Position Correlation

May 22, 2019

```
In [1]: import numpy as np
        import pandas as pd
        import seaborn as sns
        import matplotlib.pyplot as plt

In [2]: ls
```

Drafts.csv*   Standings.csv*
Lineups.csv*   TE Change.ipynb*
Matchups.csv*   notebook.tex
POS.csv*   output_9_0.png
Position Correlation.ipynb*

```
In [3]: all_lineups = pd.read_csv('./Lineups.csv',sep = ',')
        all_lineups = all_lineups.drop(['Team'],1)
        lineups = all_lineups[all_lineups.Starter == 1]
        lineups['Position'] = np.where(lineups['POS']=='W/R','FLEX',lineups['POS'])
```

```
/Users/aclark/anaconda3/lib/python3.6/site-packages/ipykernel_launcher.py:4: SettingWithCopyWa
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.html
  after removing the cwd from sys.path.
```

```
In [4]: full_standings = pd.read_csv('./Standings.csv',sep = ',')
        standings = full_standings[full_standings['Season'].isin(["2015-2016","2016-2017","201
        standings = standings[['Owner','Final_Standings','Season']]
        standings.head()

Out[4]:         Owner  Final_Standings      Season
        0        Stein                1   2017-2018
        1       Walton                2   2017-2018
        2       Ritzel                3   2017-2018
        3        Clark                4   2017-2018
        4     Whitmore                5   2017-2018
```

```
In [5]: lineups.shape

Out[5]: (5181, 40)

In [6]: lineups1 = lineups[['Season', 'Owner', 'Position', 'Fan Pts']]
        lineups1.head()

Out[6]:        Season      Owner Position  Fan Pts
        0  2016-2017  Whitmore       QB     32.3
        1  2016-2017  Whitmore       RB     16.5
        2  2016-2017  Whitmore       RB     10.1
        3  2016-2017  Whitmore       WR      4.7
        4  2016-2017  Whitmore       WR     14.8

In [7]: summed_lineups = lineups1.groupby(['Season', 'Owner', 'Position']).sum()
        summed_lineups=pd.DataFrame(summed_lineups.reset_index())

In [8]: combined = pd.merge(standings, summed_lineups)

In [9]: combined.head(6)

Out[9]:    Owner  Final_Standings     Season Position  Fan Pts
        0  Stein                1  2017-2018      DEF   263.85
        1  Stein                1  2017-2018        K   138.94
        2  Stein                1  2017-2018       QB   482.91
        3  Stein                1  2017-2018       RB   576.34
        4  Stein                1  2017-2018       TE   122.92
        5  Stein                1  2017-2018       WR   530.05

In [10]: RB = combined[combined.Position=="RB"]
         RB_matrix = RB.corr()
         ax = sns.regplot(x = "Final_Standings", y="Fan Pts", data=RB, color='r')
         ax.set(xlabel="Final Standings",xticks=np.arange(1,13,1),ylabel="Total Fantasy Points"
                 title="Total Position Points by Final Standing - RB")
         plt.legend(["Correlation:{:f}".format(round(RB_matrix.iloc[0,1],6))])
         plt.show()
```
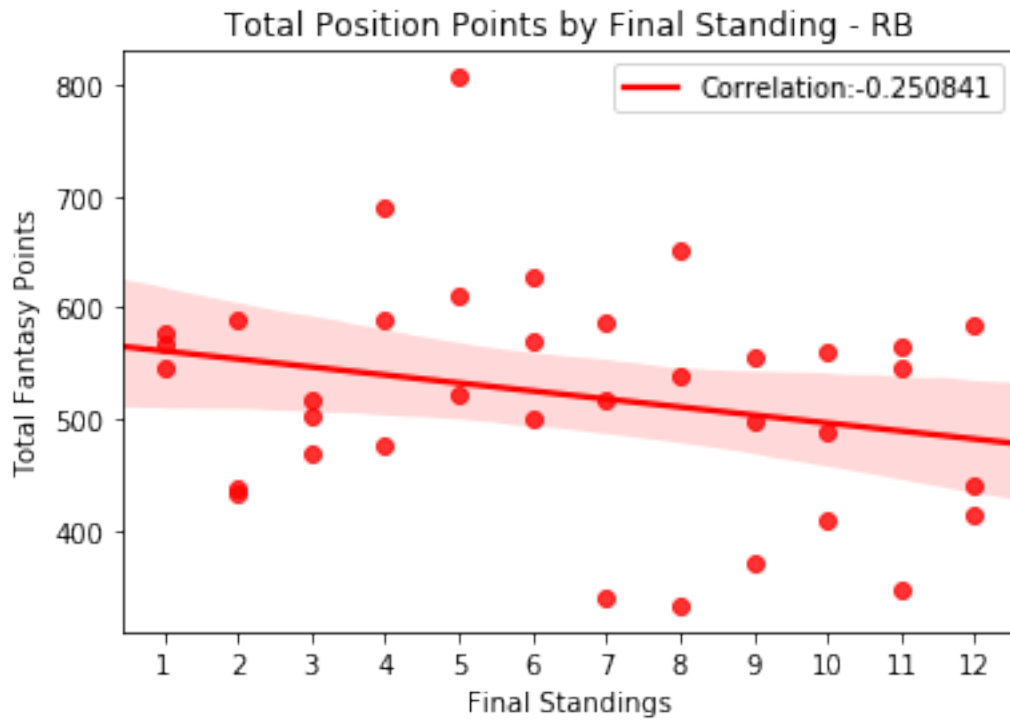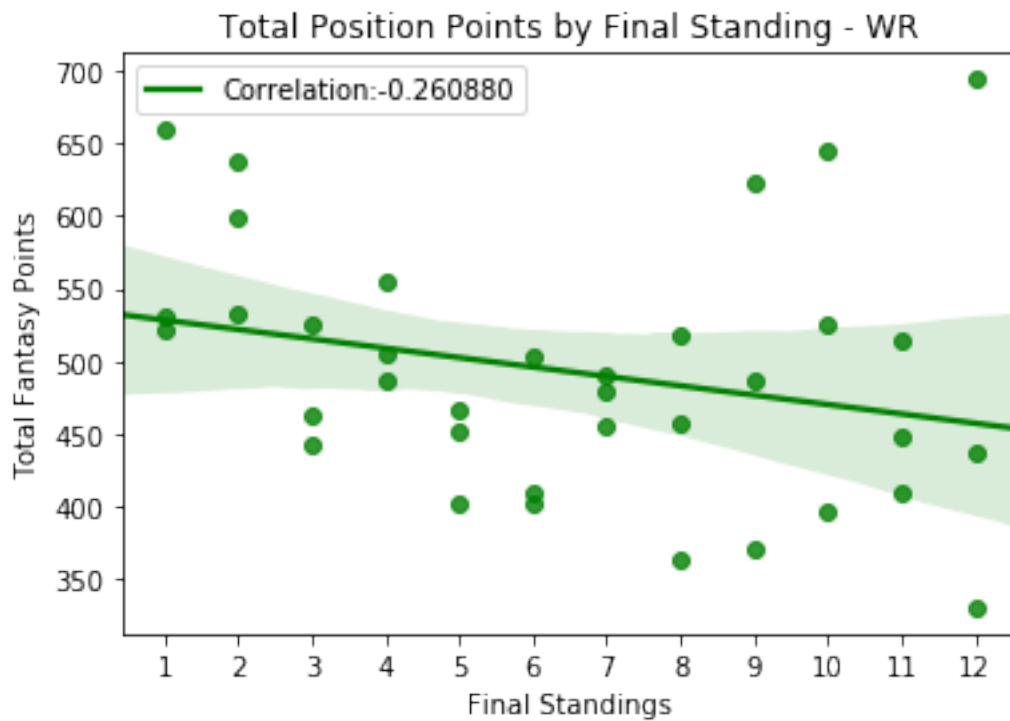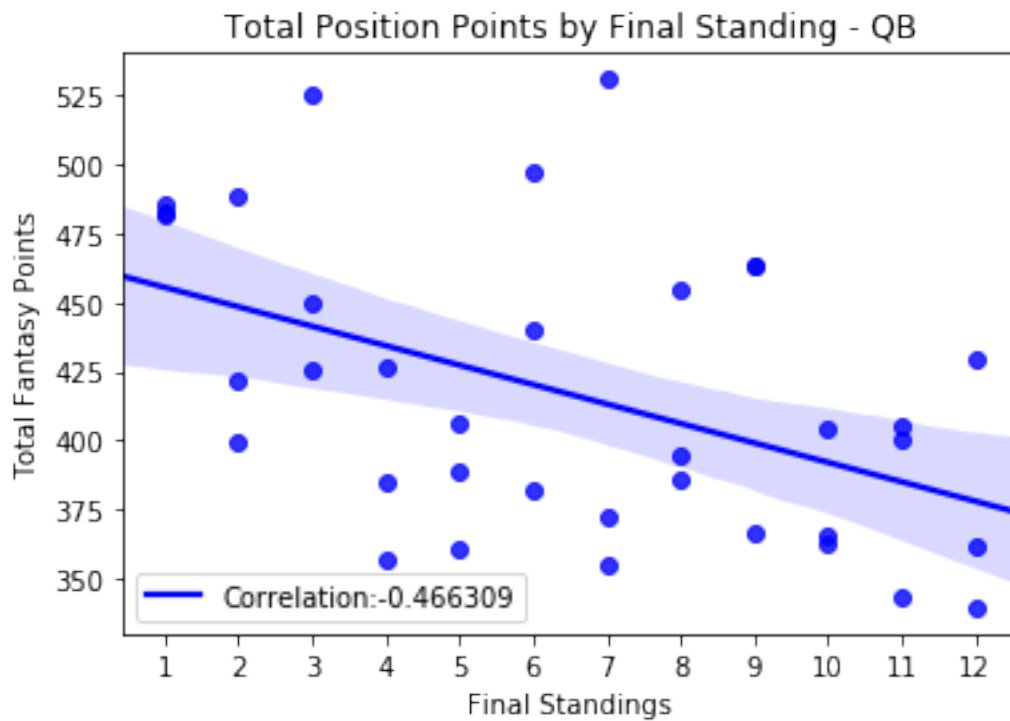
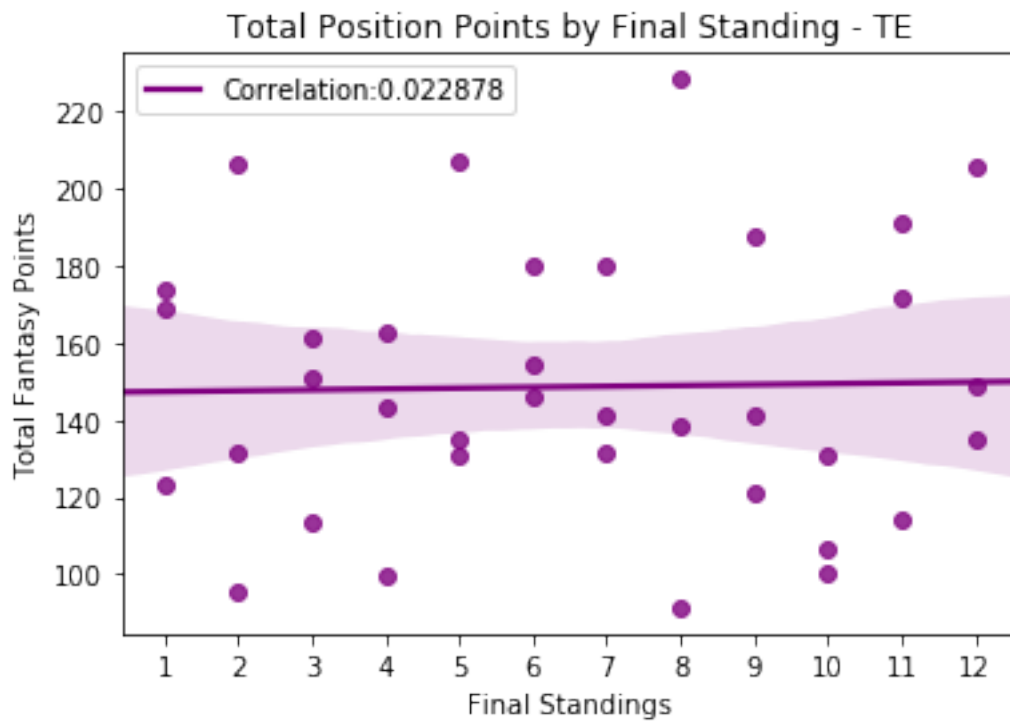Total Position Points by Final Standing - RB

```
In [11]: WR = combined[combined.Position=="WR"]
         WR_matrix = WR.corr()
         ax = sns.regplot(x = "Final_Standings", y="Fan Pts", data=WR, color='g')
         ax.set(xlabel="Final Standings",xticks=np.arange(1,13,1),ylabel="Total Fantasy Points",
                title="Total Position Points by Final Standing - WR")
         plt.legend(["Correlation:{:f}".format(round(WR_matrix.iloc[0,1],6))])
         plt.show()
```

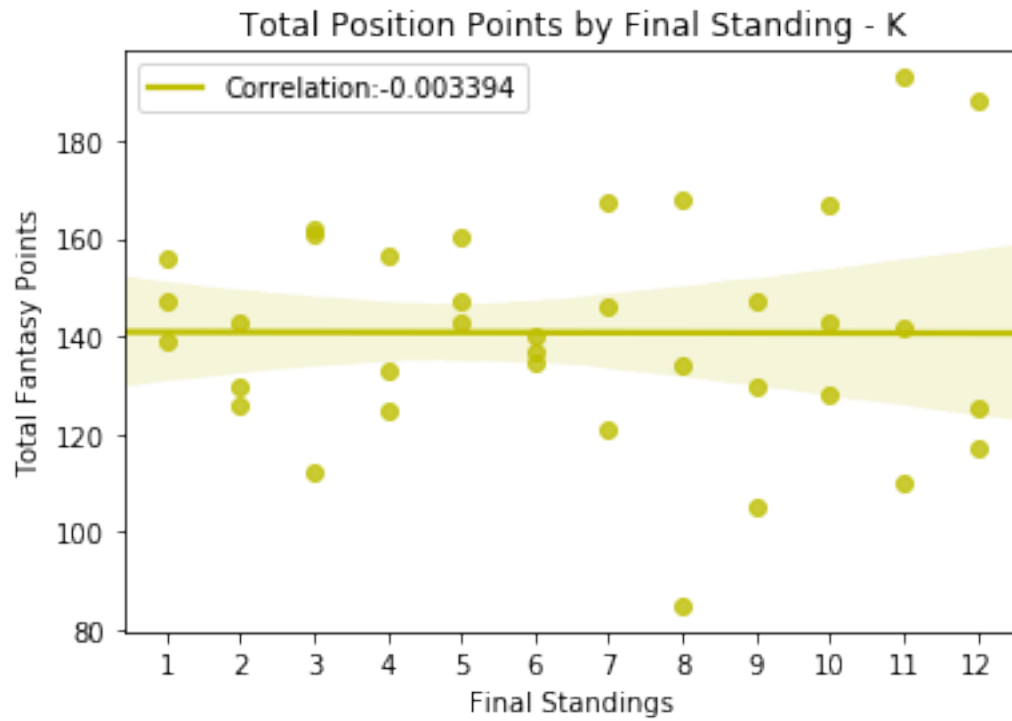Total Position Points by Final Standing - WR

```
In [12]: QB = combined[combined.Position=="QB"]
         QB_matrix = QB.corr()
         ax = sns.regplot(x = "Final_Standings", y="Fan Pts", data=QB, color='b')
         ax.set(xlabel="Final Standings",xticks=np.arange(1,13,1),ylabel="Total Fantasy Points"
                title="Total Position Points by Final Standing - QB")
         plt.legend(["Correlation:{:f}".format(round(QB_matrix.iloc[0,1],6))], loc="lower left"
         plt.show()
```

Total Position Points by Final Standing - QB

```
In [13]: TE = combined[combined.Position=="TE"]
         TE_matrix = TE.corr()
         ax = sns.regplot(x = "Final_Standings", y="Fan Pts", data=TE, color='purple')
         ax.set(xlabel="Final Standings",xticks=np.arange(1,13,1),ylabel="Total Fantasy Points"
                title="Total Position Points by Final Standing - TE")
         plt.legend(["Correlation:{:f}".format(round(TE_matrix.iloc[0,1],6))])
         plt.show()
```

Total Position Points by Final Standing - TE

```
In [14]: K = combined[combined.Position=="K"]
         K_matrix = K.corr()
         ax = sns.regplot(x = "Final_Standings", y="Fan Pts", data=K, color='y')
         ax.set(xlabel="Final Standings",xticks=np.arange(1,13,1),ylabel="Total Fantasy Points"
                title="Total Position Points by Final Standing - K")
         plt.legend(["Correlation:{:f}".format(round(K_matrix.iloc[0,1],6))])
         plt.show()
```

Total Position Points by Final Standing - K

```
In [15]: DEF = combined[combined.Position=="DEF"]
         DEF_matrix = DEF.corr()
         ax = sns.regplot(x = "Final_Standings", y="Fan Pts", data=DEF, color='teal')
         ax.set(xlabel="Final Standings",xticks=np.arange(1,13,1),ylabel="Total Fantasy Points"
                title="Total Position Points by Final Standing - DEF")
         plt.legend(["Correlation:{:f}".format(round(DEF_matrix.iloc[0,1],6))])
         plt.show()
```

Total Position Points by Final Standing - DEF