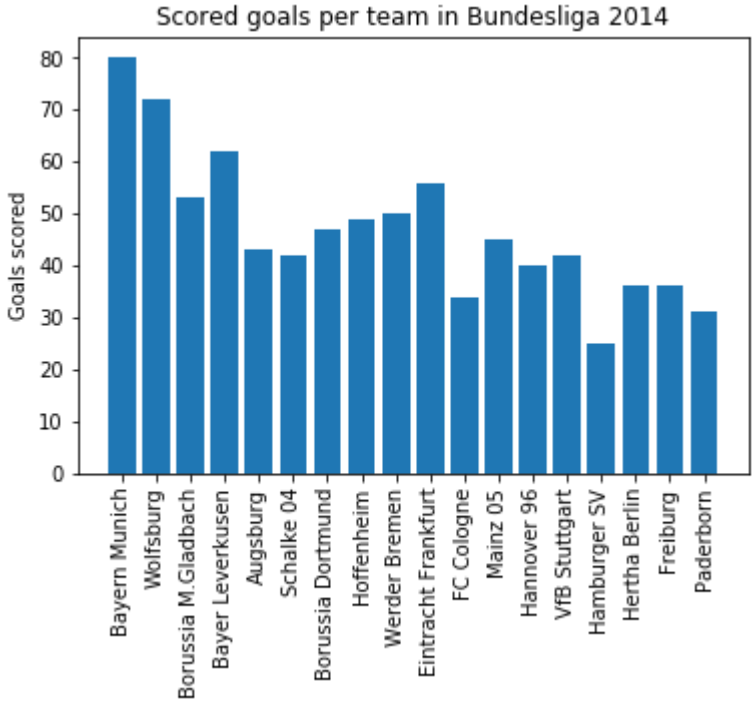
| | League | Year | position | team | matches | wins | draws | loses | scored | missed | pts |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **27** | Bundesliga | 2015 | 10 | Hamburger SV | 34 | 11 | 8 | 15 | 40 | 46 | 41 |
| **28** | Bundesliga | 2015 | 11 | Ingolstadt | 34 | 10 | 10 | 14 | 33 | 42 | 40 |
| **29** | Bundesliga | 2015 | 12 | Darmstadt | 34 | 9 | 11 | 14 | 38 | 53 | 38 |

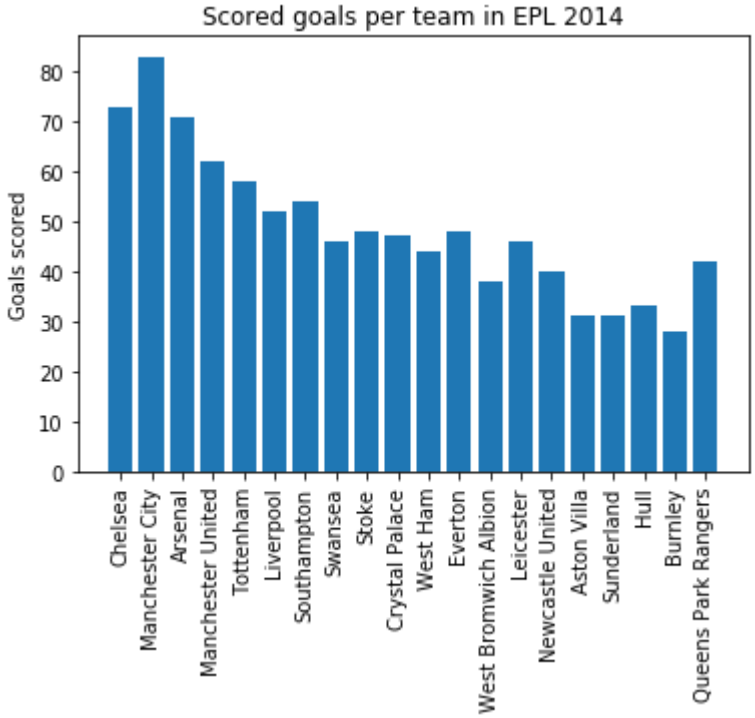
In [9]:
```
df_bundes_2014 = df_results.loc[(df_results['League'] == 'Bundesliga')&(df_res
ults['Year']==2014)]
df_bundes_2015 = df_results.loc[(df_results['League'] == 'Bundesliga')&(df_res
ults['Year']==2015)]
df_bundes_2016 = df_results.loc[(df_results['League'] == 'Bundesliga')&(df_res
ults['Year']==2016)]
df_bundes_2017 = df_results.loc[(df_results['League'] == 'Bundesliga')&(df_res
ults['Year']==2017)]
df_bundes_2018 = df_results.loc[(df_results['League'] == 'Bundesliga')&(df_res
ults['Year']==2018)]

df_epl_2014 = df_results.loc[(df_results['League'] == 'EPL')&(df_results['Yea
r']==2014)]
df_epl_2015 = df_results.loc[(df_results['League'] == 'EPL')&(df_results['Yea
r']==2015)]
df_epl_2016 = df_results.loc[(df_results['League'] == 'EPL')&(df_results['Yea
r']==2016)]
df_epl_2017 = df_results.loc[(df_results['League'] == 'EPL')&(df_results['Yea
r']==2017)]
df_epl_2018 = df_results.loc[(df_results['League'] == 'EPL')&(df_results['Yea
r']==2018)]

df_laliga_2014 = df_results.loc[(df_results['League'] == 'La_liga')&(df_result
s['Year']==2014)]
df_laliga_2015 = df_results.loc[(df_results['League'] == 'La_liga')&(df_result
s['Year']==2015)]
df_laliga_2016 = df_results.loc[(df_results['League'] == 'La_liga')&(df_result
s['Year']==2016)]
df_laliga_2017 = df_results.loc[(df_results['League'] == 'La_liga')&(df_result
s['Year']==2017)]
df_laliga_2018 = df_results.loc[(df_results['League'] == 'La_liga')&(df_result
s['Year']==2018)]

df_ligue1_2014 = df_results.loc[(df_results['League'] == 'Ligue_1')&(df_result
s['Year']==2014)]
df_ligue1_2015 = df_results.loc[(df_results['League'] == 'Ligue_1')&(df_result
s['Year']==2015)]
df_ligue1_2016 = df_results.loc[(df_results['League'] == 'Ligue_1')&(df_result
s['Year']==2016)]
df_ligue1_2017 = df_results.loc[(df_results['League'] == 'Ligue_1')&(df_result
s['Year']==2017)]
df_ligue1_2018 = df_results.loc[(df_results['League'] == 'Ligue_1')&(df_result
s['Year']==2018)]

df_RFPL_2014 = df_results.loc[(df_results['League'] == 'RFPL')&(df_results['Ye
ar']==2014)]
df_RFPL_2015 = df_results.loc[(df_results['League'] == 'RFPL')&(df_results['Ye
ar']==2015)]
df_RFPL_2016 = df_results.loc[(df_results['League'] == 'RFPL')&(df_results['Ye
ar']==2016)]
df_RFPL_2017 = df_results.loc[(df_results['League'] == 'RFPL')&(df_results['Ye
ar']==2017)]
df_RFPL_2018 = df_results.loc[(df_results['League'] == 'RFPL')&(df_results['Ye
ar']==2018)]

df_SerieA_2014 = df_results.loc[(df_results['League'] == 'Serie_A')&(df_result
```

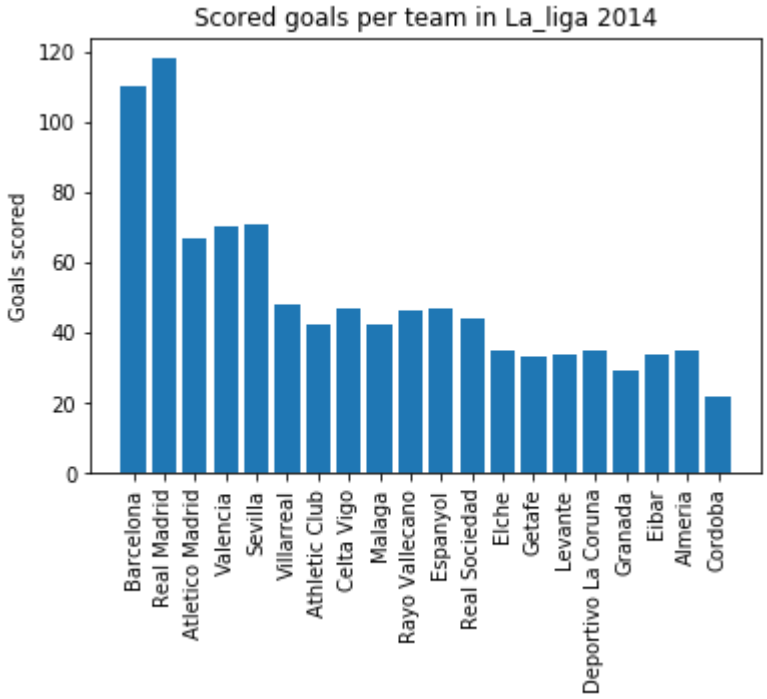
```
s['Year']==2014)]
df_SerieA_2015 = df_results.loc[(df_results['League'] == 'Serie_A')&(df_result
s['Year']==2015)]
df_SerieA_2016 = df_results.loc[(df_results['League'] == 'Serie_A')&(df_result
s['Year']==2016)]
df_SerieA_2017 = df_results.loc[(df_results['League'] == 'Serie_A')&(df_result
s['Year']==2017)]
df_SerieA_2018 = df_results.loc[(df_results['League'] == 'Serie_A')&(df_result
s['Year']==2018)]
```

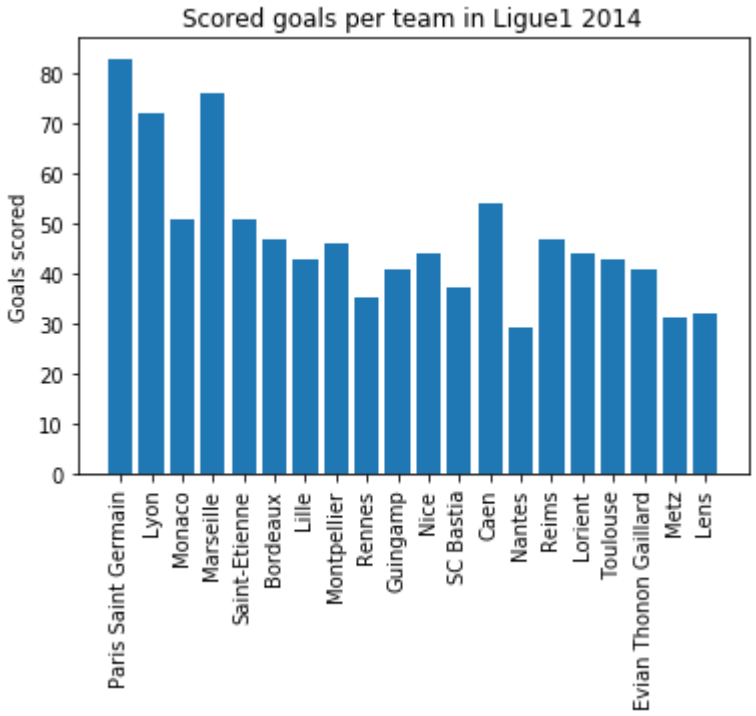
In [10]:
```
color_list = [df_SerieA_2014['position']]
```

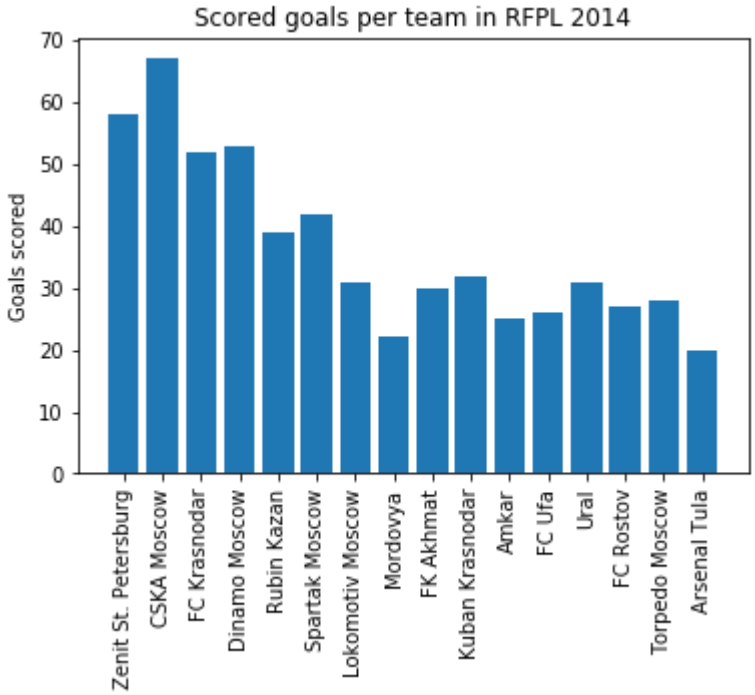
In [11]:
```
##Goals scored per team per league 2014
plt.bar(df_bundes_2014['team'], df_bundes_2014['scored'])
plt.ylabel("Goals scored")
plt.title("Scored goals per team in Bundesliga 2014")
plt.xticks(rotation=90)
plt.show()
plt.bar(df_epl_2014['team'], df_epl_2014['scored'])
plt.ylabel("Goals scored")
plt.title("Scored goals per team in EPL 2014")
plt.xticks(rotation=90)
plt.show()
plt.bar(df_laliga_2014['team'], df_laliga_2014['scored'])
plt.ylabel("Goals scored")
plt.title("Scored goals per team in La_liga 2014")
plt.xticks(rotation=90)
plt.show()
plt.bar(df_ligue1_2014['team'], df_ligue1_2014['scored'])
plt.ylabel("Goals scored")
plt.title("Scored goals per team in Ligue1 2014")
plt.xticks(rotation=90)
plt.show()
plt.bar(df_RFPL_2014['team'], df_RFPL_2014['scored'])
plt.ylabel("Goals scored")
plt.title("Scored goals per team in RFPL 2014")
plt.xticks(rotation=90)
plt.show()
plt.bar(df_SerieA_2014['team'], df_SerieA_2014['scored'])
plt.ylabel("Goals scored")
plt.title("Scored goals per team in Serie_A 2014")
plt.xticks(rotation=90)
plt.show()
```

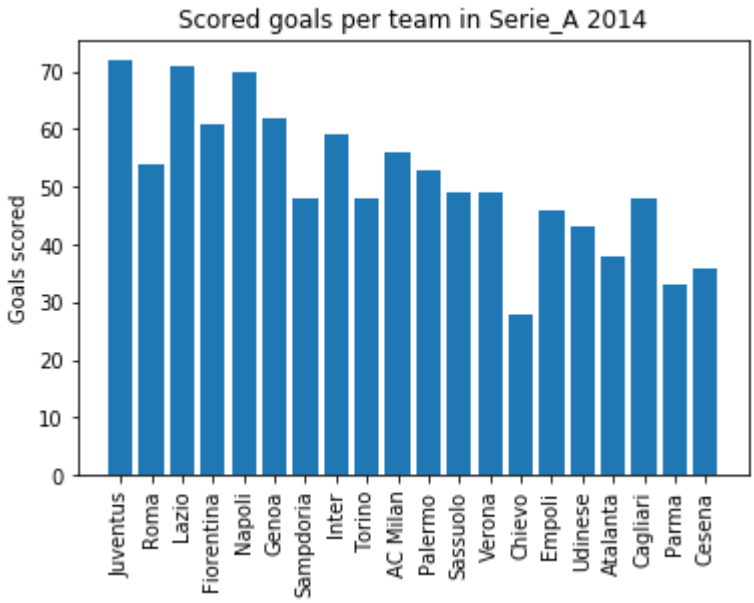

Scored goals per team in Bundesliga 2014
Goals scored
0
10
20
30
40
50
60
70
80
Bayern Munich
Wolfsburg
Borussia M.Gladbach
Bayer Leverkusen
Augsburg
Schalke 04
Borussia Dortmund
Hoffenheim
Werder Bremen
Eintracht Frankfurt
FC Cologne
Mainz 05
Hannover 96
VfB Stuttgart
Hamburger SV
Hertha Berlin
Freiburg
Paderborn


Scored goals per team in EPL 2014
Goals scored
0
10
20
30
40
50
60
70
80
Chelsea
Manchester City
Arsenal
Manchester United
Tottenham
Liverpool
Southampton
Swansea
Stoke
Crystal Palace
West Ham
Everton
West Bromwich Albion
Leicester
Newcastle United
Aston Villa
Sunderland
Hull
Burnley
Queens Park Rangers

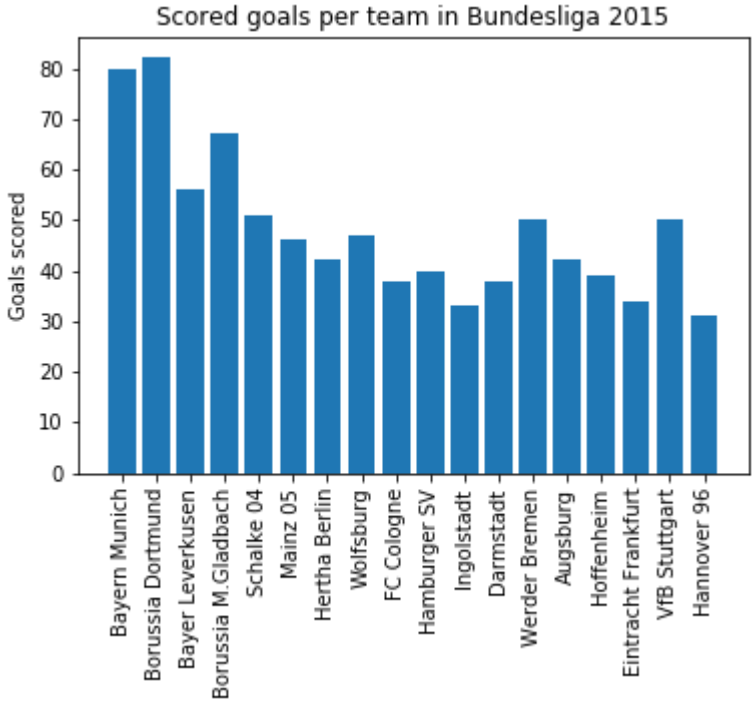

Scored goals per team in La_liga 2014
Goals scored
0
20
40
60
80
100
120
Barcelona
Real Madrid
Atletico Madrid
Valencia
Sevilla
Villarreal
Athletic Club
Celta Vigo
Malaga
Rayo Vallecano
Espanyol
Real Sociedad
Elche
Getafe
Levante
Deportivo La Coruna
Granada
Eibar
Almeria
Cordoba

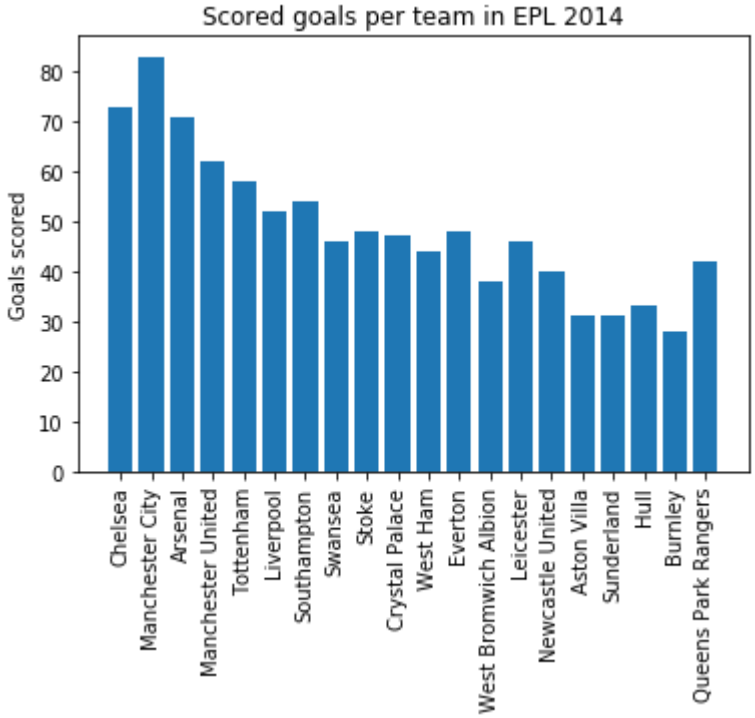

Scored goals per team in Ligue1 2014
Goals scored
0
10
20
30
40
50
60
70
80
Paris Saint Germain
Lyon
Monaco
Marseille
Saint-Etienne
Bordeaux
Lille
Montpellier
Rennes
Guingamp
Nice
SC Bastia
Caen
Nantes
Reims
Lorient
Toulouse
Evian Thonon Gaillard
Metz
Lens

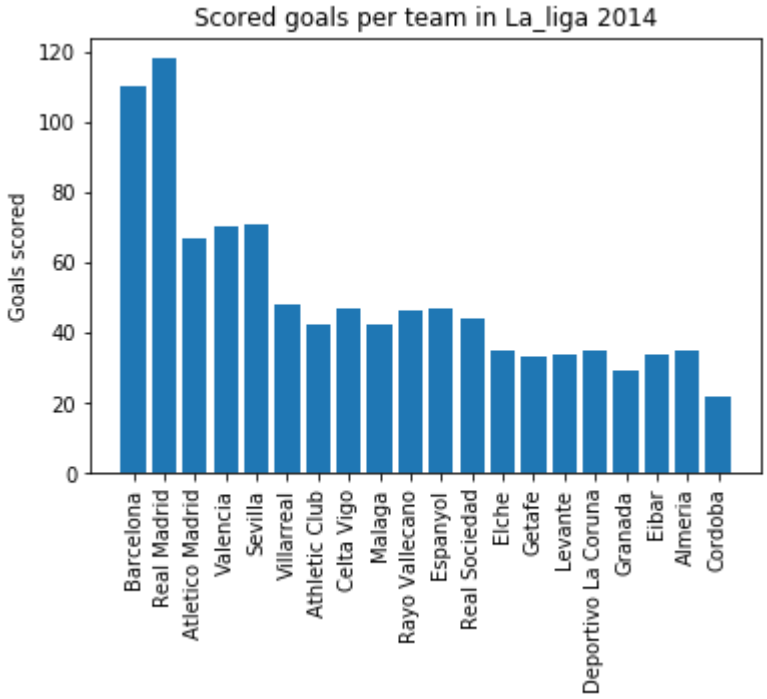

Scored goals per team in RFPL 2014
Goals scored
0
10
20
30
40
50
60
70
Zenit St. Petersburg
CSKA Moscow
FC Krasnodar
Dinamo Moscow
Rubin Kazan
Spartak Moscow
Lokomotiv Moscow
Mordovya
FK Akhmat
Kuban Krasnodar
Amkar
FC Ufa
Ural
FC Rostov
Torpedo Moscow
Arsenal Tula

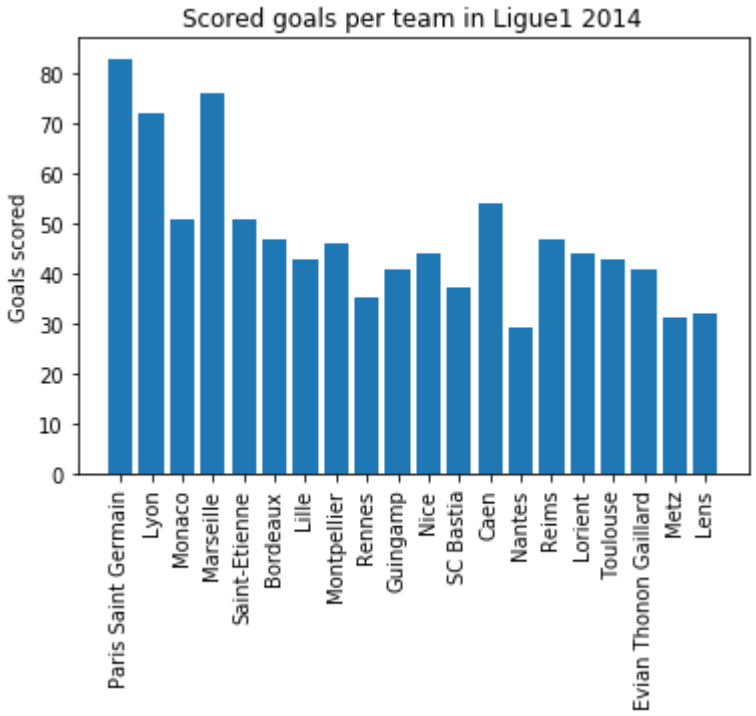

Scored goals per team in Serie_A 2014
Goals scored
0
10
20
30
40
50
60
70
Juventus
Roma
Lazio
Fiorentina
Napoli
Genoa
Sampdoria
Inter
Torino
AC Milan
Palermo
Sassuolo
Verona
Chievo
Empoli
Udinese
Atalanta
Cagliari
Parma
Cesena

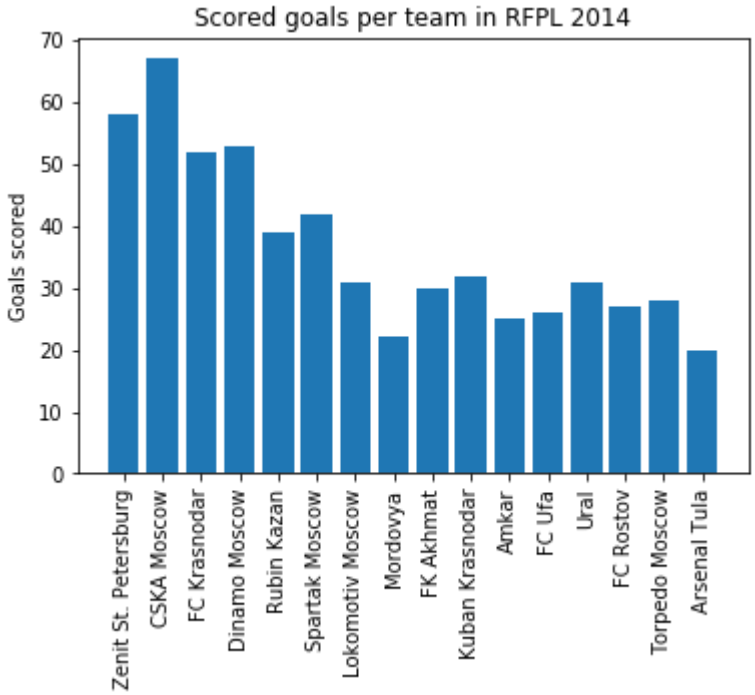
In [12]:
```
plt.bar(df_bundes_2015['team'], df_bundes_2015['scored'])
plt.ylabel("Goals scored")
plt.title("Scored goals per team in Bundesliga 2015")
plt.xticks(rotation=90)
plt.show()
plt.bar(df_epl_2014['team'], df_epl_2014['scored'])
plt.ylabel("Goals scored")
plt.title("Scored goals per team in EPL 2014")
plt.xticks(rotation=90)
plt.show()
plt.bar(df_laliga_2014['team'], df_laliga_2014['scored'])
plt.ylabel("Goals scored")
plt.title("Scored goals per team in La_liga 2014")
plt.xticks(rotation=90)
plt.show()
plt.bar(df_ligue1_2014['team'], df_ligue1_2014['scored'])
plt.ylabel("Goals scored")
plt.title("Scored goals per team in Ligue1 2014")
plt.xticks(rotation=90)
plt.show()
plt.bar(df_RFPL_2014['team'], df_RFPL_2014['scored'])
plt.ylabel("Goals scored")
plt.title("Scored goals per team in RFPL 2014")
plt.xticks(rotation=90)
plt.show()
plt.bar(df_SerieA_2014['team'], df_SerieA_2014['scored'])
plt.ylabel("Goals scored")
plt.title("Scored goals per team in Serie_A 2014")
plt.xticks(rotation=90)
plt.show()
```

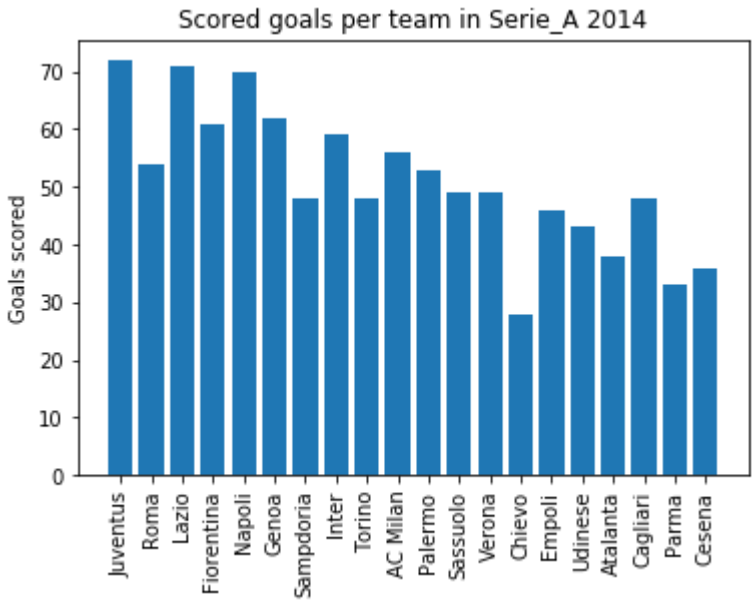

Scored goals per team in Bundesliga 2015
Goals scored
0
10
20
30
40
50
60
70
80
Bayern Munich
Borussia Dortmund
Bayer Leverkusen
Borussia M.Gladbach
Schalke 04
Mainz 05
Hertha Berlin
Wolfsburg
FC Cologne
Hamburger SV
Ingolstadt
Darmstadt
Werder Bremen
Augsburg
Hoffenheim
Eintracht Frankfurt
VfB Stuttgart
Hannover 96


Scored goals per team in EPL 2014
Goals scored
0
10
20
30
40
50
60
70
80
Chelsea
Manchester City
Arsenal
Manchester United
Tottenham
Liverpool
Southampton
Swansea
Stoke
Crystal Palace
West Ham
Everton
West Bromwich Albion
Leicester
Newcastle United
Aston Villa
Sunderland
Hull
Burnley
Queens Park Rangers

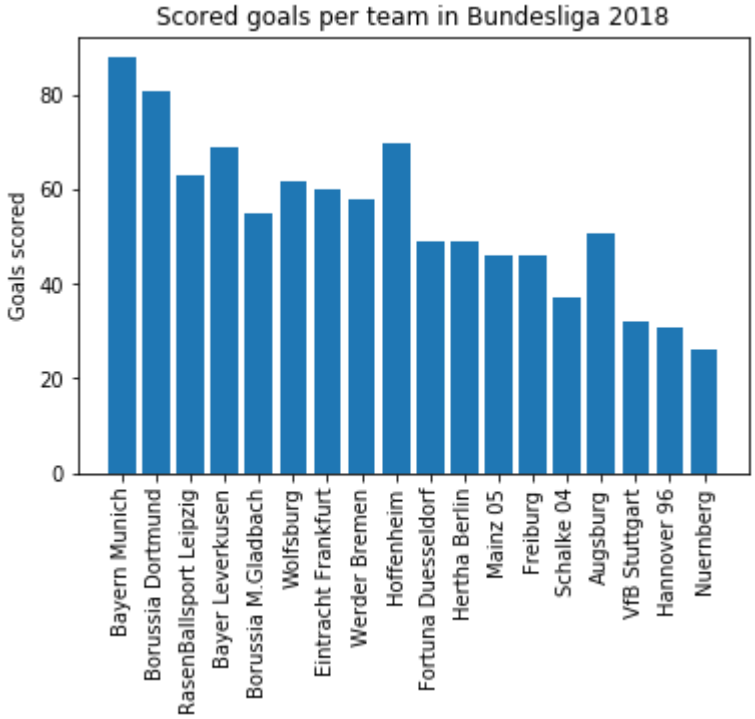

Scored goals per team in La_liga 2014
Goals scored
0
20
40
60
80
100
120
Barcelona
Real Madrid
Atletico Madrid
Valencia
Sevilla
Villarreal
Athletic Club
Celta Vigo
Malaga
Rayo Vallecano
Espanyol
Real Sociedad
Elche
Getafe
Levante
Deportivo La Coruna
Granada
Eibar
Almeria
Cordoba

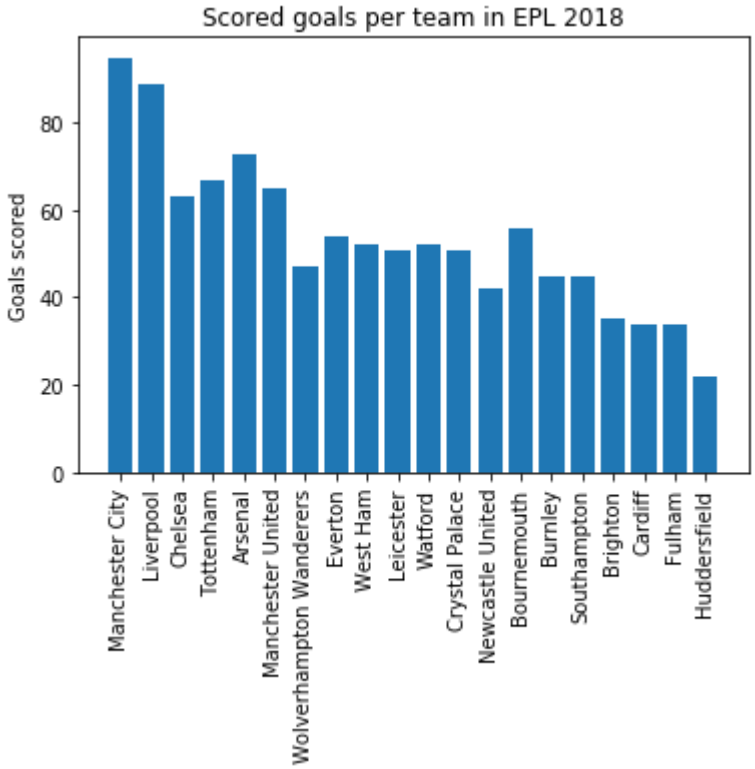

Scored goals per team in Ligue1 2014
Goals scored
0
10
20
30
40
50
60
70
80
Paris Saint Germain
Lyon
Monaco
Marseille
Saint-Etienne
Bordeaux
Lille
Montpellier
Rennes
Guingamp
Nice
SC Bastia
Caen
Nantes
Reims
Lorient
Toulouse
Evian Thonon Gaillard
Metz
Lens

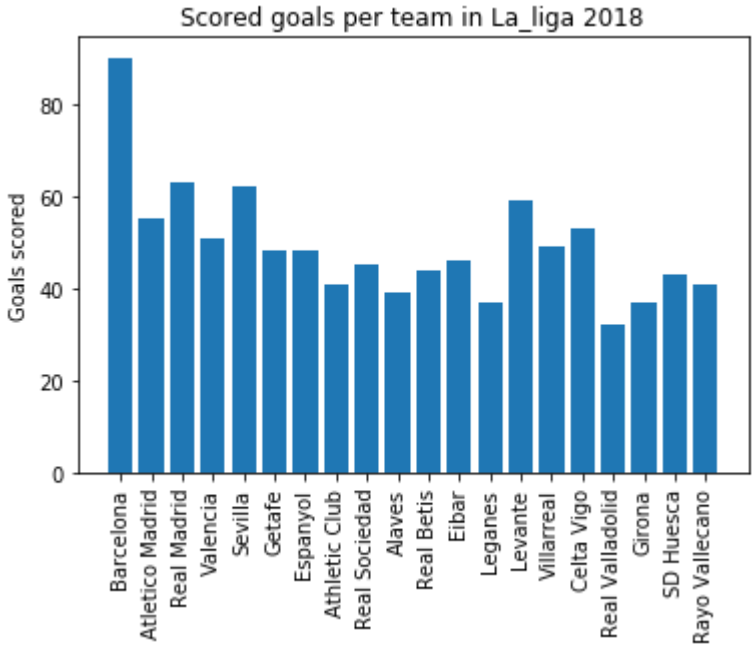

Scored goals per team in RFPL 2014
Goals scored
0
10
20
30
40
50
60
70
Zenit St. Petersburg
CSKA Moscow
FC Krasnodar
Dinamo Moscow
Rubin Kazan
Spartak Moscow
Lokomotiv Moscow
Mordovya
FK Akhmat
Kuban Krasnodar
Amkar
FC Ufa
Ural
FC Rostov
Torpedo Moscow
Arsenal Tula

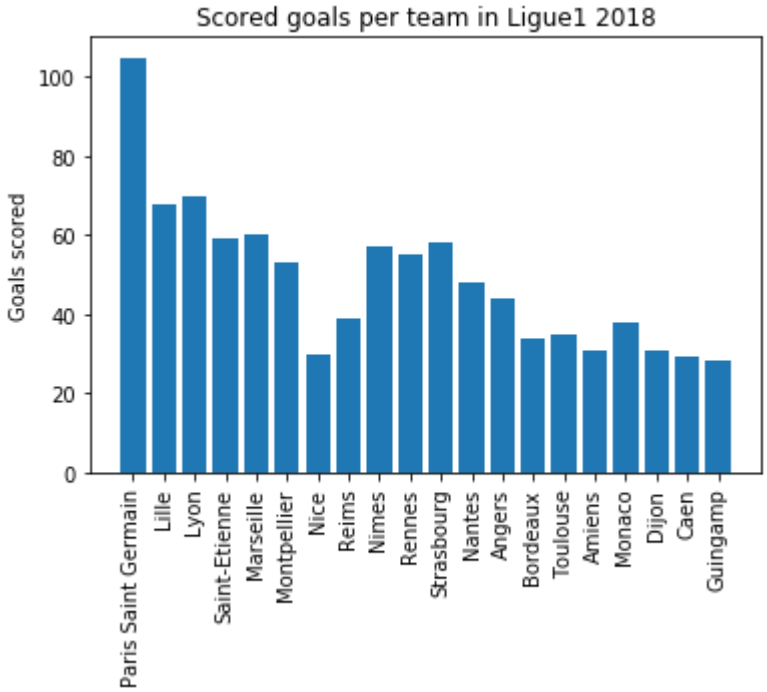

Scored goals per team in Serie_A 2014
Goals scored
0
10
20
30
40
50
60
70
Juventus
Roma
Lazio
Fiorentina
Napoli
Genoa
Sampdoria
Inter
Torino
AC Milan
Palermo
Sassuolo
Verona
Chievo
Empoli
Udinese
Atalanta
Cagliari
Parma
Cesena

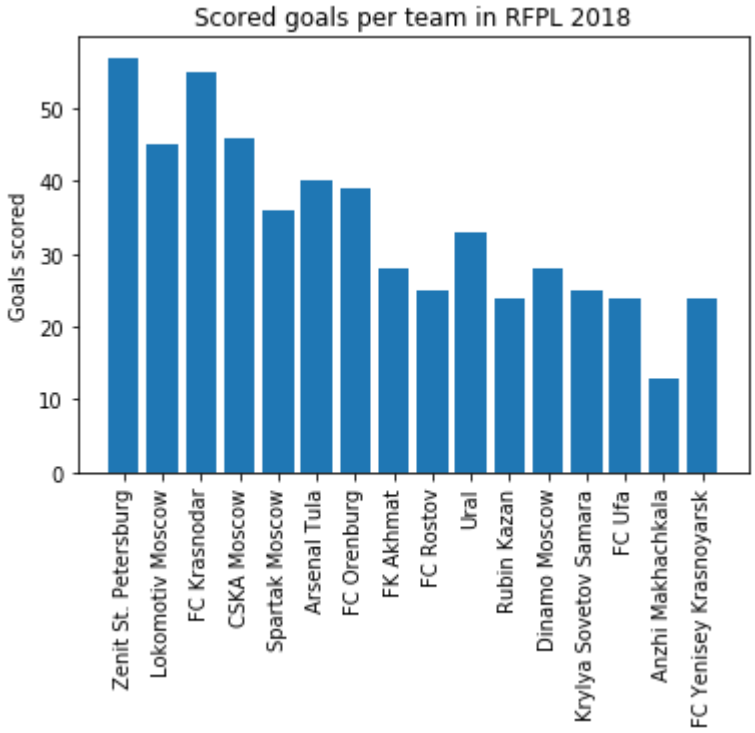
In [13]:
```
##Goals scored per team per league 2018
plt.bar(df_bundes_2018['team'], df_bundes_2018['scored'])
plt.ylabel("Goals scored")
plt.title("Scored goals per team in Bundesliga 2018")
plt.xticks(rotation=90)
plt.show()
plt.bar(df_epl_2018['team'], df_epl_2018['scored'])
plt.ylabel("Goals scored")
plt.title("Scored goals per team in EPL 2018")
plt.xticks(rotation=90)
plt.show()
plt.bar(df_laliga_2018['team'], df_laliga_2018['scored'])
plt.ylabel("Goals scored")
plt.title("Scored goals per team in La_liga 2018")
plt.xticks(rotation=90)
plt.show()
plt.bar(df_ligue1_2018['team'], df_ligue1_2018['scored'])
plt.ylabel("Goals scored")
plt.title("Scored goals per team in Ligue1 2018")
plt.xticks(rotation=90)
plt.show()
plt.bar(df_RFPL_2018['team'], df_RFPL_2018['scored'])
plt.ylabel("Goals scored")
plt.title("Scored goals per team in RFPL 2018")
plt.xticks(rotation=90)
plt.show()
plt.bar(df_SerieA_2018['team'], df_SerieA_2018['scored'])
plt.ylabel("Goals scored")
plt.title("Scored goals per team in Serie_A 2018")
plt.xticks(rotation=90)
plt.show()
```

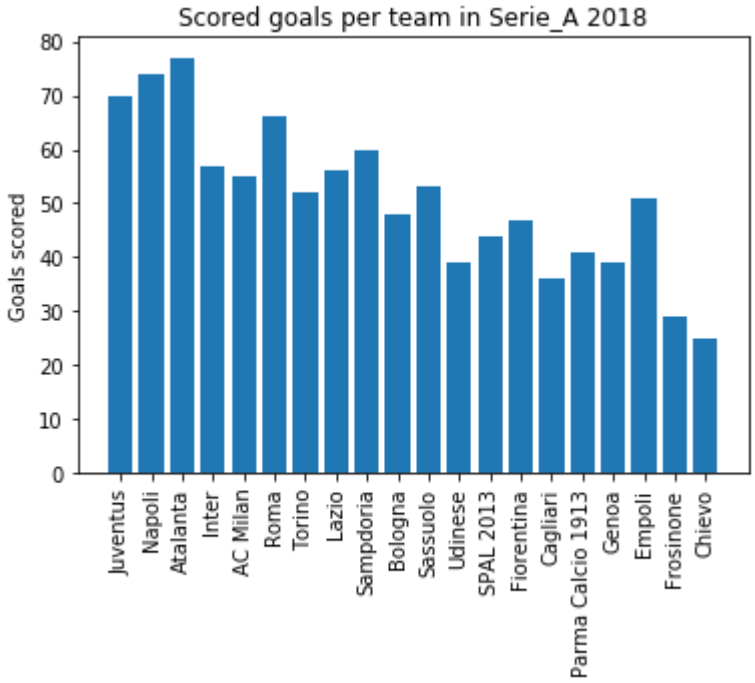

Scored goals per team in Bundesliga 2018
Goals scored
0
20
40
60
80
Bayern Munich
Borussia Dortmund
RasenBallsport Leipzig
Bayer Leverkusen
Borussia M.Gladbach
Wolfsburg
Eintracht Frankfurt
Werder Bremen
Hoffenheim
Fortuna Duesseldorf
Hertha Berlin
Mainz 05
Freiburg
Schalke 04
Augsburg
VfB Stuttgart
Hannover 96
Nuernberg


Scored goals per team in EPL 2018
Goals scored
0
20
40
60
80
Manchester City
Liverpool
Chelsea
Tottenham
Arsenal
Manchester United
Wolverhampton Wanderers
Everton
West Ham
Leicester
Watford
Crystal Palace
Newcastle United
Bournemouth
Burnley
Southampton
Brighton
Cardiff
Fulham
Huddersfield


Scored goals per team in La_liga 2018
Goals scored
0
20
40
60
80
Barcelona
Atletico Madrid
Real Madrid
Valencia
Sevilla
Getafe
Espanyol
Athletic Club
Real Sociedad
Alaves
Real Betis
Eibar
Leganes
Levante
Villarreal
Celta Vigo
Real Valladolid
Girona
SD Huesca
Rayo Vallecano

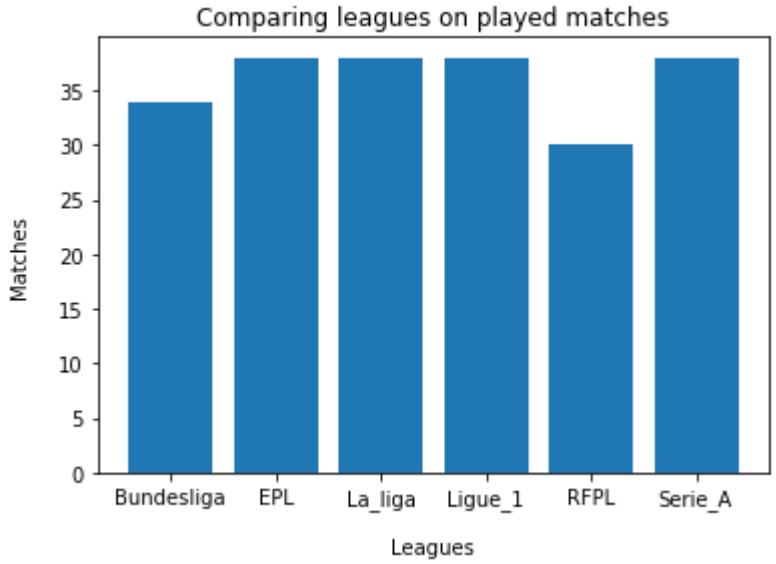

Scored goals per team in Ligue1 2018
Goals scored
0
20
40
60
80
100
Paris Saint Germain
Lille
Lyon
Saint-Etienne
Marseille
Montpellier
Nice
Reims
Nimes
Rennes
Strasbourg
Nantes
Angers
Bordeaux
Toulouse
Amiens
Monaco
Dijon
Caen
Guingamp


Scored goals per team in RFPL 2018
Goals scored
0
10
20
30
40
50
Zenit St. Petersburg
Lokomotiv Moscow
FC Krasnodar
CSKA Moscow
Spartak Moscow
Arsenal Tula
FC Orenburg
FK Akhmat
FC Rostov
Ural
Rubin Kazan
Dinamo Moscow
Krylya Sovetov Samara
FC Ufa
Anzhi Makhachkala
FC Yenisey Krasnoyarsk


Scored goals per team in Serie_A 2018
Goals scored
0
10
20
30
40
50
60
70
80
Juventus
Napoli
Atalanta
Inter
AC Milan
Roma
Torino
Lazio
Sampdoria
Bologna
Sassuolo
Udinese
SPAL 2013
Fiorentina
Cagliari
Parma Calcio 1913
Genoa
Empoli
Frosinone
Chievo

In [14]:
```
plt.bar(df_results['League'],df_results['matches'],align='center', alpha=0.5)
plt.xlabel("Leagues", labelpad=14)
plt.ylabel("Matches", labelpad=14)
plt.title("Comparing leagues on played matches")
```

Out[14]: Text(0.5, 1.0, 'Comparing leagues on played matches')



In [15]:
```
#Making dataframes for each league every year
df_bundes_top3_2014_rows = df_results.loc[(df_results['League'] == 'Bundeslig
a')&
                                          (df_results['Year']==2014)]
df_bundes_top3_2014 = df_bundes_top3_2014_rows[0:3]
df_bundes_top3_2015_rows = df_results.loc[(df_results['League'] == 'Bundeslig
a')&
                                          (df_results['Year']==2015)]
df_bundes_top3_2015 = df_bundes_top3_2015_rows[0:3]
df_bundes_top3_2016_rows = df_results.loc[(df_results['League'] == 'Bundeslig
a')&
                                          (df_results['Year']==2016)]
df_bundes_top3_2016 = df_bundes_top3_2016_rows[0:3]
df_bundes_top3_2017_rows = df_results.loc[(df_results['League'] == 'Bundeslig
a')&
                                          (df_results['Year']==2017)]
df_bundes_top3_2017 = df_bundes_top3_2017_rows[0:3]
df_bundes_top3_2018_rows = df_results.loc[(df_results['League'] == 'Bundeslig
a')&
                                          (df_results['Year']==2018)]
df_bundes_top3_2018 = df_bundes_top3_2018_rows[0:3]

df_epl_top3_2014_rows = df_results.loc[(df_results['League'] == 'EPL')&
                                          (df_results['Year']==2014)]
df_epl_top3_2014 = df_epl_top3_2014_rows[0:3]
df_epl_top3_2015_rows = df_results.loc[(df_results['League'] == 'EPL')&
                                          (df_results['Year']==2015)]
df_epl_top3_2015 = df_epl_top3_2015_rows[0:3]
df_epl_top3_2016_rows = df_results.loc[(df_results['League'] == 'EPL')&
                                          (df_results['Year']==2016)]
df_epl_top3_2016 = df_epl_top3_2016_rows[0:3]
df_epl_top3_2017_rows = df_results.loc[(df_results['League'] == 'EPL')&
                                          (df_results['Year']==2017)]
df_epl_top3_2017 = df_epl_top3_2017_rows[0:3]
df_epl_top3_2018_rows = df_results.loc[(df_results['League'] == 'EPL')&
                                          (df_results['Year']==2018)]
df_epl_top3_2018 = df_epl_top3_2018_rows[0:3]

df_laliga_top3_2014_rows = df_results.loc[(df_results['League'] == 'La_liga')&
                                          (df_results['Year']==2014)]
df_laliga_top3_2014 = df_laliga_top3_2014_rows[0:3]
df_laliga_top3_2015_rows = df_results.loc[(df_results['League'] == 'La_liga')&
                                          (df_results['Year']==2015)]
df_laliga_top3_2015 = df_laliga_top3_2015_rows[0:3]
df_laliga_top3_2016_rows = df_results.loc[(df_results['League'] == 'La_liga')&
                                          (df_results['Year']==2016)]
df_laliga_top3_2016 = df_laliga_top3_2016_rows[0:3]
df_laliga_top3_2017_rows = df_results.loc[(df_results['League'] == 'La_liga')&
                                          (df_results['Year']==2017)]
df_laliga_top3_2017 = df_laliga_top3_2017_rows[0:3]
df_laliga_top3_2018_rows = df_results.loc[(df_results['League'] == 'La_liga')&
                                          (df_results['Year']==2018)]
df_laliga_top3_2018 = df_laliga_top3_2018_rows[0:3]

df_ligue1_top3_2014_rows = df_results.loc[(df_results['League'] == 'Ligue_1')&
                                          (df_results['Year']==2014)]
```

```
df_ligue1_top3_2014 = df_ligue1_top3_2014_rows[0:3]
df_ligue1_top3_2015_rows = df_results.loc[(df_results['League'] == 'Ligue_1')&
                                          (df_results['Year']==2015)]
df_ligue1_top3_2015 = df_ligue1_top3_2015_rows[0:3]
df_ligue1_top3_2016_rows = df_results.loc[(df_results['League'] == 'Ligue_1')&
                                          (df_results['Year']==2016)]
df_ligue1_top3_2016 = df_ligue1_top3_2016_rows[0:3]
df_ligue1_top3_2017_rows = df_results.loc[(df_results['League'] == 'Ligue_1')&
                                          (df_results['Year']==2017)]
df_ligue1_top3_2017 = df_ligue1_top3_2017_rows[0:3]
df_ligue1_top3_2018_rows = df_results.loc[(df_results['League'] == 'Ligue_1')&
                                          (df_results['Year']==2018)]
df_ligue1_top3_2018 = df_ligue1_top3_2018_rows[0:3]

df_RFPL_top3_2014_rows = df_results.loc[(df_results['League'] == 'RFPL')&
                                          (df_results['Year']==2014)]
df_RFPL_top3_2014 = df_RFPL_top3_2014_rows[0:3]
df_RFPL_top3_2015_rows = df_results.loc[(df_results['League'] == 'RFPL')&
                                          (df_results['Year']==2015)]
df_RFPL_top3_2015 = df_RFPL_top3_2015_rows[0:3]
df_RFPL_top3_2016_rows = df_results.loc[(df_results['League'] == 'RFPL')&
                                          (df_results['Year']==2016)]
df_RFPL_top3_2016 = df_RFPL_top3_2016_rows[0:3]
df_RFPL_top3_2017_rows = df_results.loc[(df_results['League'] == 'RFPL')&
                                          (df_results['Year']==2017)]
df_RFPL_top3_2017 = df_RFPL_top3_2017_rows[0:3]
df_RFPL_top3_2018_rows = df_results.loc[(df_results['League'] == 'RFPL')&
                                          (df_results['Year']==2018)]
df_RFPL_top3_2018 = df_RFPL_top3_2018_rows[0:3]

df_SerieA_top3_2014_rows = df_results.loc[(df_results['League'] == 'Serie_A')&
                                          (df_results['Year']==2014)]
df_SerieA_top3_2014 = df_SerieA_top3_2014_rows[0:3]
df_SerieA_top3_2015_rows = df_results.loc[(df_results['League'] == 'Serie_A')&
                                          (df_results['Year']==2015)]
df_SerieA_top3_2015 = df_SerieA_top3_2015_rows[0:3]
df_SerieA_top3_2016_rows = df_results.loc[(df_results['League'] == 'Serie_A')&
                                          (df_results['Year']==2016)]
df_SerieA_top3_2016 = df_SerieA_top3_2016_rows[0:3]
df_SerieA_top3_2017_rows = df_results.loc[(df_results['League'] == 'Serie_A')&
                                          (df_results['Year']==2017)]
df_SerieA_top3_2017 = df_SerieA_top3_2017_rows[0:3]
df_SerieA_top3_2018_rows = df_results.loc[(df_results['League'] == 'Serie_A')&
                                          (df_results['Year']==2018)]
df_SerieA_top3_2018 = df_SerieA_top3_2018_rows[0:3]
```

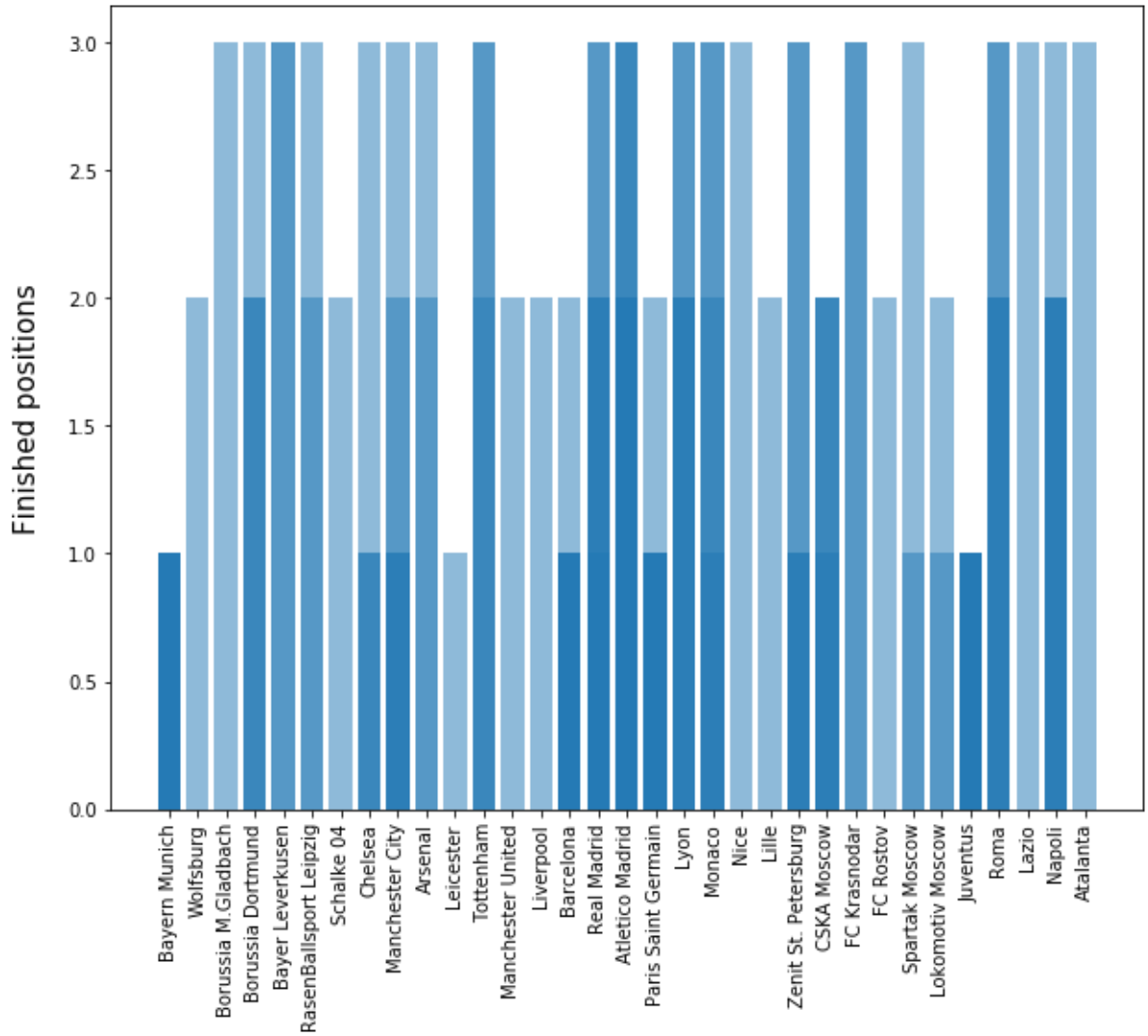
In [16]:
```
#Combining top 3 teams from all years into one DataFrame
frames = [df_bundes_top3_2014, df_bundes_top3_2015, df_bundes_top3_2016, df_bu
ndes_top3_2017, df_bundes_top3_2018,
          df_epl_top3_2014, df_epl_top3_2015, df_epl_top3_2016, df_epl_top3_201
7, df_epl_top3_2018,
          df_laliga_top3_2014,df_laliga_top3_2015, df_laliga_top3_2016, df_lali
ga_top3_2017, df_laliga_top3_2018,
          df_ligue1_top3_2014, df_ligue1_top3_2015, df_ligue1_top3_2016, df_lig
ue1_top3_2017, df_ligue1_top3_2018,
          df_RFPL_top3_2014, df_RFPL_top3_2015, df_RFPL_top3_2016, df_RFPL_top3
_2017, df_RFPL_top3_2018,
          df_SerieA_top3_2014, df_SerieA_top3_2015, df_SerieA_top3_2016, df_Ser
ieA_top3_2017, df_SerieA_top3_2018]

df_leagues_top3 = pd.concat(frames)
```

In [17]:
```
df_leagues_top3.head()
```

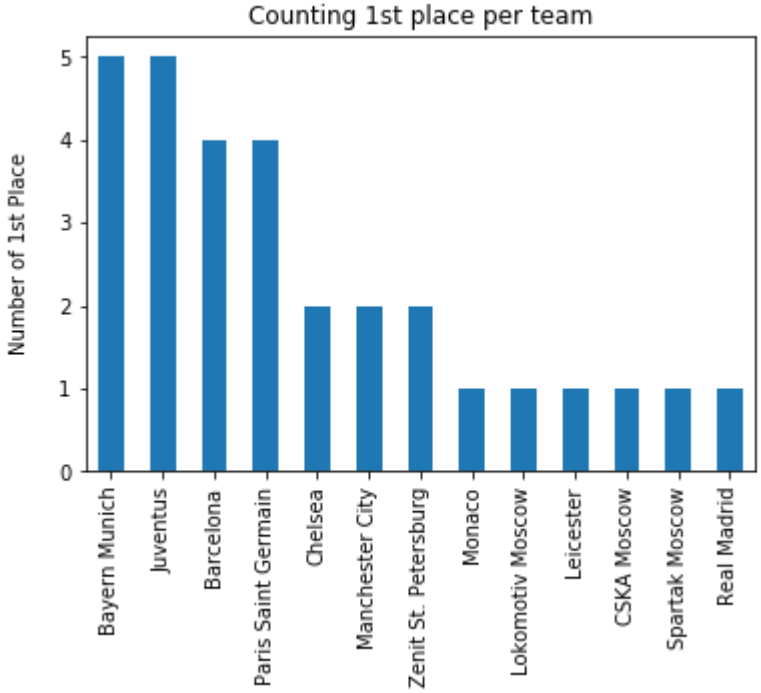
Out[17]:

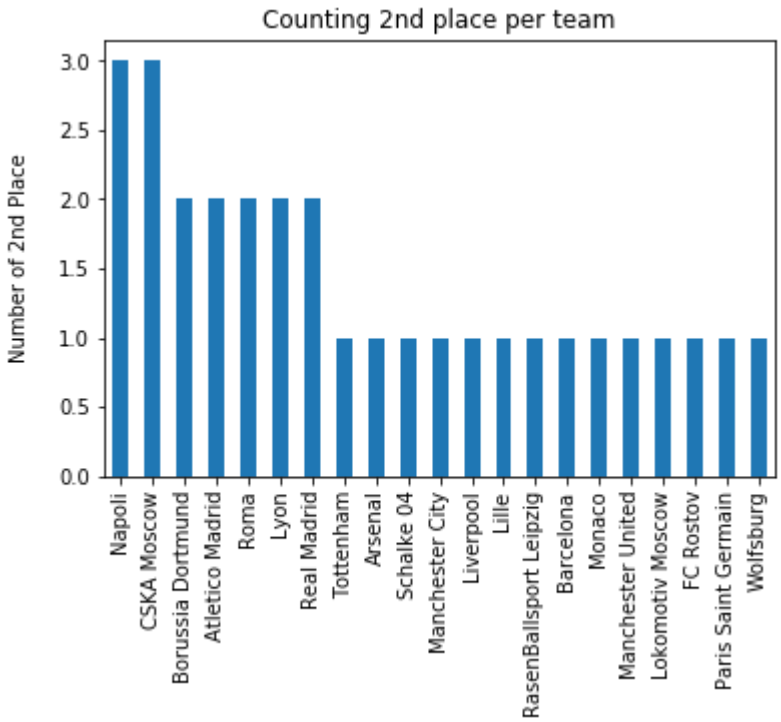
| | League | Year | position | team | matches | wins | draws | loses | scored | missed | pts |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | Bundesliga | 2014 | 1 | Bayern Munich | 34 | 25 | 4 | 5 | 80 | 18 | 79 |
| **1** | Bundesliga | 2014 | 2 | Wolfsburg | 34 | 20 | 9 | 5 | 72 | 38 | 69 |
| **2** | Bundesliga | 2014 | 3 | Borussia M.Gladbach | 34 | 19 | 9 | 6 | 53 | 26 | 66 |
| **18** | Bundesliga | 2015 | 1 | Bayern Munich | 34 | 28 | 4 | 2 | 80 | 17 | 88 |
| **19** | Bundesliga | 2015 | 2 | Borussia Dortmund | 34 | 24 | 6 | 4 | 82 | 34 | 78 |

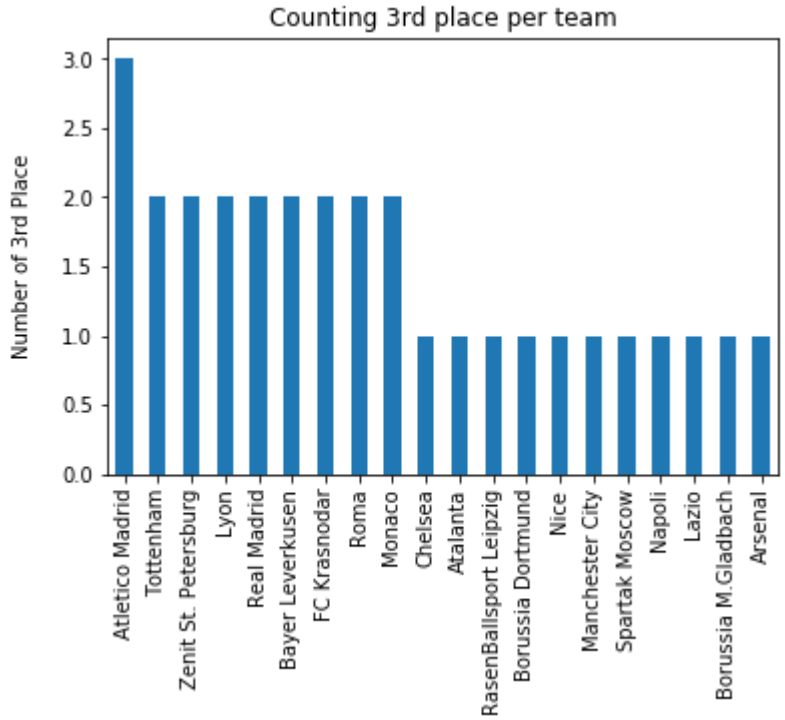
In [18]:
```
#Brief overview of how which teams appear on which position for all years
plt.bar(df_leagues_top3['team'], df_leagues_top3['position'], align='center',
alpha=0.5)
plt.ylabel("Finished positions", labelpad=14, fontsize = 15)
plt.xticks(rotation=90)
plt.gcf().set_size_inches(10, 8)
plt.show()
#plt.legend(df_leagues_top3['position'].value_counts()) doesn't work as wanted
```



In [19]:
```
#Counting the number of times teams have finished on first, second and third p
lace
df_leagues_1 = df_leagues_top3.loc[(df_leagues_top3['position'] == 1)]
df_leagues_1['team'].value_counts().plot(kind = 'bar',rot=90)
plt.ylabel("Number of 1st Place", labelpad=14)
plt.title("Counting 1st place per team")
plt.show()

df_leagues_2 = df_leagues_top3.loc[(df_leagues_top3['position'] == 2)]
df_leagues_2['team'].value_counts().plot(kind = 'bar',rot=90)
plt.ylabel("Number of 2nd Place", labelpad=14)
plt.title("Counting 2nd place per team")
plt.show()

df_leagues_3 = df_leagues_top3.loc[(df_leagues_top3['position'] == 3)]
df_leagues_3['team'].value_counts().plot(kind = 'bar',rot=90)
plt.ylabel("Number of 3rd Place", labelpad=14)
plt.title("Counting 3rd place per team")
plt.show()
```

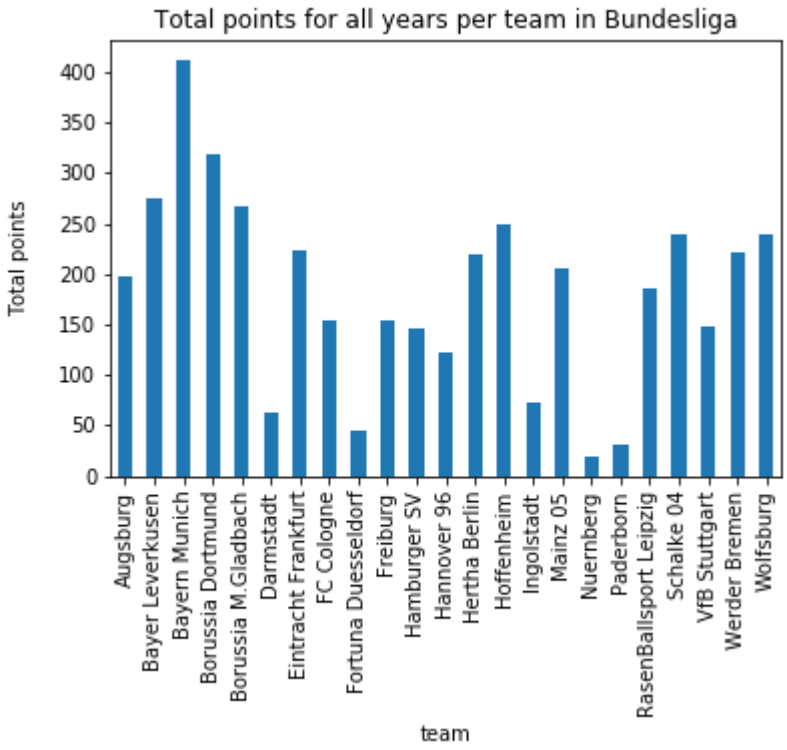

Counting 1st place per team
Number of 1st Place
0
1
2
3
4
5
Bayern Munich
Juventus
Barcelona
Paris Saint Germain
Chelsea
Manchester City
Zenit St. Petersburg
Monaco
Lokomotiv Moscow
Leicester
CSKA Moscow
Spartak Moscow
Real Madrid

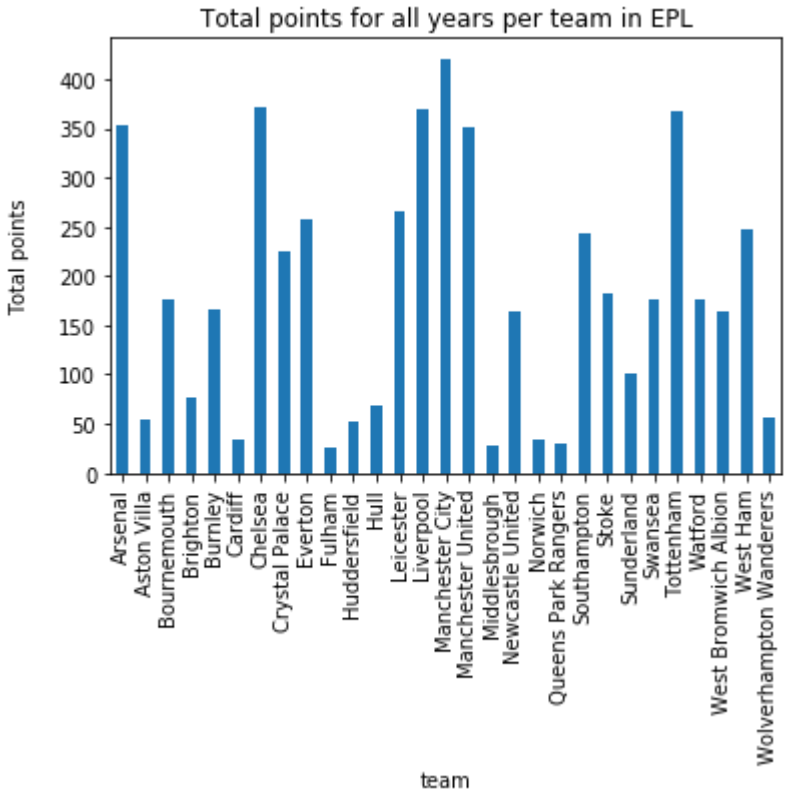

Counting 2nd place per team
Number of 2nd Place
0.0
0.5
1.0
1.5
2.0
2.5
3.0
Napoli
CSKA Moscow
Borussia Dortmund
Atletico Madrid
Roma
Lyon
Real Madrid
Tottenham
Arsenal
Schalke 04
Manchester City
Liverpool
Lille
RasenBallsport Leipzig
Barcelona
Monaco
Manchester United
Lokomotiv Moscow
FC Rostov
Paris Saint Germain
Wolfsburg

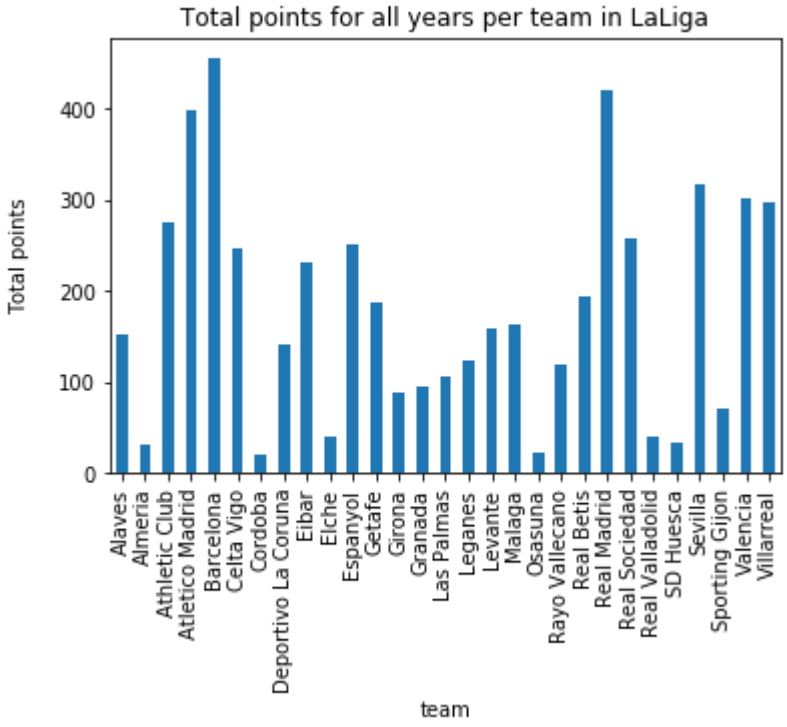


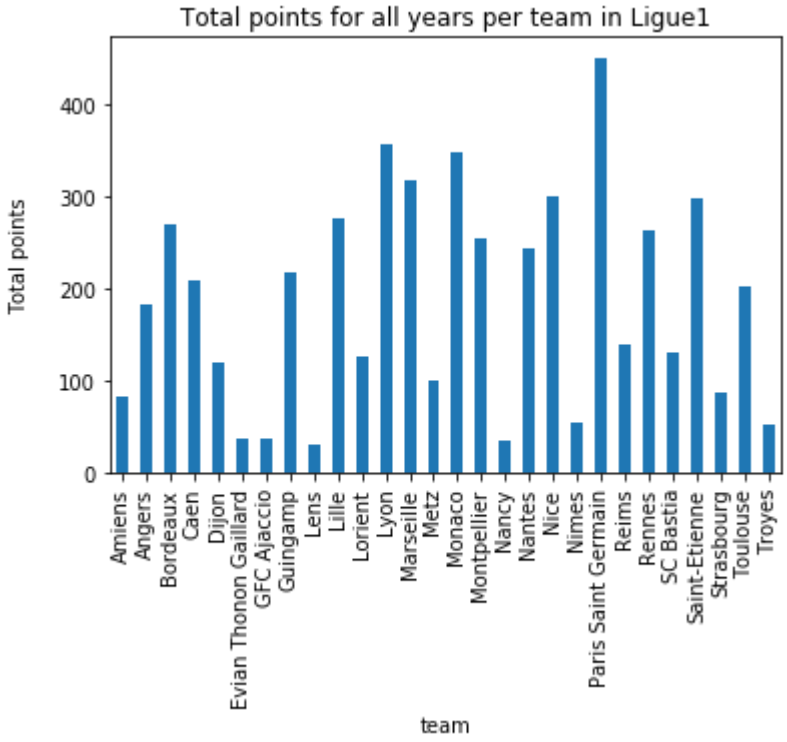

In [20]:
```
#Making datasets for all top3 teams per league
df_bundes_top3_alltime = df_leagues_top3.loc[(df_leagues_top3['League']=='Bundesliga')]
df_epl_top3_alltime = df_leagues_top3.loc[(df_leagues_top3['League']=='EPL')]
df_laliga_top3_alltime = df_leagues_top3.loc[(df_leagues_top3['League']=='La_liga')]
df_ligue1_top3_alltime = df_leagues_top3.loc[(df_leagues_top3['League']=='Ligue_1')]
df_RFPL_top3_alltime = df_leagues_top3.loc[(df_leagues_top3['League']=='RFPL')]
df_SerieA_top3_alltime = df_leagues_top3.loc[(df_leagues_top3['League']=='Serie_A')]

#Making datasets for all teams per league
df_bundes_alltime = df_results.loc[(df_results['League']=='Bundesliga')]
df_epl_alltime = df_results.loc[(df_results['League']=='EPL')]
df_laliga_alltime = df_results.loc[(df_results['League']=='La_liga')]
df_ligue1_alltime = df_results.loc[(df_results['League']=='Ligue_1')]
df_RFPL_alltime = df_results.loc[(df_results['League']=='RFPL')]
df_SerieA_alltime = df_results.loc[(df_results['League']=='Serie_A')]
```

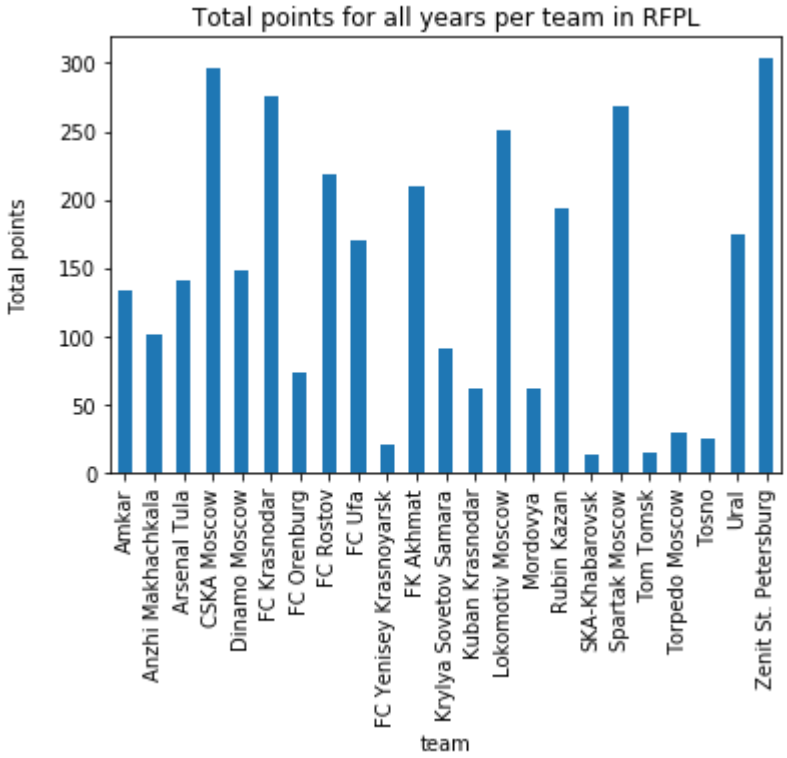
In [21]:
```
#Counting the amount of goals each team has scored through all seasons
df_bundes_alltime['pts'].groupby([df_bundes_alltime['team']]).sum().plot(kind
= 'bar',rot=90)
plt.ylabel("Total points", labelpad=14)
plt.title("Total points for all years per team in Bundesliga")
plt.show()

df_epl_alltime['pts'].groupby([df_epl_alltime['team']]).sum().plot(kind = 'ba
r',rot=90)
plt.ylabel("Total points", labelpad=14)
plt.title("Total points for all years per team in EPL")
plt.show()

df_laliga_alltime['pts'].groupby([df_laliga_alltime['team']]).sum().plot(kind
= 'bar',rot=90)
plt.ylabel("Total points", labelpad=14)
plt.title("Total points for all years per team in LaLiga")
plt.show()

df_ligue1_alltime['pts'].groupby([df_ligue1_alltime['team']]).sum().plot(kind
= 'bar',rot=90)
plt.ylabel("Total points", labelpad=14)
plt.title("Total points for all years per team in Ligue1")
plt.show()

df_RFPL_alltime['pts'].groupby([df_RFPL_alltime['team']]).sum().plot(kind = 'b
ar',rot=90)
plt.ylabel("Total points", labelpad=14)
plt.title("Total points for all years per team in RFPL")
plt.show()

df_SerieA_alltime['pts'].groupby([df_SerieA_alltime['team']]).sum().plot(kind
= 'bar',rot=90)
plt.ylabel("Total points", labelpad=14)
plt.title("Total points for all years per team in Serie A")
plt.show()
```

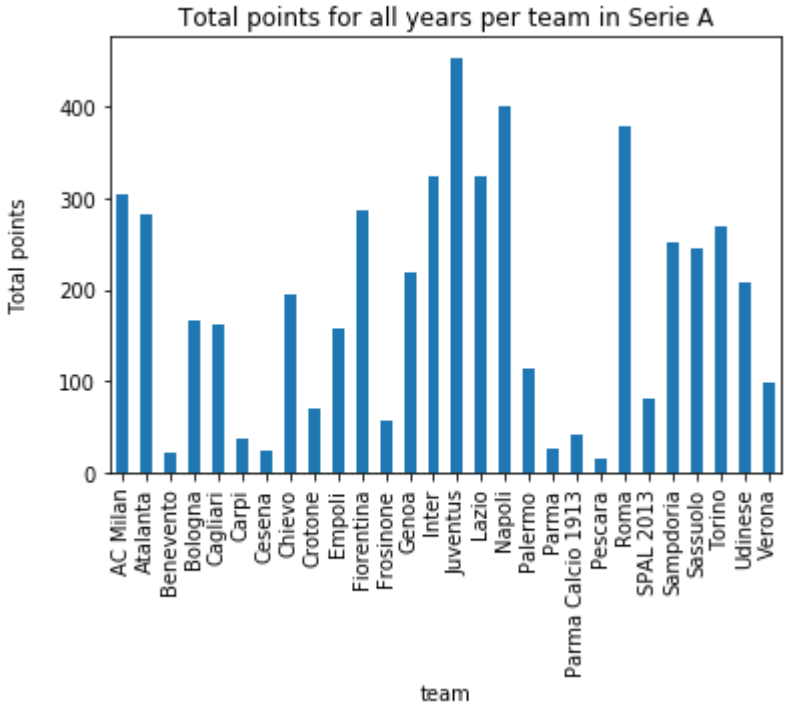

Total points for all years per team in Bundesliga
Total points
0
50
100
150
200
250
300
350
400
Augsburg
Bayer Leverkusen
Bayern Munich
Borussia Dortmund
Borussia M.Gladbach
Darmstadt
Eintracht Frankfurt
FC Cologne
Fortuna Duesseldorf
Freiburg
Hamburger SV
Hannover 96
Hertha Berlin
Hoffenheim
Ingolstadt
Mainz 05
Nuernberg
Paderborn
RasenBallsport Leipzig
Schalke 04
VfB Stuttgart
Werder Bremen
Wolfsburg
team


Total points for all years per team in EPL
Total points
0
50
100
150
200
250
300
350
400
Arsenal
Aston Villa
Bournemouth
Brighton
Burnley
Cardiff
Chelsea
Crystal Palace
Everton
Fulham
Huddersfield
Hull
Leicester
Liverpool
Manchester City
Manchester United
Middlesbrough
Newcastle United
Norwich
Queens Park Rangers
Southampton
Stoke
Sunderland
Swansea
Tottenham
Watford
West Bromwich Albion
West Ham
Wolverhampton Wanderers
team


Total points for all years per team in LaLiga
Total points
0
100
200
300
400
Alaves
Almeria
Athletic Club
Atletico Madrid
Barcelona
Celta Vigo
Cordoba
Deportivo La Coruna
Eibar
Elche
Espanyol
Getafe
Girona
Granada
Las Palmas
Leganes
Levante
Malaga
Osasuna
Rayo Vallecano
Real Betis
Real Madrid
Real Sociedad
Real Valladolid
SD Huesca
Sevilla
Sporting Gijon
Valencia
Villarreal
team


Total points for all years per team in Ligue1
Total points
0
100
200
300
400
Amiens
Angers
Bordeaux
Caen
Dijon
Evian Thonon Gaillard
GFC Ajaccio
Guingamp
Lens
Lille
Lorient
Lyon
Marseille
Metz
Monaco
Montpellier
Nancy
Nantes
Nice
Nimes
Paris Saint Germain
Reims
Rennes
SC Bastia
Saint-Etienne
Strasbourg
Toulouse
Troyes
team


Total points for all years per team in RFPL
Total points
team
Amkar
Anzhi Makhachkala
Arsenal Tula
CSKA Moscow
Dinamo Moscow
FC Krasnodar
FC Orenburg
FC Rostov
FC Ufa
FC Yenisey Krasnoyarsk
FK Akhmat
Krylya Sovetov Samara
Kuban Krasnodar
Lokomotiv Moscow
Mordovya
Rubin Kazan
SKA-Khabarovsk
Spartak Moscow
Tom Tomsk
Torpedo Moscow
Tosno
Ural
Zenit St. Petersburg


Total points for all years per team in Serie A
Total points
team
AC Milan
Atalanta
Benevento
Bologna
Cagliari
Carpi
Cesena
Chievo
Crotone
Empoli
Fiorentina
Frosinone
Genoa
Inter
Juventus
Lazio
Napoli
Palermo
Parma
Parma Calcio 1913
Pescara
Roma
SPAL 2013
Sampdoria
Sassuolo
Torino
Udinese
Verona

In [22]:
```
df_bundes_top3_alltime.head(20)
```

Out[22]:

| | League | Year | position | team | matches | wins | draws | loses | scored | missed | pts |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | Bundesliga | 2014 | 1 | Bayern Munich | 34 | 25 | 4 | 5 | 80 | 18 | 7 |
| **1** | Bundesliga | 2014 | 2 | Wolfsburg | 34 | 20 | 9 | 5 | 72 | 38 | 6 |
| **2** | Bundesliga | 2014 | 3 | Borussia M.Gladbach | 34 | 19 | 9 | 6 | 53 | 26 | 6 |
| **18** | Bundesliga | 2015 | 1 | Bayern Munich | 34 | 28 | 4 | 2 | 80 | 17 | 8 |
| **19** | Bundesliga | 2015 | 2 | Borussia Dortmund | 34 | 24 | 6 | 4 | 82 | 34 | 7 |
| **20** | Bundesliga | 2015 | 3 | Bayer Leverkusen | 34 | 18 | 6 | 10 | 56 | 40 | 6 |
| **36** | Bundesliga | 2016 | 1 | Bayern Munich | 34 | 25 | 7 | 2 | 89 | 22 | 8 |
| **37** | Bundesliga | 2016 | 2 | RasenBallsport Leipzig | 34 | 20 | 7 | 7 | 66 | 39 | 6 |
| **38** | Bundesliga | 2016 | 3 | Borussia Dortmund | 34 | 18 | 10 | 6 | 72 | 40 | 6 |
| **54** | Bundesliga | 2017 | 1 | Bayern Munich | 34 | 27 | 3 | 4 | 92 | 28 | 8 |
| **55** | Bundesliga | 2017 | 2 | Schalke 04 | 34 | 18 | 9 | 7 | 53 | 37 | 6 |
| **56** | Bundesliga | 2017 | 3 | Bayer Leverkusen | 34 | 15 | 10 | 9 | 58 | 44 | 5 |
| **72** | Bundesliga | 2018 | 1 | Bayern Munich | 34 | 24 | 6 | 4 | 88 | 32 | 7 |
| **73** | Bundesliga | 2018 | 2 | Borussia Dortmund | 34 | 23 | 7 | 4 | 81 | 44 | 7 |
| **74** | Bundesliga | 2018 | 3 | RasenBallsport Leipzig | 34 | 19 | 9 | 6 | 63 | 29 | 6 |

In [23]:
```
features_goals = df_results[['Year','position','matches', 'wins','draws','loses','missed','pts']]
features_position = df_results[['Year','matches', 'wins','draws','loses','scored','missed','pts']]
```

In [24]:
```
#SelectKBest for goals
from sklearn.feature_selection import SelectKBest, chi2, f_regression, f_classif
column_goals = SelectKBest(score_func=f_classif,k=5).fit_transform(features_goals,df_results['scored'])
print(column_goals)
```

```
[[ 1 34 25  5 79]
 [ 2 34 20  5 69]
 [ 3 34 19  6 66]
 ...
 [18 38 10 20 38]
 [19 38  5 23 25]
 [20 38  2 22 20]]
```

In [25]:
```
#SelectKBest for position
from sklearn.feature_selection import SelectKBest, chi2, f_regression, f_class
if
column_position = SelectKBest(score_func=f_classif,k=5).fit_transform(features
_position,df_results['position'])
print(column_position)
```

```
[[25  5 80 18 79]
 [20  5 72 38 69]
 [19  6 53 26 66]
 ...
 [10 20 51 70 38]
 [ 5 23 29 69 25]
 [ 2 22 25 75 20]]
```

In [26]:
```
#Defining the columns(features) to use for training the algorithm and which co
lumn
#I want to predict(X is for features and Y is for the predicted column)
X_goals = df_results[['wins','loses','pts','missed','draws']]
y_goals = df_results['scored']
X_position = df_results[['wins','loses','scored','missed','pts']]
y_position = df_results['position']
```

In [27]:
```
#Splitting the data for predicting goals into test and train sets
X_train_goals, X_test_goals, y_train_goals, y_test_goals = train_test_split(X_
goals, y_goals, test_size=0.20)
#Splitting the data for predicting position into test and train sets
X_train_position, X_test_position, y_train_position, y_test_position = train_t
est_split(X_position, y_position, test_size=0.20)
```

In [28]:
```
#KNN algorithm for predicting goals
knn = KNeighborsClassifier(n_neighbors=5)
knn.fit(X_train_goals, y_train_goals)
pred = knn.predict(X_test_goals)
accuracy = metrics.accuracy_score(y_test_goals, pred)
print(accuracy)
print(pred)
```

```
0.06140350877192982
[60 44 90 69 45 36 32 47 13 44 72 34 28 49 63 31 34 35 72 33 61 75 57 33
 28 20 40 22 41 29 28 41 42 36 42 20 32 71 39 29 41 59 17 44 18 40 30 55
 36 18 48 49 27 20 36 50 34 40 46 45 47 47 42 34 68 41 32 47 30 63 42 44
 58 40 33 20 42 55 59 56 42 28 56 68 32 42 63 41 48 70 54 33 35 62 36 75
 34 22 20 52 35 39 71 44 44 70 24 35 45 24 47 44 28 25]
```

In [29]:
```
#SVM algorithm for predicting position
clf_position=svm.SVC(kernel='linear',C=1).fit(X_train_position,y_train_positio
n)
predict = clf_position.predict(X_test_position)
score_goals=accuracy_score(y_test_position,predict)
print(score_goals)
print(predict)
```

```
0.34210526315789475
[14 17  2  6  5 15  6  9 15  7 15 17 20 13 19 10  6  2 16 15 20  9  9  8
 11 19 18  9 15 19 15  9  1 13  2 10 14 13  9 10 11  3 15 15 18 11 15  6
  6  8 16  3 13  6 11  1  7 19  2 15 19  5 15 15 16 11 10 16  9 10  3  4
  3  7 13 12 11 15  4 16 11 17 20  7 11  5 13  5  6  4 12 16 18  8  4  3
 15  8  1  9  5  9 11 15 16 13  9 13  3 10  2  8  9  7]
```

In [30]:
```
#Testing SVM for predicting position on smaller DataFrame(Top 3 teams from Bun
desliga for all time)
X_position2 = df_bundes_top3_alltime[['wins','loses','scored','missed','pts']]
y_position2 = df_bundes_top3_alltime['position']
X_train_position2, X_test_position2, y_train_position2, y_test_position2 = tra
in_test_split(X_position2, y_position2, test_size=0.20)
predict2 = clf_position.predict(X_test_position2)
score_goals2=accuracy_score(y_test_position2,predict2)
print(score_goals2)
print(predict2)
```

```
1.0
[2 1 2]
```

In [31]:
```
#Testing KNN for predicting goals on smaller DataFrame(Top 3 teams from Bundes
liga for all time)
X_goals2 = df_bundes_top3_alltime[['wins','loses','pts','missed','draws']]
y_goals2 = df_bundes_top3_alltime['scored']
X_train_goals2, X_test_goals2, y_train_goals2, y_test_goals2 = train_test_spli
t(X_goals2, y_goals2, test_size=0.20)
knn2 = KNeighborsClassifier(n_neighbors=5)
knn2.fit(X_train_goals2, y_train_goals2)
pred2 = knn2.predict(X_test_goals2)
accuracy2 = metrics.accuracy_score(y_test_goals2, pred2)
print(accuracy2)
print(pred2)
```

```
0.0
[53 53 53]
```