P2

October 27, 2022

[7]:
```python
import pandas, sqlite3, matplotlib.pyplot as plt, numpy as np
```

[8]:
```python
### Problem 1

# Connect to the database and create a datafram from the query below.
# More infor on query below.

sqlite_file = 'lahman2014.sqlite'
conn = sqlite3.connect(sqlite_file)

query = """with pay as
           (select yearid, teamid, sum(salary) as total_payroll from salaries␣
 ↪group by yearid, teamid),
           wr as
           (select yearid, teamid, franchid, w, g, (cast(w as real) / g) * 100␣
 ↪as win_rate from teams)
           select * from pay natural join wr;"""
res = pandas.read_sql(query, conn)
res
```

[8]:
```
     yearid teamid  total_payroll franchid   w    g   win_rate
0      1985    ATL     14807000.0      ATL  66  162  40.740741
1      1985    BAL     11560712.0      BAL  83  161  51.552795
2      1985    BOS     10897560.0      BOS  81  163  49.693252
3      1985    CAL     14427894.0      ANA  90  162  55.555556
4      1985    CHA      9846178.0      CHW  85  163  52.147239
..      …      …              …        …  ..  …          …
853    2014    SLN    120693000.0      STL  90  162  55.555556
854    2014    TBA     72689100.0      TBD  77  162  47.530864
855    2014    TEX    112255059.0      TEX  67  162  41.358025
856    2014    TOR    109920100.0      TOR  83  162  51.234568
857    2014    WAS    131983680.0      WSN  96  162  59.259259

[858 rows x 7 columns]
```

I used the following query to get the total payroll for each team for each year:

`select yearid, teamid, sum(salary) as total_payroll`

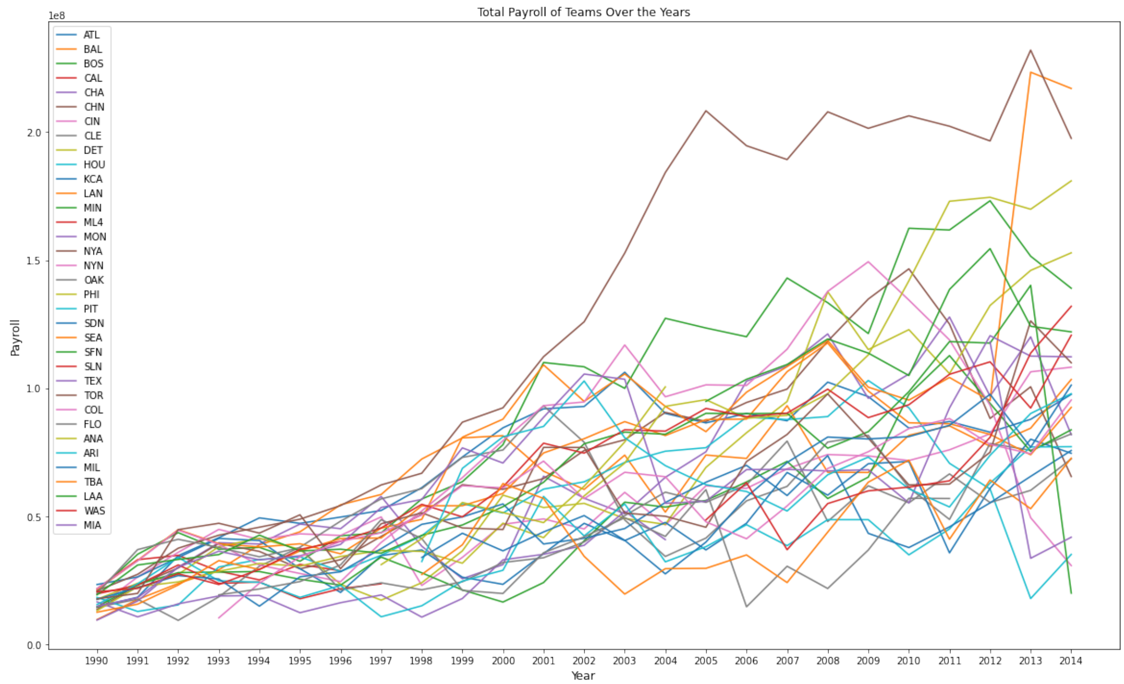
```
from salaries
group by yearid, teamid;
```

I used this second query to get the winrate for each team for each year recorded in the Teams relation:

```
select yearid, teamid, franchid, w, g, (cast(w as real) / g) * 100 as win_rate
from teams;
```

Looking at the resulting number of rows from the two queries, it was apparent that the salaries relation was missing payroll data for some teams for some years. To deal with the missing data, I chose to use a natural join which would merge the data for team and year combination that existed in both tables.

[9]:
```
### Problem 2

# Filtered for all data between [1990, 2014]
# For each team in the database, plotted their total spendings over the years␣
↪in a line graph.
# Set the team names (TeamID) as the labels for the legend.

filtered = res.loc[res['yearid'] >= 1990]

plt.figure(figsize = (20, 12))
plt.title('Total Payroll of Teams Over the Years')
plt.xlabel('Year', size=12)
plt.ylabel('Payroll', size=12)
plt.xticks(np.arange(1990, 2015))

for teams in filtered['teamid'].unique():
  team_data = filtered.loc[filtered['teamid'] == teams]
  plt.plot(team_data['yearid'], team_data['total_payroll'], label = teams)

plt.legend()
plt.show()
```

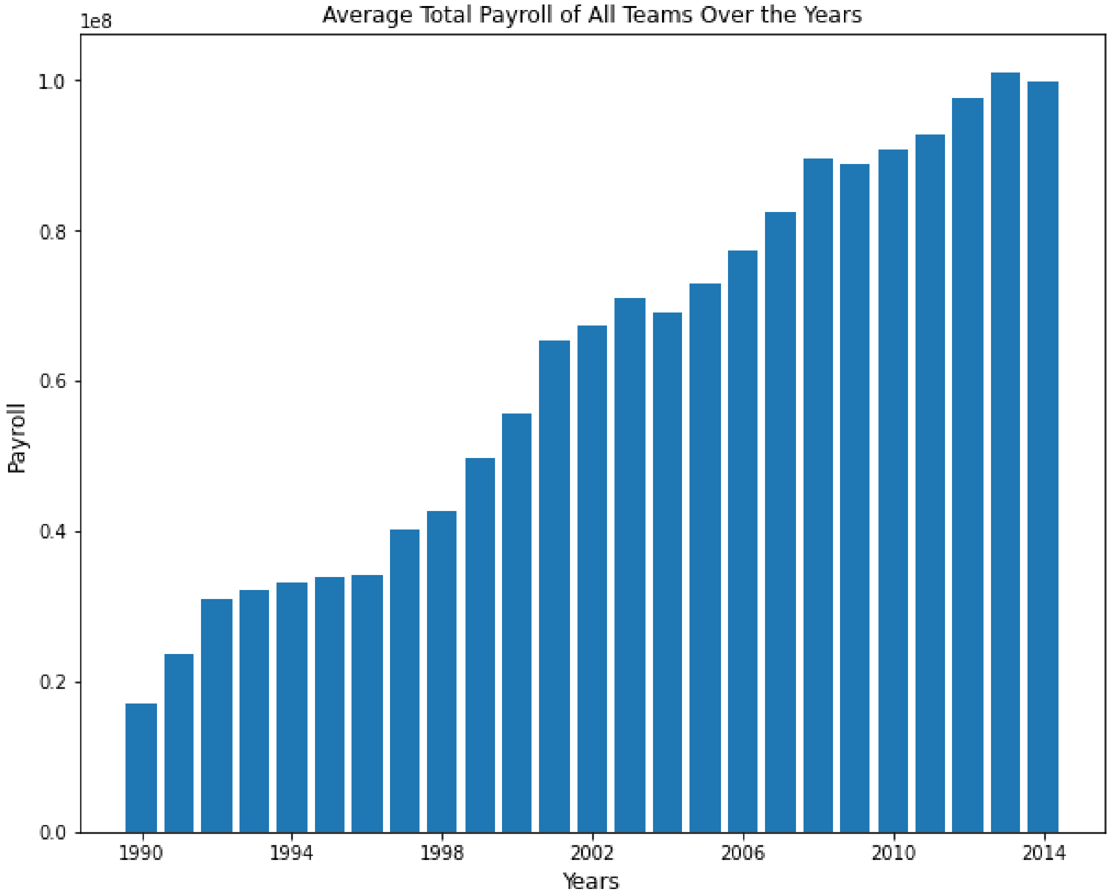


0.0.1 Question 1

From this line graph we can clearly see that there is an upward trend meaning that for the most part each team saw an increase in payroll over time. However, we can also see that some teams that saw an increase in payroll in earlier years are now seeing a decrease in payrolls in recent years. The spread of the payrolls between teams also saw an increase over the years. We see that in the beginning, at 1990, all teams had very similar total payrolls. By 2014, there is signficant difference between the teams with the highest and lowest payrolls. This graph can additionally be used analyze the average total payrolls for all teams over time.

[ ]:
```
### Problem 3

# Got the unique list of years in the database and found the average payroll␣
  ↪for all teams for each year.
# Plotted the data to get a bar graph of average total payroll over the years.

trend_x = filtered['yearid'].unique()
trend_y = filtered.groupby(['yearid'])['total_payroll'].mean().values

plt.figure(figsize = (10, 8))
plt.title('Average Total Payroll of All Teams Over the Years')
plt.xlabel('Years', size=12)
plt.ylabel('Payroll', size=12)
plt.xticks(np.arange(1990, 2015, 4))
```

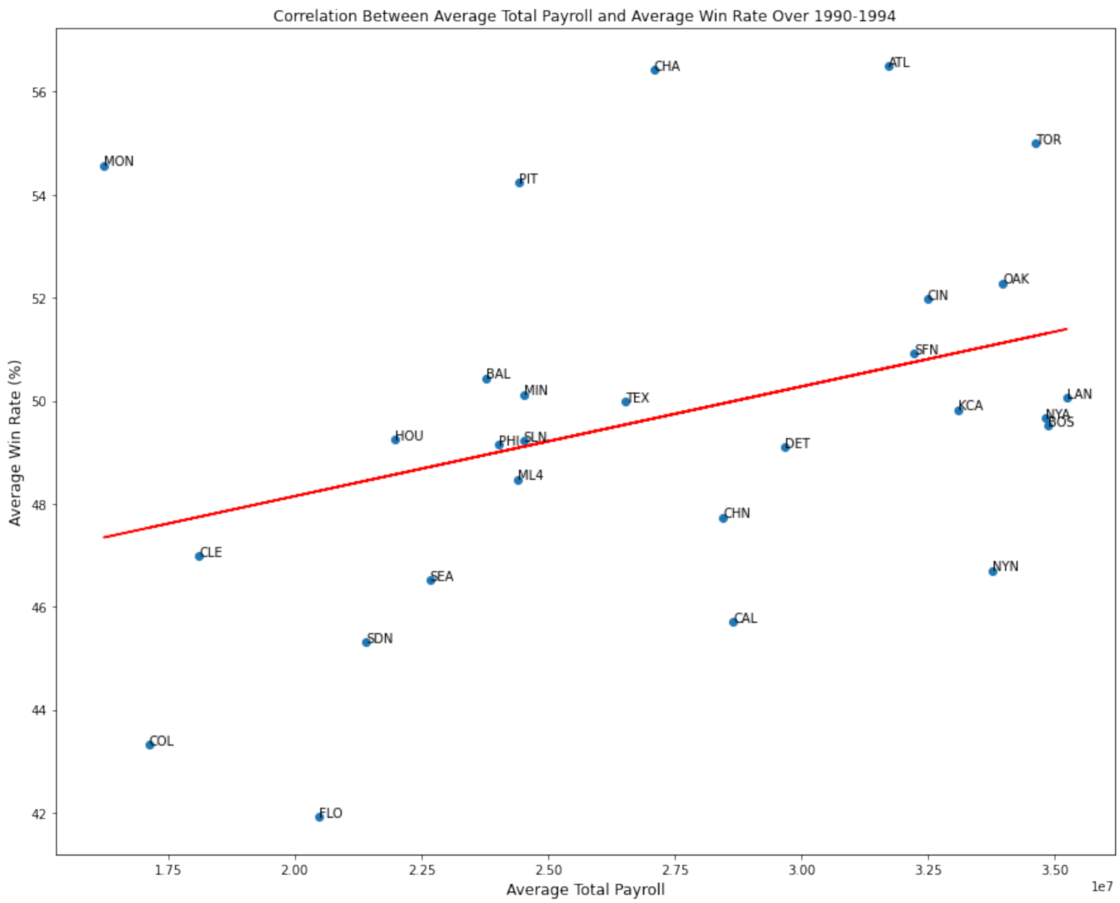
```
plt.bar(trend_x, trend_y)
plt.show()
```

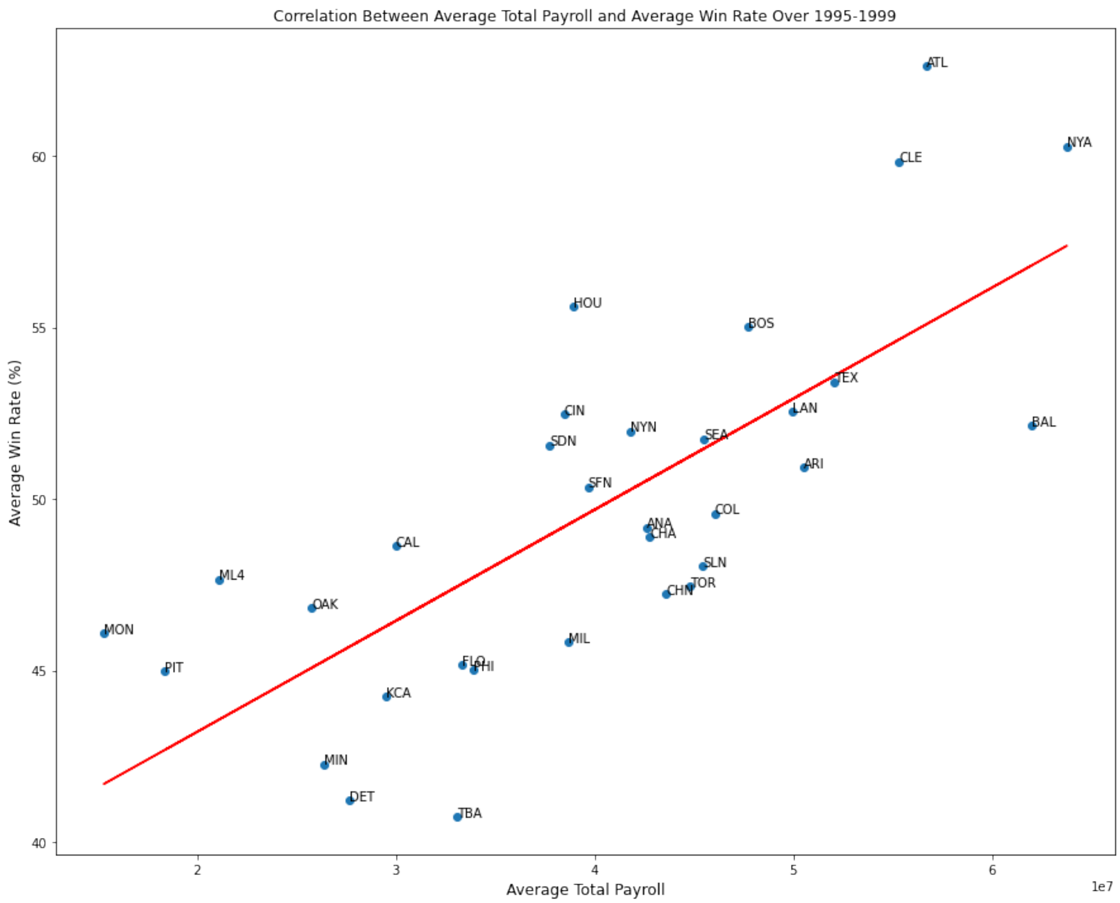


Generated a bar graph to show that on average, all teams saw continous increase in total payroll over the years.*italicized text*

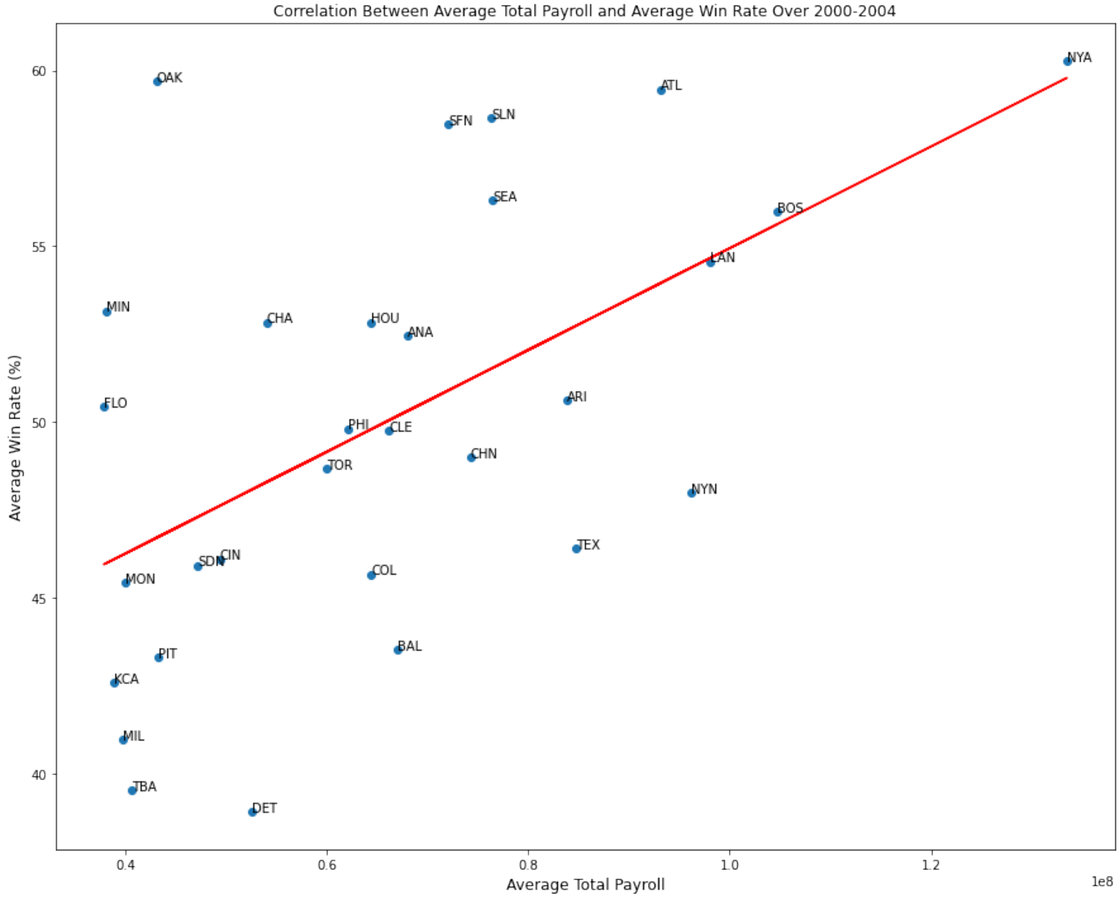
[10]:
```
### Problem 4

# Divided the years from 1990-2014 into 5 groups with each starting year as␣
↪1990, 1995, 2000, 2005, 2010
# Incremented by 5s to not double count the ending year for periods (ie,␣
↪1990-1994, 1994-1998)
# Separated the dataframe using pandas.cut() into 5 bins by their yearid and␣
↪labeled rows in each bin by the starting year of their period
# For each period, found the average total payroll and win rate for each team␣
↪during the period
# Plotted the data using a scatterplot and included a regression line found␣
↪using numpy.ployfit()
```

```
# For each scatterplot, iterated through each point and annotated the team name

ptable = filtered.copy()
periods = np.arange(1990, 2014, 5)
ptable['period'] = pandas.cut(x=ptable['yearid'], bins=5, labels=periods)

for period in periods:
  temp = ptable.loc[ptable['period'] == period]
  avg_pay = temp.groupby(['teamid'])['total_payroll'].mean().to_frame()
  avg_wr = temp.groupby(['teamid'])['win_rate'].mean().to_frame()

  avg = avg_pay.merge(avg_wr, how='inner', on='teamid')
  avg['team'] = avg.index

  plt.figure(figsize=(15,12))
  title = "Correlation Between Average Total Payroll and Average Win Rate Over␣
 ↪{}-{}".format(period, period + 4)
  plt.title(title)
  plt.xlabel('Average Total Payroll', size=12)
  plt.ylabel('Average Win Rate (%)', size=12)
  plt.scatter(avg['total_payroll'], avg['win_rate'])

  x = avg['total_payroll'].values
  y = avg['win_rate'].values
  a, b = np.polyfit(x, y, 1)
  plt.plot(x, a * x + b, "r-")

  for k, v in enumerate(avg['team']):
    plt.annotate(v, (avg['total_payroll'][k], avg['win_rate'][k]), size = 10)

  plt.show()
```

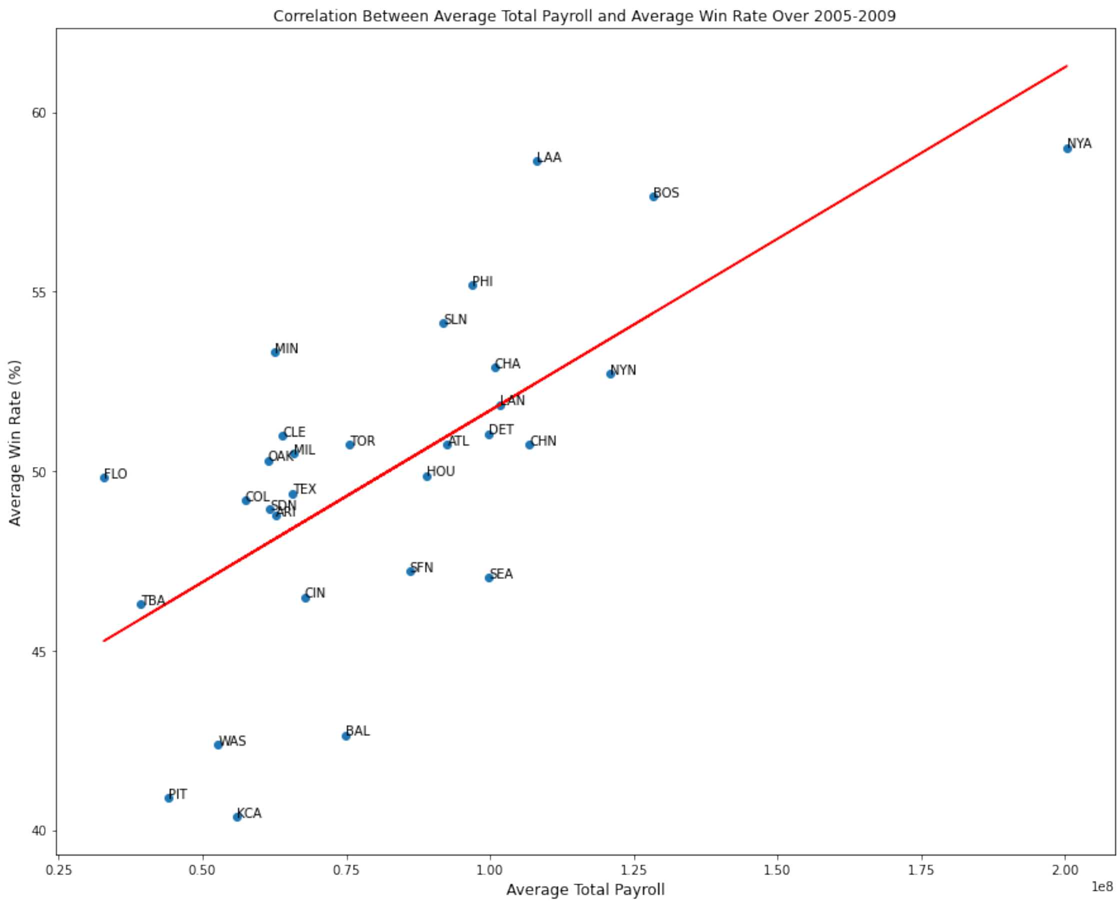

Correlation Between Average Total Payroll and Average Win Rate Over 1990-1994
Average Win Rate (%)
Average Total Payroll
1e7
56
54
52
50
48
46
44
42
1.75
2.00
2.25
2.50
2.75
3.00
3.25
3.50
CHA
ATL
TOR
MON
PIT
OAK
CIN
SFN
BAL
MIN
TEX
LAN
KCA
NYA
BOS
HOU
PHI
SLN
DET
ML4
CHN
CLE
NYN
SEA
CAL
SDN
COL
FLO



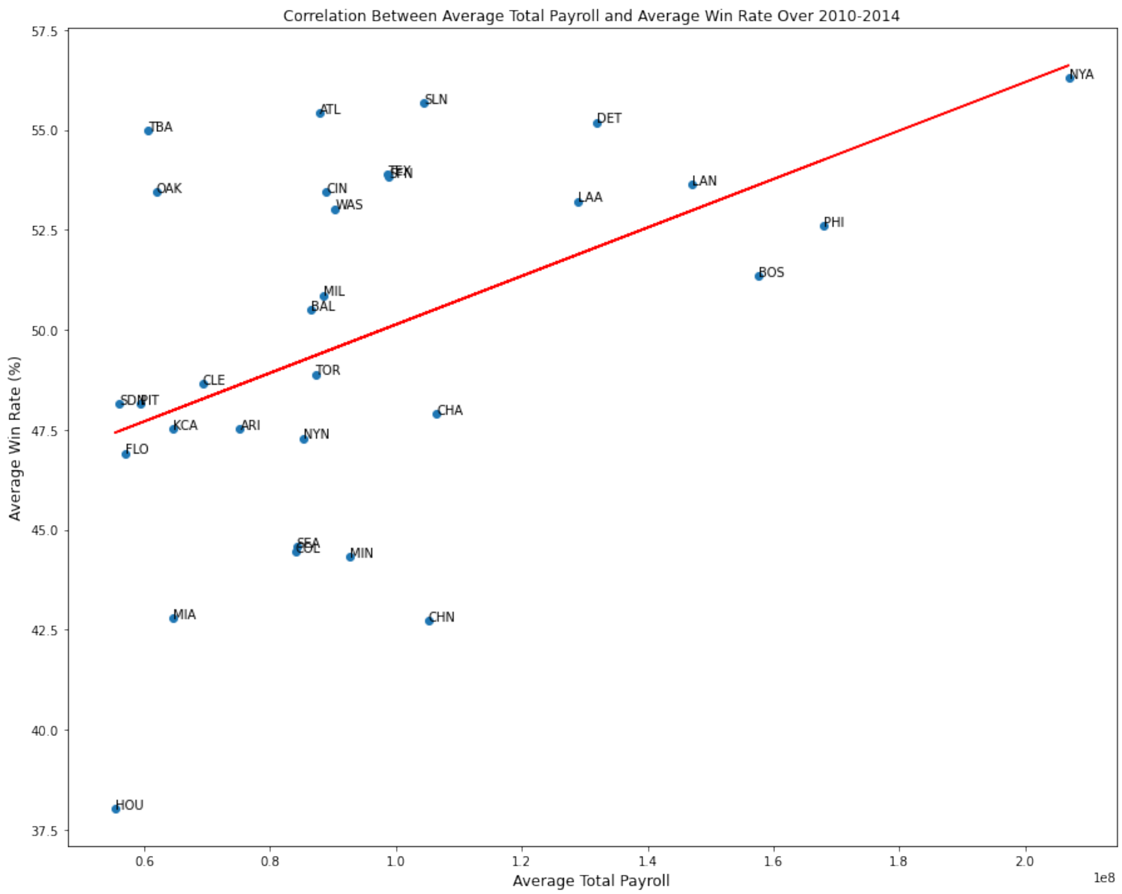
Correlation Between Average Total Payroll and Average Win Rate Over 1995-1999
Average Win Rate (%)
Average Total Payroll
1e7
40
45
50
55
60
2
3
4
5
6
ATL
NYA
CLE
HOU
BOS
TEX
LAN
BAL
CIN
NYN
SEA
SDN
ARI
SFN
COL
ANA
CHA
CAL
SLN
ML4
TOR
CHN
OAK
MON
MIL
FLO
PHI
PIT
KCA
MIN
DET
TBA



Correlation Between Average Total Payroll and Average Win Rate Over 2000-2004
Average Win Rate (%)
Average Total Payroll
1e8
60
55
50
45
40
0.4
0.6
0.8
1.0
1.2
NYA
OAK
ATL
SLN
SFN
SEA
BOS
LAN
MIN
CHA
HOU
ANA
ARI
FLO
PHI
CLE
CHN
TOR
NYN
TEX
SDN
CIN
MON
COL
BAL
PIT
KCA
MIL
TBA
DET



Correlation Between Average Total Payroll and Average Win Rate Over 2005-2009
Average Win Rate (%)
Average Total Payroll
1e8
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
40
45
50
55
60
NYA
LAA
BOS
PHI
SLN
MIN
CHA
NYN
LAN
CLE
TOR
DET
ATL
CHN
MIL
OAK
FLO
HOU
TEX
COL
SDN
ARI
SFN
SEA
CIN
TBA
BAL
WAS
PIT
KCA



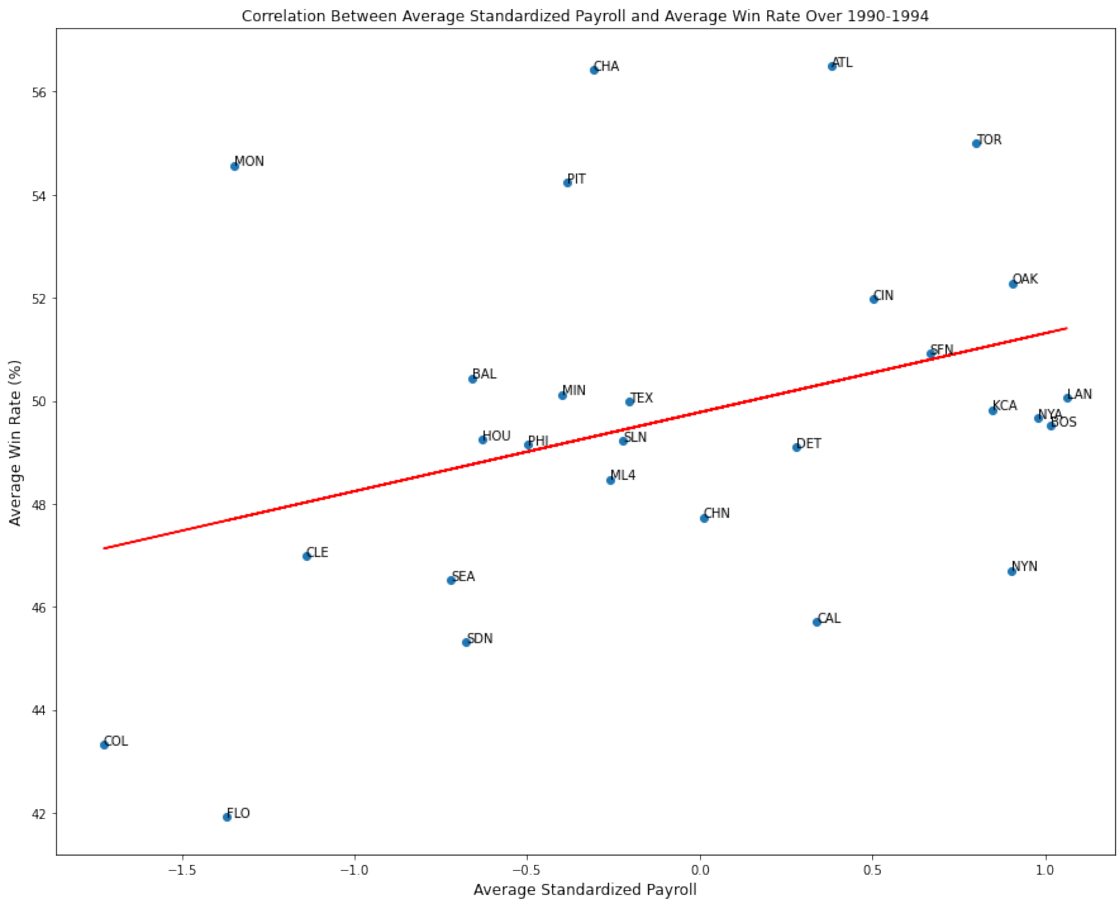
0.0.2 Question 2

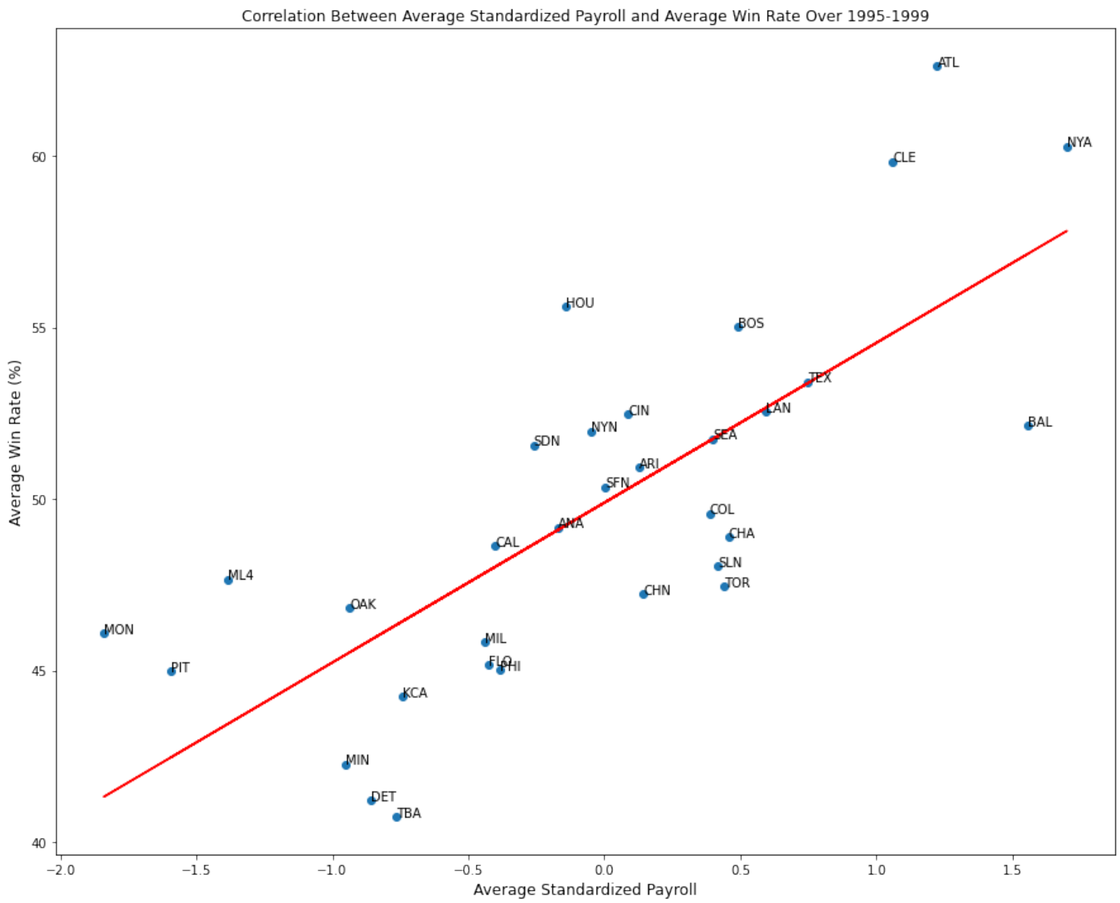
Refering to the graphs generated for each period, it is apparent by looking at the regression line that there is a positive correlation between the average total payroll and the average win rate. In other words, generally, the more money a team spends, the higher their win rate. Two teams that really stood out in these graphs were NYA and ATL. Throughout the periods, it is apparent that NYA has consistently spent the most out of any other team and as a result they maintained one of the highest win rates through the years (excluding 1990-1994). ATL on the other hand was a team that initially had high spendings like NYA but over the years has successfully managed to maintain a high win rate while cutting back on their spendings. Taking a look at OAK, it appears that they took a similar approach to ATL. In the 1990-1994 period, OAK spent as much as some of the highest spending teams and saw a relatively high win rate. But over the years, they would cut back on their spendings while maintaining win rates higher than the average win rate at their spending range (indicated by the regression line). While their win rates fluctuated throughout the years dropping significantly during the 1995-1999 and 2005-2009 periods, they generally had a relatively high spending efficiency. The team's spending to win rate ratio peaked in the 2000-2004 period when it achieved one of the highest win rates out of all teams while spending than most other teams.

[11]:
```python
### Problem 5

# Found the mean and standard deviation for all teams' payroll by each year
# Applied the given formula to add the standardized payroll column

standardized = filtered.copy()
standardized['avg_pay'] = standardized['total_payroll'].
 ↪groupby(standardized['yearid']).transform('mean')
standardized['std'] = standardized['total_payroll'].
 ↪groupby(standardized['yearid']).transform('std')

def get_stdpay(pay, meanpay, std):
  return (pay - meanpay) / std

standardized['stdpay'] = standardized.apply(lambda x :␣
 ↪get_stdpay(x['total_payroll'], x['avg_pay'], x['std']), axis=1)

standardized
```

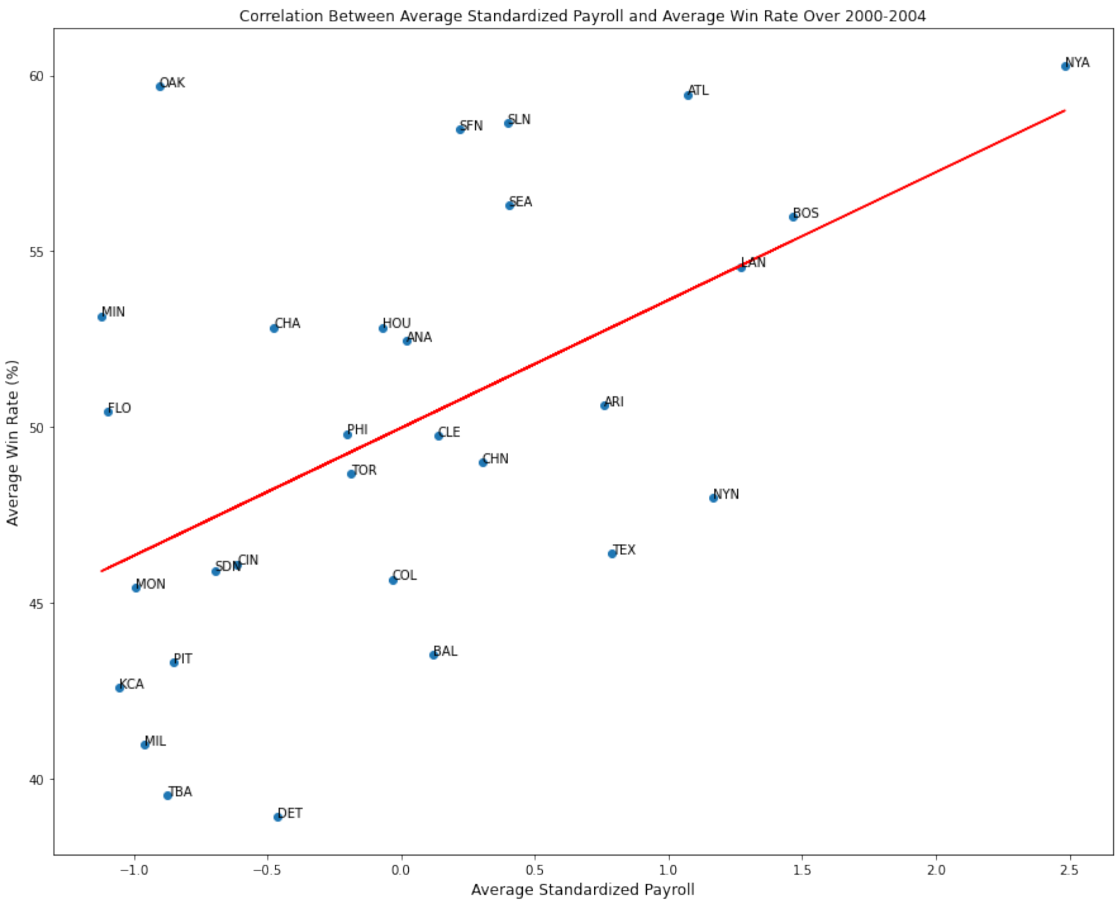
[11]:
```
     yearid teamid  total_payroll franchid   w    g   win_rate       avg_pay  \
130    1990    ATL     14555501.0      ATL  65  162  40.123457  1.707235e+07
131    1990    BAL      9680084.0      BAL  76  161  47.204969  1.707235e+07
132    1990    BOS     20558333.0      BOS  88  162  54.320988  1.707235e+07
133    1990    CAL     21720000.0      ANA  80  162  49.382716  1.707235e+07
134    1990    CHA      9491500.0      CHW  94  162  58.024691  1.707235e+07
..      ...    ...            ...      ...  ..  ...        ...           ...
853    2014    SLN    120693000.0      STL  90  162  55.555556  9.980002e+07
854    2014    TBA     72689100.0      TBD  77  162  47.530864  9.980002e+07
855    2014    TEX    112255059.0      TEX  67  162  41.358025  9.980002e+07
856    2014    TOR    109920100.0      TOR  83  162  51.234568  9.980002e+07
857    2014    WAS    131983680.0      WSN  96  162  59.259259  9.980002e+07

              std    stdpay
130  3.771834e+06 -0.667275
131  3.771834e+06 -1.959861
132  3.771834e+06  0.924213
133  3.771834e+06  1.232198
134  3.771834e+06 -2.009859
..            ...       ...
853  4.570505e+07  0.457126
854  4.570505e+07 -0.593171
855  4.570505e+07  0.272509
856  4.570505e+07  0.221422
857  4.570505e+07  0.704160

[728 rows x 10 columns]
```

```
[ ]: ### Problem 6

     # Essentially did the thing mentioned in Problem 4's comments except this time␣
      ↪found the average standardized payroll for each team
     # and used that data instead of average total payroll.

     ptable2 = standardized.drop(['avg_pay', 'std'], axis=1).copy()
     ptable2['period'] = pandas.cut(x=ptable['yearid'], bins=5, labels=periods)

     for period in periods:
       temp = ptable2.loc[ptable2['period'] == period]
       stdpay = temp.groupby(['teamid'])['stdpay'].mean().to_frame()
       avg_wr = temp.groupby(['teamid'])['win_rate'].mean().to_frame()

       avg = avg_wr.merge(stdpay, how='inner', on='teamid')
       avg['team'] = avg.index

       plt.figure(figsize=(15,12))
       title = "Correlation Between Average Standardized Payroll and Average Win␣
      ↪Rate Over {}-{}".format(period, period + 4)
       plt.title(title)
       plt.xlabel('Average Standardized Payroll', size=12)
       plt.ylabel('Average Win Rate (%)', size=12)
       plt.scatter(avg['stdpay'], avg['win_rate'])

       x = avg['stdpay'].values
       y = avg['win_rate'].values
       a, b = np.polyfit(x, y, 1)
       plt.plot(x, a * x + b, "r-")

       for k, v in enumerate(avg['team']):
         plt.annotate(v, (avg['stdpay'][k], avg['win_rate'][k]), size = 10)

       plt.show()
```

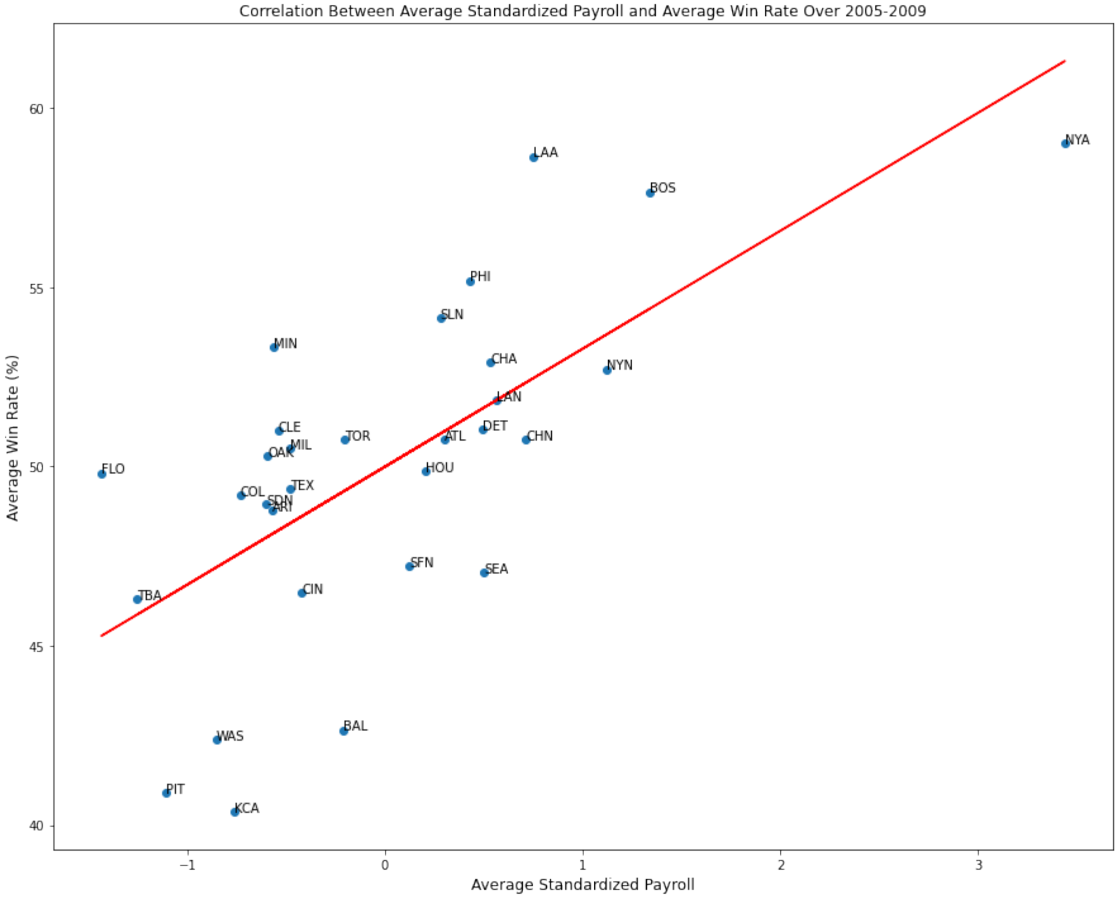

Correlation Between Average Standardized Payroll and Average Win Rate Over 1990-1994
Average Win Rate (%)
Average Standardized Payroll
CHA
ATL
TOR
MON
PIT
OAK
CIN
SFN
BAL
MIN
TEX
LAN
KCA
NYA
BOS
HOU
PHI
SLN
DET
ML4
CHN
CLE
SEA
NYN
CAL
SDN
COL
FLO



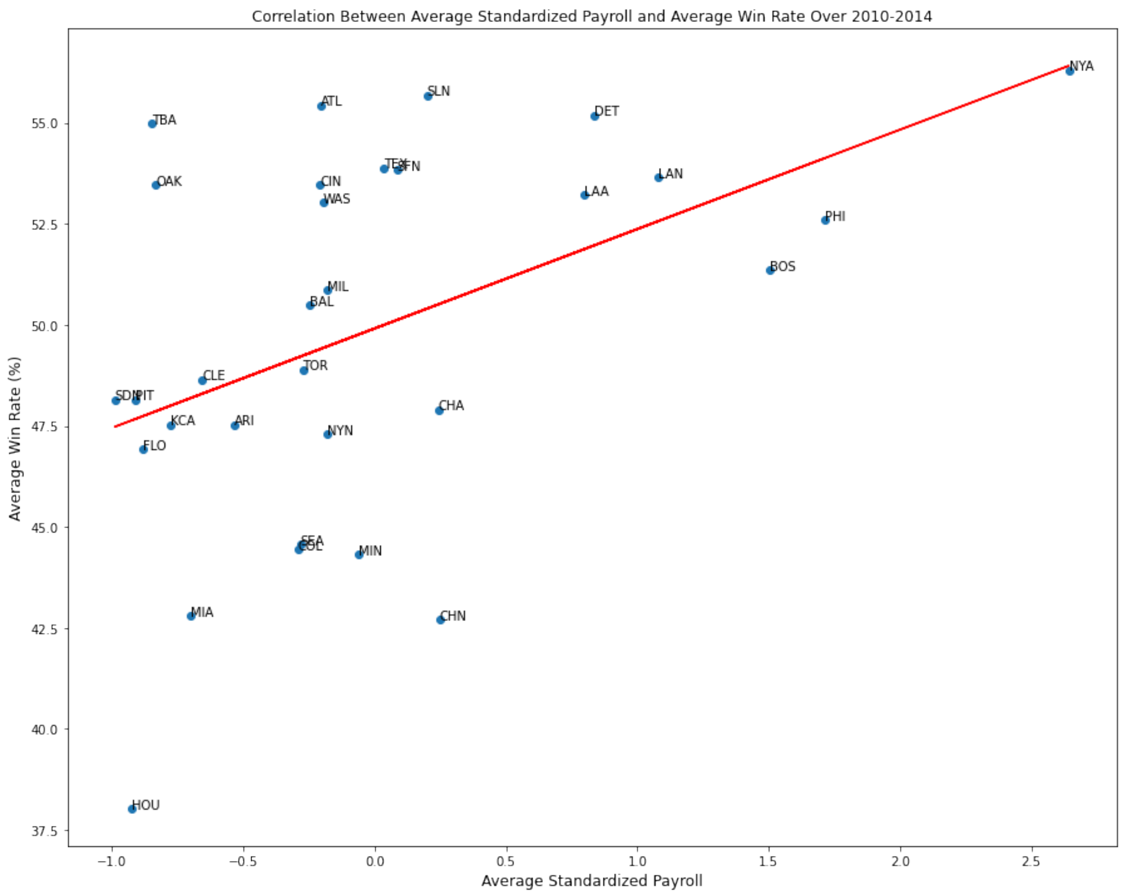
Correlation Between Average Standardized Payroll and Average Win Rate Over 1995-1999
Average Win Rate (%)
Average Standardized Payroll
40
45
50
55
60
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
ATL
NYA
CLE
HOU
BOS
TEX
LAN
BAL
CIN
NYN
SEA
SDN
ARI
SFN
COL
ANA
CHA
CAL
SLN
ML4
TOR
CHN
OAK
MON
MIL
FLO
PHI
PIT
KCA
MIN
DET
TBA



Correlation Between Average Standardized Payroll and Average Win Rate Over 2000-2004
Average Win Rate (%)
Average Standardized Payroll
60
55
50
45
40
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
2.5
NYA
OAK
ATL
SLN
SFN
SEA
BOS
LAN
MIN
CHA
HOU
ANA
ARI
FLO
PHI
CLE
CHN
TOR
NYN
TEX
CIN
SDN
COL
MON
BAL
PIT
KCA
MIL
TBA
DET



Correlation Between Average Standardized Payroll and Average Win Rate Over 2005-2009
Average Win Rate (%)
Average Standardized Payroll
60
55
50
45
40
−1
0
1
2
3
NYA
LAA
BOS
PHI
SLN
MIN
CHA
NYN
LAN
CLE
TOR
DET
ATL
CHN
MIL
OAK
FLO
HOU
TEX
COL
SDN
ARI
SFN
SEA
TBA
CIN
BAL
WAS
PIT
KCA

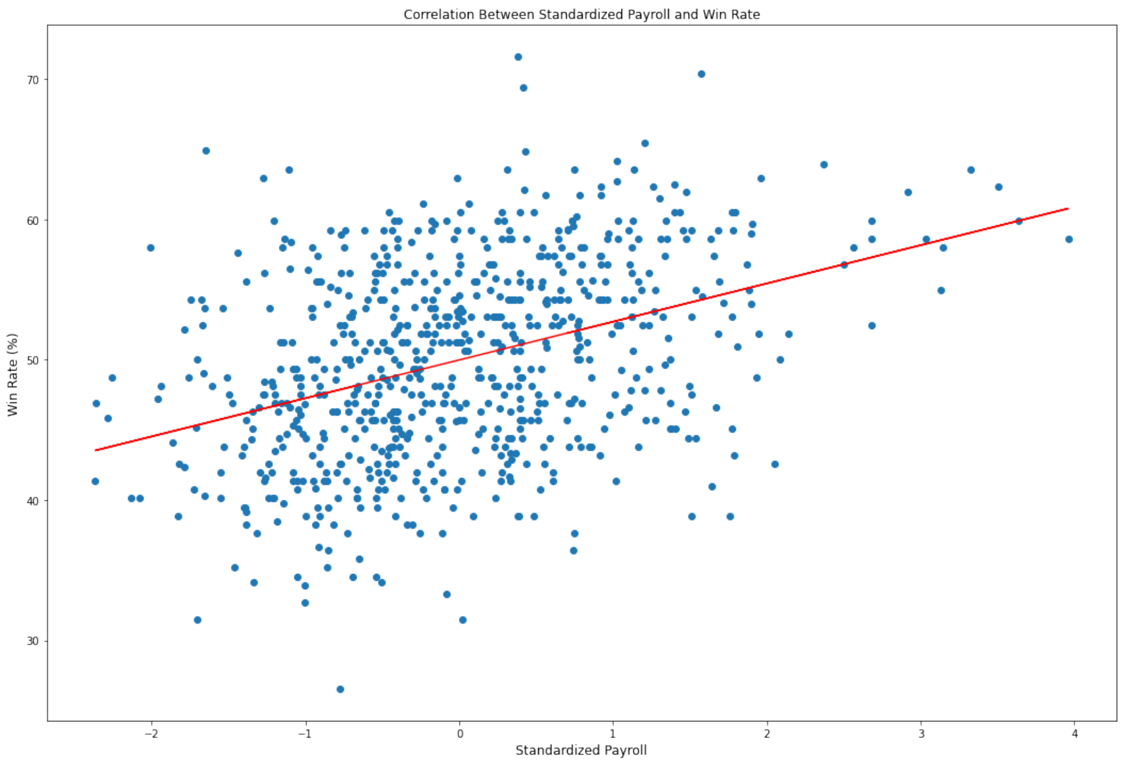


0.0.3 Question 3

The scatter points and lines of best fit from Problem 4 are almost identical to the ones generated by problem 6. The main difference of the plots generated by the two problems is the representation on the x-axis. In Problem 4, the x-axis for each plot only represented the average total payroll of each team for each time period, but the payroll amount does not tell us much about the data. In Problem 6, the x-axis is the average standardized payroll which can be used to analyze for each team's average payroll during that period, how many standard deviations away is it from the average payroll of all teams in that period. We can see that the scale of the x-axes for the plots in Problem 6 include negatives and positives indicating whether a team's spendings is below or above the average payroll of all teams, respectively.

[ ]:
```
### Problem 7

# Did the same thing as Problem 6 instead this time, removed the periods part
# Didn't use the average standardized payroll and win rate for each period
# Plotted standardized payroll and win rate for all years and all teams
# Found regression line using numpy.polyfit()
```

```python
ptable3 = standardized.drop(['avg_pay', 'std'], axis=1).copy()

plt.figure(figsize=(18,12))
title = "Correlation Between Standardized Payroll and Win Rate"
plt.title(title)
plt.xlabel('Standardized Payroll', size=12)
plt.ylabel('Win Rate (%)', size=12)
plt.scatter(ptable3['stdpay'], ptable3['win_rate'])

x = ptable3['stdpay'].values
y = ptable3['win_rate'].values
a, b = np.polyfit(x, y, 1)
plt.plot(x, a * x + b, "r-")

plt.show()
```



[ ]:
```python
### Problem 8

# Added the spending efficiency column using the formula provided
# Filtered the dataframe for the teams provided per instruction
# For each team, plotted spending efficiency over the years and labeled each␣
↪team in legend
```

```
eff = standardized.drop(['avg_pay', 'std'], axis=1).copy()

def get_efficiency(wr, ew):
  return wr - ew

eff['exp_win'] = eff['stdpay'].apply(lambda x : 50 + (2.5 * x))
eff['efficiency'] = eff.apply(lambda x : get_efficiency(x['win_rate'],␣
↪x['exp_win']), axis=1)
selected_teams = ['OAK', 'BOS', 'NYA', 'ATL', 'TBA']

plt.figure(figsize = (20, 12))
plt.title('Spending Efficiency of Teams Over the Years')
plt.xlabel('Year', size=12)
plt.ylabel('Efficiency', size=12)
plt.xticks(np.arange(1990, 2015, 4))

for team in selected_teams:
  team_data = eff.loc[eff['teamid'] == team]
  plt.plot(team_data['yearid'], team_data['efficiency'], label = team)

plt.legend()
plt.show()
```

0.0.4 Question 4

From this plot we can see that Oakland A's spending efficiency fluctuated a lot over the years. Oakland A's spending efficiency during the Moneyball period was significantly higher than any other team during that period and maintains as one of the highest spending efficiency of all teams during the entire 1990-2014 period that was studied. They started off with a high spending efficiency in their first year, but they struggled around 1993-1997 before going back up. They saw some struggle again around 2007-2011, but the spending efficiency was nowhere as bad as 1993-1997. For the last remaining years, they seem to have a relatively high spending efficiency. This plots reflects with what I had concluded from looking at the plots in Question 2 and 3, but offers more insight on the magnitude of their success and failures over the years represented using the numerical spending efficiency value. From the previous plots, I was able to notice that during the 1993-1997 and 2007-2011 periods they had lower win rates compared to other years, but I was not able to conclude how much worse they were performing during 1993-1997 compared to 2007-2011 as shown in this plot here.