

# Stats Final Project

Siddharth Gowda

6/3/2020

## NFL Stats Project

### Introduction

For my final project I decided to learn how to use R. R is a statistical programming language that despite being basic is extremely useful for statisticians. I'm learning almost from scratch so I will mostly be doing simple plotting and analysis on the 2019 NFL statistics per team.

---

**Was there a correlation between a teams strength of schedule and their win percentage in the 2019 NFL season?**

```
AFC = read.csv("AFC_Data.csv")
NFC = read.csv("NFC_Data.csv")
head(AFC)
```

```
##           Tm W  L T  W.L.  PF  PA  PD  MoV  SoS  SRS OSRS DSRS
## 1 New England Patriots* 12  4  0  0.750  420  225  195  12.2 -1.8  10.4  2.8  7.6
## 2      Buffalo Bills+ 10  6  0  0.625  314  259   55   3.4 -1.3   2.2 -3.5  5.7
## 3      New York Jets  7  9  0  0.438  276  359  -83  -5.2 -1.1  -6.3 -5.7 -0.6
## 4      Miami Dolphins  5 11  0  0.313  306  494 -188 -11.8  0.2 -11.6 -2.4 -9.1
## 5      Baltimore Ravens* 14  2  0  0.875  531  282  249  15.6  0.1  15.6 11.0  4.7
## 6      Pittsburgh Steelers  8  8  0  0.500  289  303  -14  -0.9  1.2   0.3 -4.3  4.6
```

```
head(NFC)
```

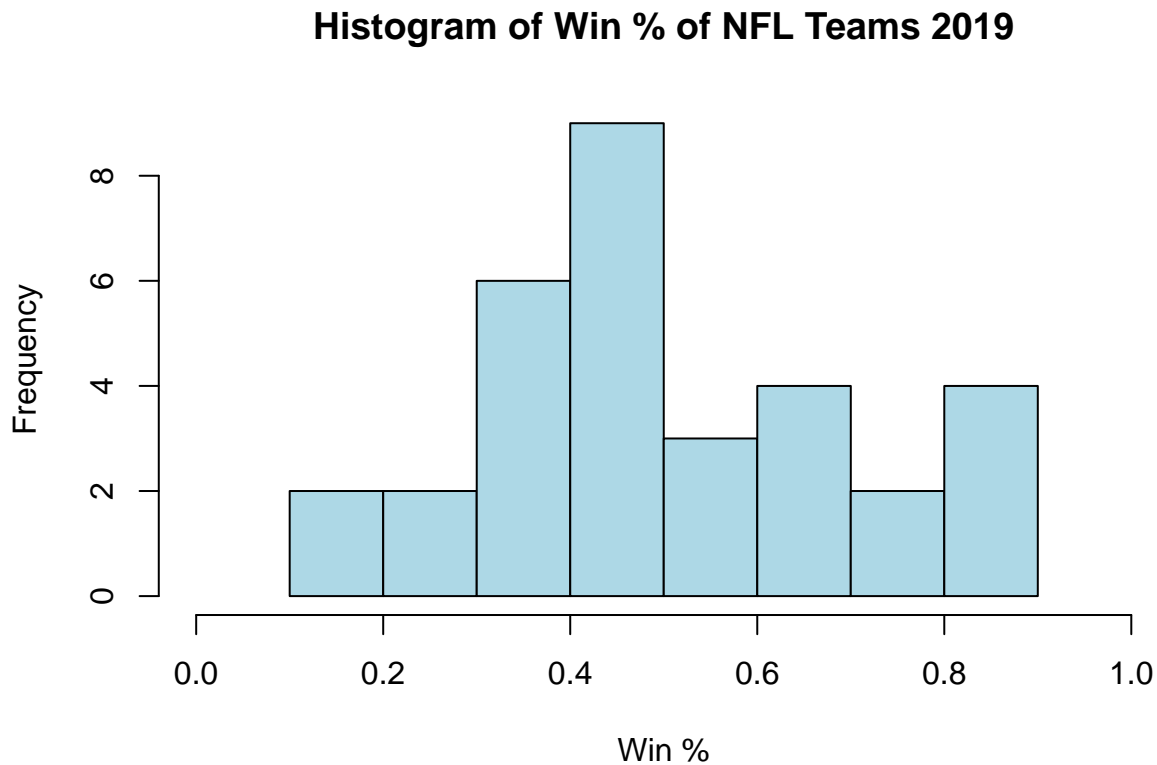
```
##           Tm W  L T  W.L.  PF  PA  PD  MoV  SoS  SRS OSRS DSRS
## 1 Philadelphia Eagles*  9  7  0  0.563  385  354   31   1.9 -1.7   0.3  0.7 -0.4
## 2      Dallas Cowboys  8  8  0  0.500  434  321  113   7.1 -1.8   5.3  3.8  1.5
## 3      New York Giants  4 12  0  0.250  341  451 -110  -6.9 -1.0  -7.9 -1.8 -6.1
## 4 Washington Redskins  3 13  0  0.188  266  435 -169 -10.6 -0.2 -10.8 -6.3 -4.5
## 5  Green Bay Packers* 13  3  0  0.813  376  313   63   3.9 -0.7   3.2  0.6  2.6
## 6  Minnesota Vikings+ 10  6  0  0.625  407  303  104   6.5 -1.1   5.4  2.5  2.9
```

```
NFL = rbind(AFC, NFC)
tail(NFL)
```

```
##           Tm W  L T W.L.  PF  PA  PD  MoV  SoS  SRS  OSRS  DSRS
## 27 Tampa Bay Buccaneers  7  9  0 0.438 458 449   9  0.6 -0.2  0.4  4.9 -4.5
## 28   Carolina Panthers  5 11  0 0.313 340 470 -130 -8.1  1.1 -7.0 -1.9 -5.1
## 29 San Francisco 49ers* 13  3  0 0.813 479 310  169 10.6  0.4 11.0  6.7  4.3
## 30   Seattle Seahawks+ 11  5  0 0.688 405 398   7  0.4  2.3  2.7  2.9 -0.2
## 31    Los Angeles Rams  9  7  0 0.563 394 364   30  1.9  2.0  3.9  2.2  1.7
## 32   Arizona Cardinals  5 10  1 0.344 361 442  -81 -5.1  1.8 -3.2 -0.3 -2.9
```

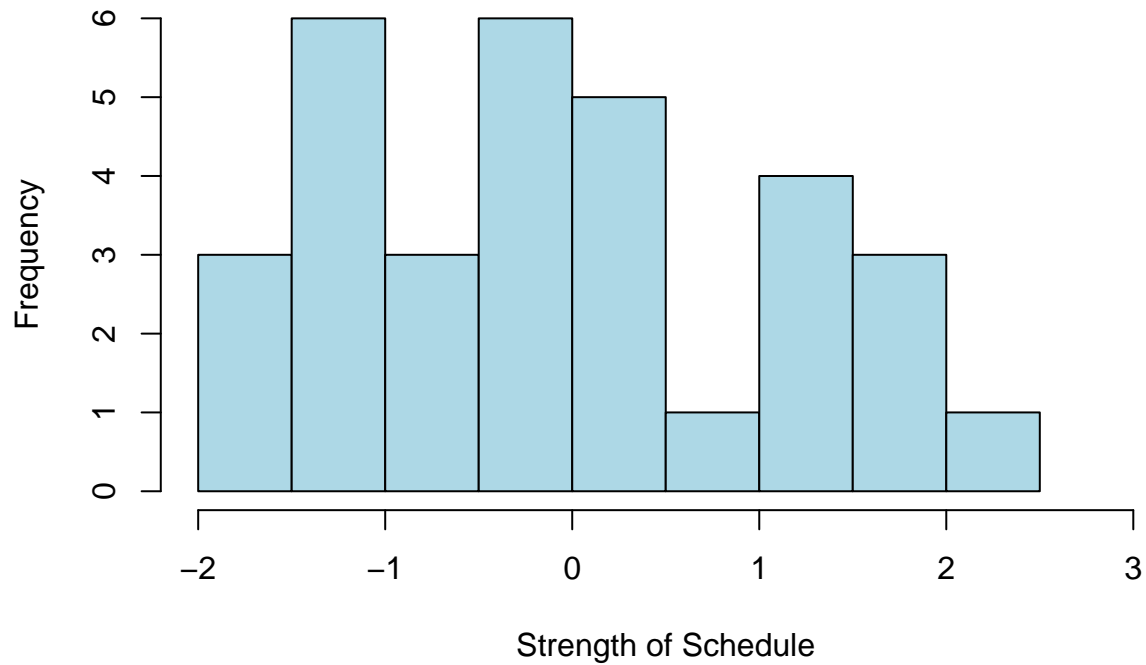
Just loaded and combined the NFC and AFC datasets/dataframes into one set (from Pro Football Reference and converted into csv)

```
hist(NFL$W.L., main = "Histogram of Win % of NFL Teams 2019",
     xlab = "Win %",
     col = "Light Blue", xlim = c(0,1))
```



```
hist(NFL$SoS, main = "Histogram of Strength of Schedule of NFL Teams 2019",
     xlab = "Strength of Schedule", col = "Light Blue", xlim = c(-2,3))
```

## Histogram of Strength of Schedule of NFL Teams 2019



Strength of schedule measures the average difficulty of each opponent the team plays. It is a statistics calculated by Football Reference that is usually centered around 0 and approximately symmetric (although it is not that apparent in this graph).

Win percentage is typically centered around 0 and is approximately symmetric.

W.L. = Win percentage SoS = Strength of schedule

Why did I plot a histogram: I wanted to see the distribution of the two variables and if there were apparent outliers.

---

### *#Linear Regression*

```
NFL_WLSoS = lm(NFL$W.L. ~ NFL$SoS)
summary(NFL_WLSoS)
```

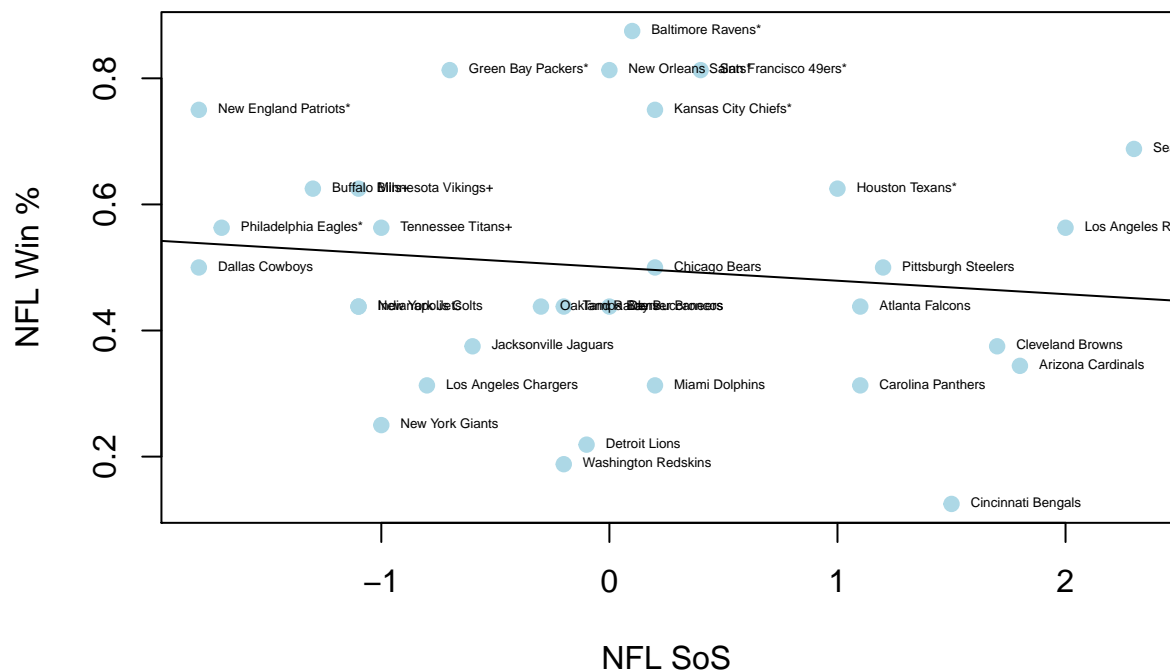
```
##
## Call:
## lm(formula = NFL$W.L. ~ NFL$SoS)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.34331 -0.12295 -0.03874  0.11552  0.37685
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.50028    0.03532   14.162 7.99e-15 ***
## NFL$SoS       -0.02131    0.03114   -0.684   0.499
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1998 on 30 degrees of freedom
## Multiple R-squared:  0.01538,    Adjusted R-squared:  -0.01744
## F-statistic: 0.4685 on 1 and 30 DF,  p-value: 0.4989
```

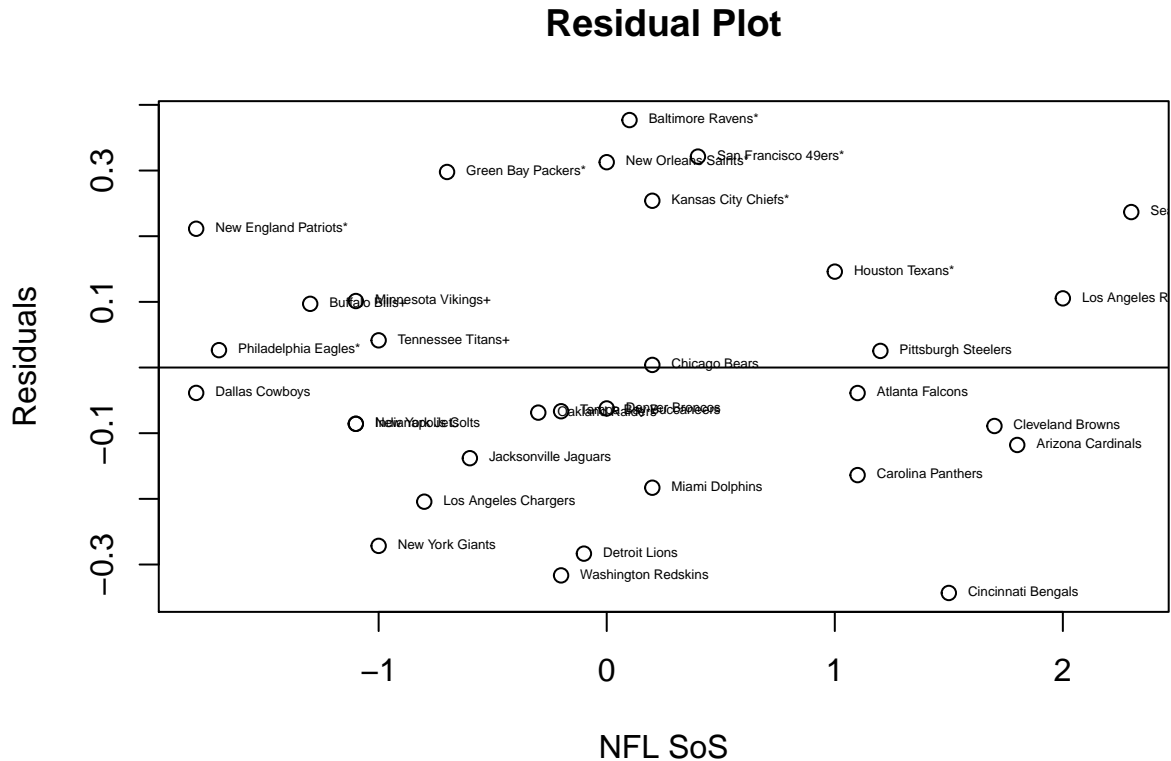
Since the p-value (0.4989) is greater than  $\alpha = 0.05$  we fail to reject the null hypothesis. At  $\alpha = 0.05$  there is no significant evidence that there is a linear relationship between the strength of schedule and the win percentage of an NFL team in 2019. However, these calculations provide some evidence to the motto “any given Sunday” because the lack of evidence for a linear relationship suggests that perhaps upset occur frequently in the NFL. This is a question that I would investigate further if I had more time.

```
plot(NFL$SoS, NFL$W.L.,
     xlab = "NFL SoS",
     ylab = "NFL Win %",
     main = "NFL Strength of Schedule against Win % 2019",
     pch = 19,
     col = "light blue")
with(NFL, text(NFL$W.L. ~ NFL$SoS,
              labels=Tm,
              pos=4,
              cex=.4))
abline(NFL_WLSoS)
```

## NFL Strength of Schedule against Win % 2019



```
res_NFL_WLSoS = residuals(NFL_WLSoS)
plot(NFL$SoS, res_NFL_WLSoS, main = "Residual Plot", xlab = "NFL SoS", ylab = "Residuals")
with(NFL, text(res_NFL_WLSoS ~ NFL$SoS, labels=Tm, pos=4, cex=.4))
abline(h = 0)
```



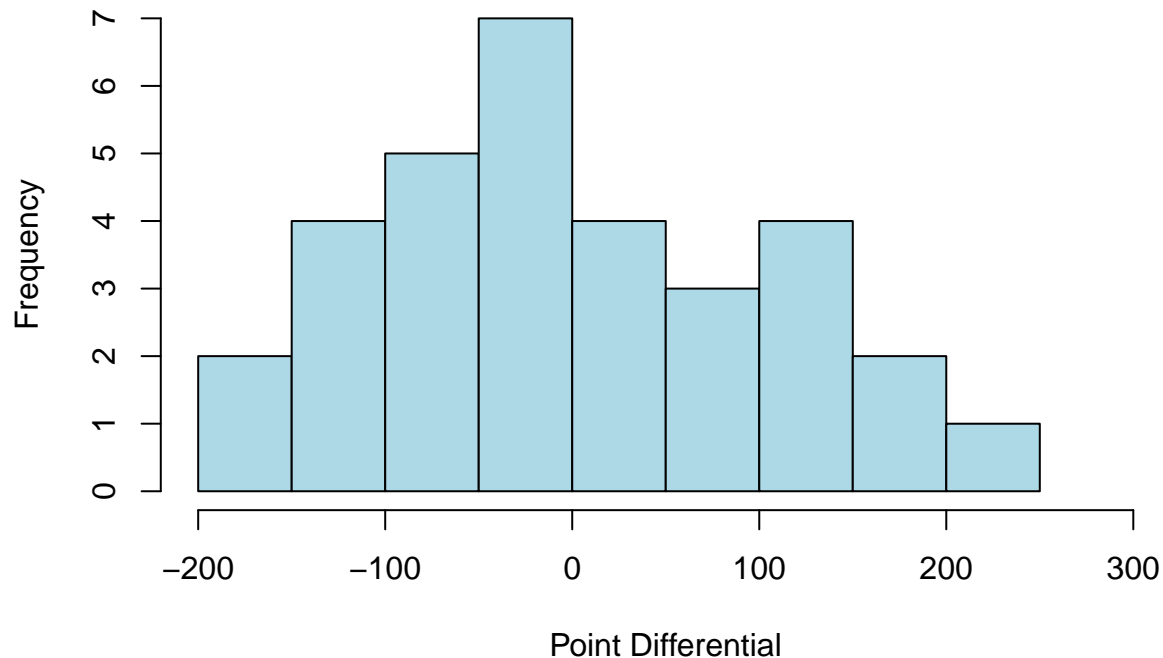
Despite the fact that there does not appear to be a linear relationship between the two variables, there seems to be random scatter in the residual plot.

---

**Was there a correlation between the win percentage of a NFL team and their point differential in 2019?**

```
hist(NFL$PD, breaks = 8, main = "Histogram of NFL Point Differential 2019",
     xlab = "Point Differential",
     col = "Light Blue", xlim = c(-200, 300))
```

## Histogram of NFL Point Differential 2019



Point Differential (the sum of all of the points a team scores subtracted by all the points the team surrenders over the season, typically centered around 0 and approximately symmetric)

### # Linear Regression

```
NFL_WLPD = lm(NFL$W.L. ~ NFL$PD)
summary(NFL_WLPD)
```

```
##
## Call:
## lm(formula = NFL$W.L. ~ NFL$PD)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.182555 -0.045230  0.002638  0.037029  0.211097
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.5002812   0.0171115   29.237 < 2e-16 ***
## NFL$PD       0.0016130   0.0001614    9.992 4.66e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0968 on 30 degrees of freedom
## Multiple R-squared:  0.769, Adjusted R-squared:  0.7613
## F-statistic: 99.85 on 1 and 30 DF, p-value: 4.657e-11
```

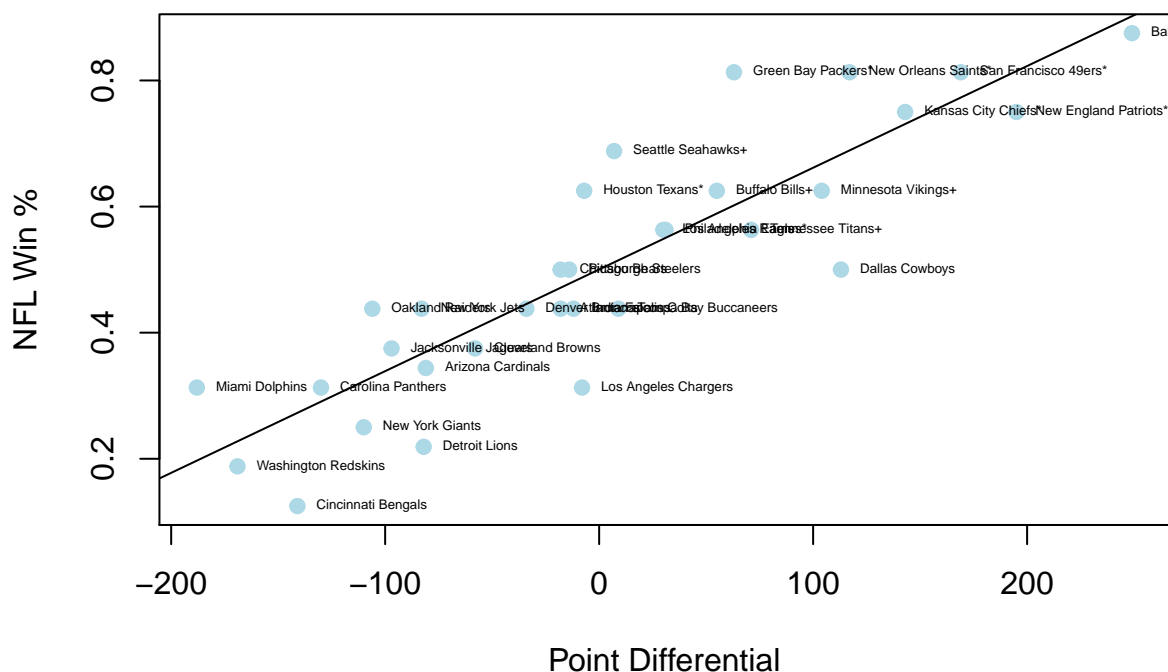
Since the p-value ( $4.567 \times 10^{-11}$ ) is less than  $\alpha = 0.05$  we reject the null hypothesis. At  $\alpha = 0.05$

there is statistically evidence that suggests that the relationship between point differential and NFL Win % in the 2019 season could be a positive linear relationship.

---

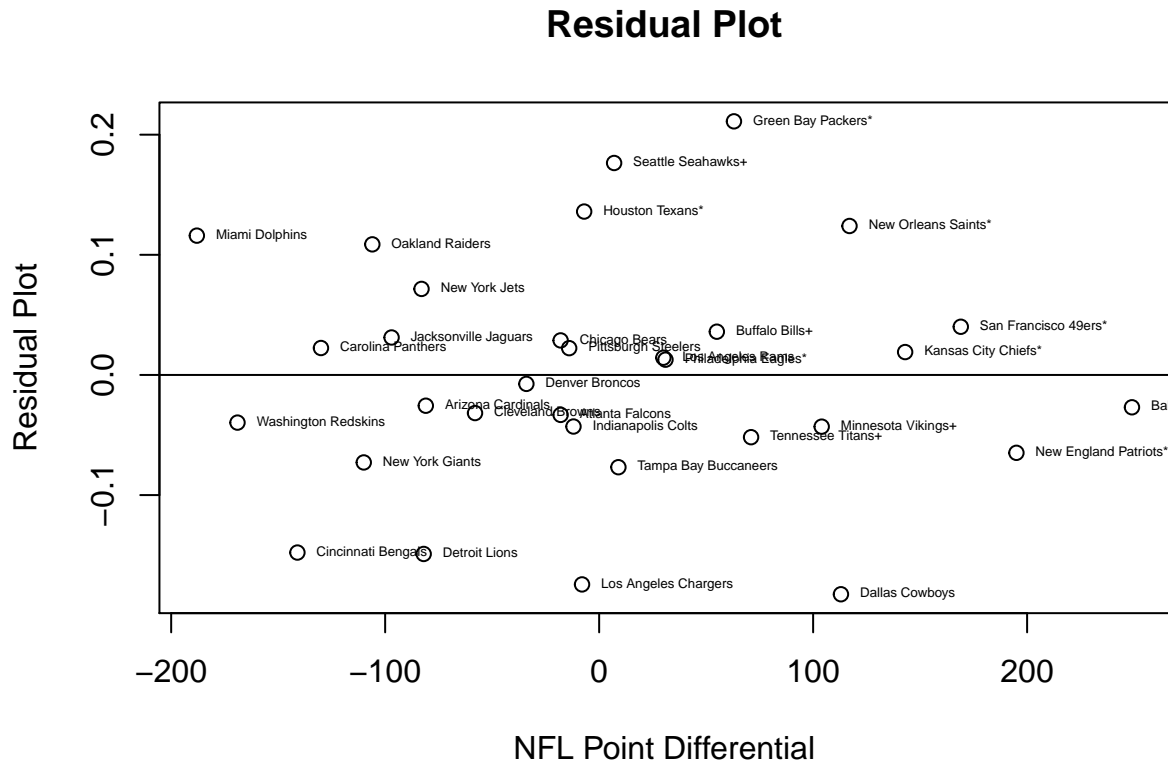
```
plot(NFL$PD, NFL$W.L.,
     xlab = "Point Differential",
     ylab = "NFL Win %",
     main = "Point Differential against Win % NFL 2019",
     pch = 19,
     col = "light blue")
with(NFL, text(NFL$W.L. ~ NFL$PD,
              labels=Tm,
              pos=4,
              cex = .4))
abline(NFL_WLPD)
```

### Point Differential against Win % NFL 2019




---

```
res_WLPD = residuals(NFL_WLPD)
plot(NFL$PD,
     res_WLPD,
     main = "Residual Plot",
     xlab = "NFL Point Differential",
     ylab = "Residual Plot")
with(NFL, text(res_WLPD ~ NFL$PD, labels=Tm, pos=4, cex = .4))
abline(h = 0)
```



The residual plot is pretty evenly scattered, which supports the fact that a relationship between the point differential and win % in the 2019 NFL season could be linear.

## Finding Clusters

```
NFL_cluster = read.csv("cluster_NFL_reducedvariables4.csv")
head(NFL_cluster)
```

```
##           Tm  W.L.   PD OSRS DSRS
## 1 Philadelphia Eagles* 0.563   31  0.7 -0.4
## 2      Dallas Cowboys 0.500  113  3.8  1.5
## 3    New York Giants 0.250 -110 -1.8 -6.1
## 4 Washington Redskins 0.188 -169 -6.3 -4.5
## 5   Green Bay Packers* 0.813   63  0.6  2.6
## 6 Minnesota Vikings+ 0.625  104  2.5  2.9
```

I removed some of the variables so the clustering model would not over-fit. This occurs because as the number of variables increases, out-of sample error (model error) decreases but then increases. This is true since some variables should have a greater impact on the model.

I removed points scored by opposition, points scored by team, and margin of victory since those variables are similar to point differential. Also, win loss count of was similar to win % and the  $SRS = OSRS + DSRS$ ; thus, I removed those two variables as well. Strength of schedule was removed because it seems to have a weak relationship with the other variables, and therefore is not a good variable to cluster teams with.



---

Normalization (Z-score)

```
z = NFL_cluster[, -c(1,1)]
m = apply(z,2,mean) # Find means
s = apply(z,2,sd) # Find Standard devs
z = scale(z,m,s) # Z-score
```

2 = columns

Euclidean Distance (distance formula with 4 variables)

```
d = dist(z)
print(d, digits = 3)
```

```
##      1      2      3      4      5      6      7      8      9     10     11     12
## 2  1.223
## 3  2.599 3.408
## 4  3.328 4.222 1.333
## 5  1.511 1.845 4.008 4.561
## 6  1.216 0.798 3.748 4.424 1.125
## 7  2.033 2.670 3.257 3.148 2.334 2.361
## 8  2.280 2.967 0.639 1.498 3.730 3.344 2.872
## 9  1.954 1.621 4.472 5.264 1.188 1.145 3.292 4.165
## 10 0.784 1.597 2.016 2.698 2.181 1.786 1.908 1.605 2.638
## 11 1.617 1.874 2.229 3.435 2.882 2.382 3.450 2.042 2.772 1.565
## 12 2.389 3.291 0.451 1.307 3.731 3.555 2.978 0.731 4.258 1.802 2.201
## 13 2.619 1.947 5.131 5.921 1.839 1.564 3.774 4.770 0.820 3.243 3.341 4.939
## 14 0.860 1.453 3.117 3.914 1.228 1.251 2.557 2.873 1.457 1.432 1.752 2.860
## 15 0.655 0.922 3.035 3.725 1.376 0.820 2.066 2.638 1.654 1.052 1.855 2.810
## 16 1.668 2.482 1.057 1.899 3.076 2.771 2.485 0.726 3.531 1.003 1.645 0.841
## 17 2.784 2.173 5.311 5.848 1.888 1.611 3.187 4.874 1.671 3.323 3.939 5.107
## 18 1.920 2.246 3.926 4.076 1.597 1.696 1.081 3.525 2.512 2.168 3.498 3.659
## 19 1.991 3.014 1.973 1.808 2.911 2.955 1.486 1.790 3.797 1.585 2.906 1.674
## 20 3.371 4.302 1.118 1.659 4.636 4.571 4.036 1.738 5.133 2.896 2.887 1.173
## 21 3.826 2.992 6.215 7.113 3.134 2.816 5.076 5.861 2.033 4.425 4.208 6.064
## 22 1.854 2.438 3.228 3.214 2.169 2.133 0.272 2.816 3.080 1.741 3.275 2.947
## 23 1.318 2.138 1.471 2.152 2.708 2.377 2.106 1.061 3.199 0.591 1.663 1.243
## 24 3.067 3.801 1.253 0.745 4.377 4.078 2.934 1.103 4.988 2.353 3.125 1.268
## 25 0.478 1.541 2.658 3.373 1.405 1.403 2.127 2.407 1.925 0.965 1.700 2.388
## 26 0.711 0.703 3.188 3.877 1.318 0.563 2.097 2.783 1.552 1.229 1.968 2.997
## 27 0.861 1.781 1.909 2.512 2.235 1.923 1.770 1.527 2.794 0.338 1.740 1.698
## 28 2.045 3.045 1.396 1.377 3.191 3.123 1.972 1.253 3.953 1.505 2.614 1.108
## 29 2.122 1.464 4.596 5.424 1.590 1.160 3.467 4.242 0.519 2.742 2.791 4.414
## 30 1.791 2.543 2.667 2.601 2.477 2.390 0.598 2.279 3.359 1.488 3.039 2.401
## 31 1.923 2.988 1.060 1.672 3.146 3.122 2.562 1.176 3.772 1.485 2.099 0.709
## 32 1.439 1.997 1.934 2.343 2.729 2.240 1.715 1.394 3.253 0.809 2.117 1.800
##      13     14     15     16     17     18     19     20     21     22     23     24
## 2
## 3
## 4
## 5
```

```

## 6
## 7
## 8
## 9
## 10
## 11
## 12
## 13
## 14 2.208
## 15 2.218 0.845
## 16 4.169 2.187 2.018
## 17 1.340 2.686 2.364 4.326
## 18 2.890 2.252 1.777 3.027 2.167
## 19 4.453 2.590 2.361 1.522 4.236 2.340
## 20 5.857 3.719 3.841 1.961 6.146 4.722 2.613
## 21 1.328 3.385 3.440 5.296 2.285 4.176 5.733 6.925
## 22 3.550 2.360 1.830 2.392 2.994 0.963 1.560 4.034 4.848
## 23 3.819 1.903 1.630 0.449 3.905 2.595 1.372 2.378 4.981 1.990
## 24 5.582 3.692 3.380 1.592 5.491 3.831 1.800 1.975 6.741 2.956 1.809
## 25 2.655 0.582 0.791 1.723 2.915 2.027 2.016 3.308 3.875 1.954 1.414 3.169
## 26 2.075 1.032 0.393 2.212 2.146 1.693 2.480 4.028 3.295 1.872 1.819 3.531
## 27 3.418 1.598 1.256 0.980 3.428 2.116 1.324 2.792 4.620 1.642 0.581 2.217
## 28 4.619 2.650 2.452 1.092 4.527 2.780 0.584 2.092 5.856 2.000 1.099 1.343
## 29 0.553 1.728 1.744 3.646 1.549 2.683 4.018 5.325 1.720 3.238 3.310 5.085
## 30 3.898 2.415 1.934 1.929 3.471 1.509 0.992 3.478 5.188 0.624 1.576 2.364
## 31 4.496 2.369 2.408 0.847 4.647 3.164 1.273 1.567 5.669 2.537 1.061 1.723
## 32 3.775 2.197 1.646 1.127 3.624 2.233 1.393 2.957 4.950 1.602 0.785 1.892
##      25      26      27      28      29      30      31
## 2
## 3
## 4
## 5
## 6
## 7
## 8
## 9
## 10
## 11
## 12
## 13
## 14
## 15
## 16
## 17
## 18
## 19
## 20
## 21
## 22
## 23
## 24
## 25
## 26 0.992
## 27 1.067 1.370

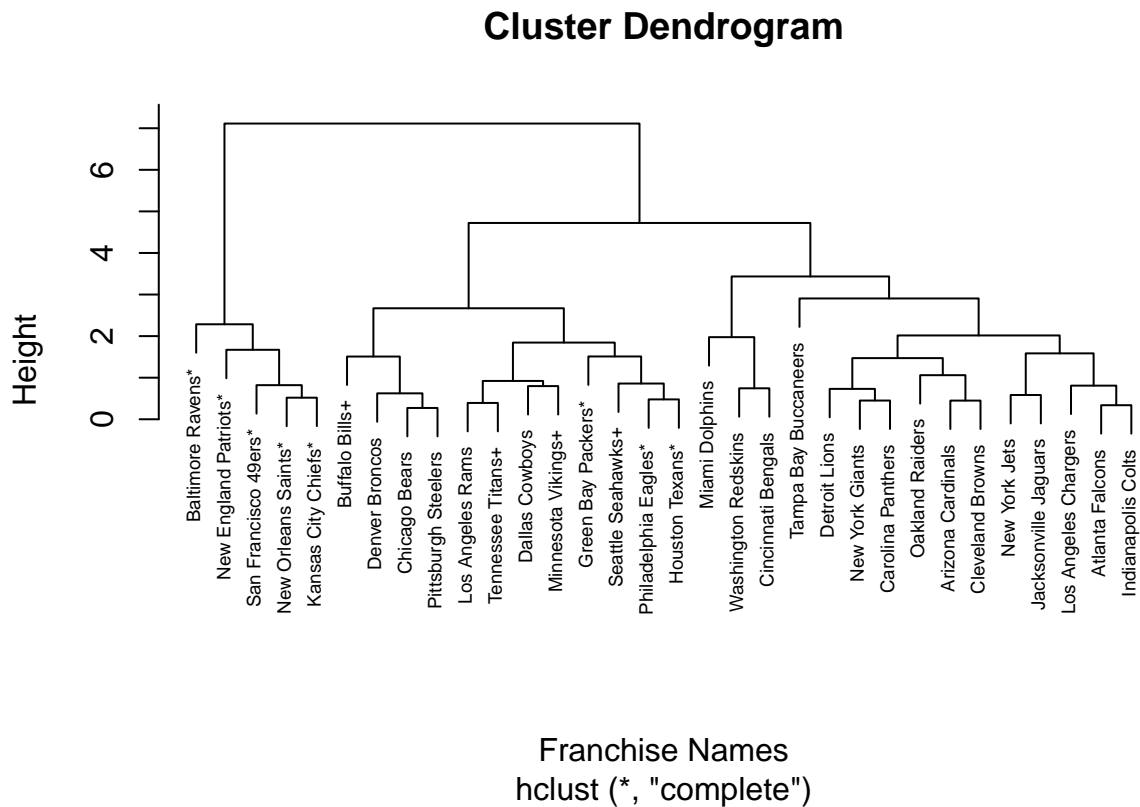
```

```
## 28 2.081 2.596 1.269
## 29 2.173 1.609 2.925 4.144
## 30 1.909 2.003 1.312 1.409 3.521
## 31 1.866 2.587 1.328 0.851 3.988 2.033
## 32 1.711 1.718 0.708 1.305 3.303 1.205 1.659
```

Example: team x and y =  $[(x.w\%-y.w\%)^2 + (x.PD-y.PD)^2 + (x.OSRS-y.OSRS)^2 + (x.DSRS-y.PDSRS)^2]^{.5}$  but calculate for every team against every other team

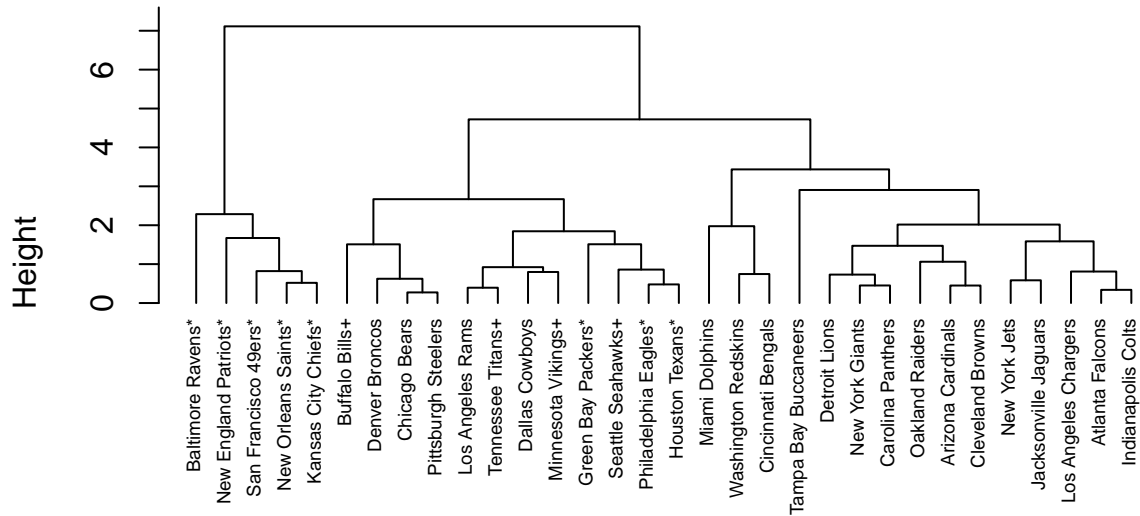
Cluster Dendrogram

```
hc_c = hclust(d)
plot(hc_c, labels = NFL_cluster$Tm, cex = .6, xlab = "Franchise Names")
```



```
plot(hc_c, hang = -1,
     labels = NFL_cluster$Tm,
     cex = .6,
     xlab = "Franchise Names")
```

## Cluster Dendrogram



### Franchise Names hclust (\*, "complete")

Complete

Linkage Used (means smaller clusters are used to create bigger ones) This means that the algorithm is finding the two, in this case, teams that have the closest overall z-scores.

Height = The number of levels in the dendrogram

Based on the diagram, there are three main clusters, in my opinion: Baltimore to Kansas City (the elite teams), Buffalo to Houston (best of the non-playoff teams and lower-end play-off teams), and Miami to Indianapolis (non-playoff teams). However, there are some potential anomalies like the Broncos and Bears in the second group.