Stat 154 HW02

Mokssh Surve

2/9/2020

```
library("plotrix")
library('fields')
```

```
## Loading required package: spam
```

```
## Loading required package: dotCall64
```

```
## Loading required package: grid
```

```
## Spam version 2.5-1 (2019-12-12) is loaded.
## Type 'help( Spam)' or 'demo( spam)' for a short introduction
## and overview of this package.
## Help for individual functions is also obtained by adding the
## suffix '.spam' to the function name, e.g. 'help( chol.spam)'.
```

```
##
## Attaching package: 'spam'
```

```
## The following objects are masked from 'package:base':
##
##     backsolve, forwardsolve
```

```
## Loading required package: maps
```

```
## See https://github.com/NCAR/Fields for
##  an extensive vignette, other supplements and source code
```

```
##
## Attaching package: 'fields'
```

```
## The following object is masked from 'package:plotrix':
##
##     color.scale
```

Problem 6

```
full_data <- read.csv("nba-teams-2019.csv")

#trimming data frame to only the 10 columns we desire
keeps <- c("NAME", "TEAM", "W", "L", "PTS", "FGM", "X3PM", "REB", "AST", "TOV", "STL", "BLK")
unscaled.data <- full_data[keeps]
```

```r
num_col <- ncol(unscaled.data)

# scaling all except the first 2 columns (since they're team names and abbreviated team names)
data <- scale(unscaled.data[, 3:num_col])
named.data <- cbind(unscaled.data[, 1:2], data)
```

6.1) Calculating Primary PCA Outputs

Now that the data has been trimmed according to requirements, analysis can be carried on

```r
data.svd <- svd(data)

loading_mat <- data.svd$v

row_names <- c("W", "L", "PTS", "FGM", "X3PM", "REB", "AST", "TOV", "STL", "BLK")
col_names <- c("Comp.1", "Comp.2", "Comp.3", "Comp.4", "Comp.5", "Comp.6", "Comp.7", "Comp.8", "Comp.9"

rownames(loading_mat) <- row_names
colnames(loading_mat) <- col_names

# a) display first 5 loadings
loading_mat[, 1:5]
```

```
##           Comp.1      Comp.2      Comp.3      Comp.4      Comp.5
## W     0.40600381  0.29891549 -0.14260733 -0.02637302 -0.17235753
## L    -0.40600381 -0.29891549  0.14260733  0.02637302  0.17235753
## PTS   0.40724023 -0.01061588  0.08175797  0.05629045  0.46649307
## FGM   0.37362464 -0.16367180  0.20553866 -0.35560648  0.38753089
## X3PM  0.21846557  0.24649676 -0.34906618  0.65429465  0.22225879
## REB   0.31469077  0.07342866  0.53015653  0.21373952 -0.13394941
## AST   0.34648213 -0.30638542  0.07188100 -0.26202158 -0.06485243
## TOV   0.01351429 -0.57979049  0.25239195  0.56156391  0.03579438
## STL   0.10576071 -0.44518308 -0.64437884 -0.09398996  0.19647884
## BLK   0.29445970 -0.31627279 -0.15293786  0.04852648 -0.67923293
```

```r
S <- diag(nrow=10)
diag(S) <- data.svd$d
U <- data.svd$u

pc_scores <- U %*% S

colnames(pc_scores) <- c("PC1", "PC2", "PC3", "PC4", "PC5", "PC6", "PC7", "PC8", "PC9", "PC10")
rownames(pc_scores) <- named.data[, 1]

# b) display first 5 PCs
pc_scores[, 1:5]
```

```
##                              PC1         PC2         PC3         PC4
## Atlanta Hawks          0.22312797 -2.53974191  0.50893819  2.21269787
## Boston Celtics         1.50229613  0.27295771 -1.57309555 -0.76900198
## Brooklyn Nets         -0.17081754  0.85190629  1.11473576  1.66015507
```

```
## Charlotte Hornets      -0.83228821  1.53752969 -0.71457340 -0.52859392
## Chicago Bulls          -3.59341035 -0.47267724  0.34089408 -0.61094419
## Cleveland Cavaliers    -4.97975311  1.47227071  0.86468418 -0.11586246
## Dallas Mavericks       -1.68929896  1.00036865  0.61993032  1.29273628
## Denver Nuggets          1.39186707  0.71579495  0.16158925 -1.04129396
## Detroit Pistons        -1.66627199  1.60376076  0.03596878  0.85670065
## Golden State Warriors   4.20125912 -0.60677459 -0.14540178 -0.04135550
## Houston Rockets         0.40837035  1.93183657 -3.29868766  2.02430885
## Indiana Pacers         -0.04078675 -0.59070844 -1.17673785 -1.65743203
## LA Clippers             0.46936898  0.49353167  1.10460923 -0.20085348
## Los Angeles Lakers      0.54831144 -1.67303628  1.33356780  0.19233952
## Memphis Grizzlies      -2.51248460 -0.88749043 -1.52783335 -0.38701188
## Miami Heat             -0.66999933 -0.45056357  0.10001741  0.75118815
## Milwaukee Bucks         4.07620724  0.74965117  0.51621866  0.70517383
## Minnesota Timberwolves -0.23127206 -0.35370324 -0.36242970 -1.26619776
## New Orleans Pelicans    1.19457679 -1.60750808  1.71444152 -0.56178329
## New York Knicks        -3.97267805 -0.06199543  0.70658518  0.61103071
## Oklahoma City Thunder   1.76807418 -0.43340244 -0.60801576 -0.06501535
## Orlando Magic          -0.23600407  0.84732666  0.36952122 -0.32586076
## Philadelphia 76ers      1.98949116 -0.39658651  1.06529065  0.15256065
## Phoenix Suns           -2.89460784 -3.00036401 -1.19876442 -0.21656802
## Portland Trail Blazers  1.44116557  1.36160685  1.30444813  0.01855138
## Sacramento Kings        0.65581700 -0.17610878 -0.10259882 -0.99106994
## San Antonio Spurs       0.18403696  2.01655534  0.94439164 -1.88517335
## Toronto Raptors         2.23523673  0.30866041 -1.01737035 -0.01347638
## Utah Jazz               1.49078825 -0.77783736 -0.41265866  0.94884896
## Washington Wizards     -0.29032207 -1.13525910 -0.66766471 -0.74879764
##                                PC5
## Atlanta Hawks           0.89751288
## Boston Celtics          0.17175446
## Brooklyn Nets           0.63550810
## Charlotte Hornets      -0.11428099
## Chicago Bulls          -0.02423137
## Cleveland Cavaliers     1.57067664
## Dallas Mavericks       -0.02050681
## Denver Nuggets          0.04589741
## Detroit Pistons        -0.11952612
## Golden State Warriors  -0.31912776
## Houston Rockets         0.74844044
## Indiana Pacers         -0.41147741
## LA Clippers             0.14459966
## Los Angeles Lakers     -0.13598051
## Memphis Grizzlies      -1.70030234
## Miami Heat             -1.48171293
## Milwaukee Bucks        -0.15770516
## Minnesota Timberwolves  0.32489880
## New Orleans Pelicans    0.50111536
## New York Knicks        -1.09985277
## Oklahoma City Thunder   0.51059879
## Orlando Magic          -1.35859975
## Philadelphia 76ers     -0.40585105
## Phoenix Suns            0.27993407
## Portland Trail Blazers -0.11978528
## Sacramento Kings        1.48603962
```

```
## San Antonio Spurs      -0.23819910
## Toronto Raptors         0.09083437
## Utah Jazz              -1.11035558
## Washington Wizards      1.40968435
```

```
# c) eigen values
eigen.values <- ((data.svd$d)^2)/(nrow(data)-1)
eigen.values
```

```
##  [1] 4.574094e+00 1.581170e+00 1.191798e+00 1.000163e+00 6.600746e-01
##  [6] 3.862735e-01 3.543119e-01 1.968092e-01 5.530558e-02 1.831126e-34
```

```
eigen.sum <- sum(eigen.values)
eigen.sum
```

```
## [1] 10
```

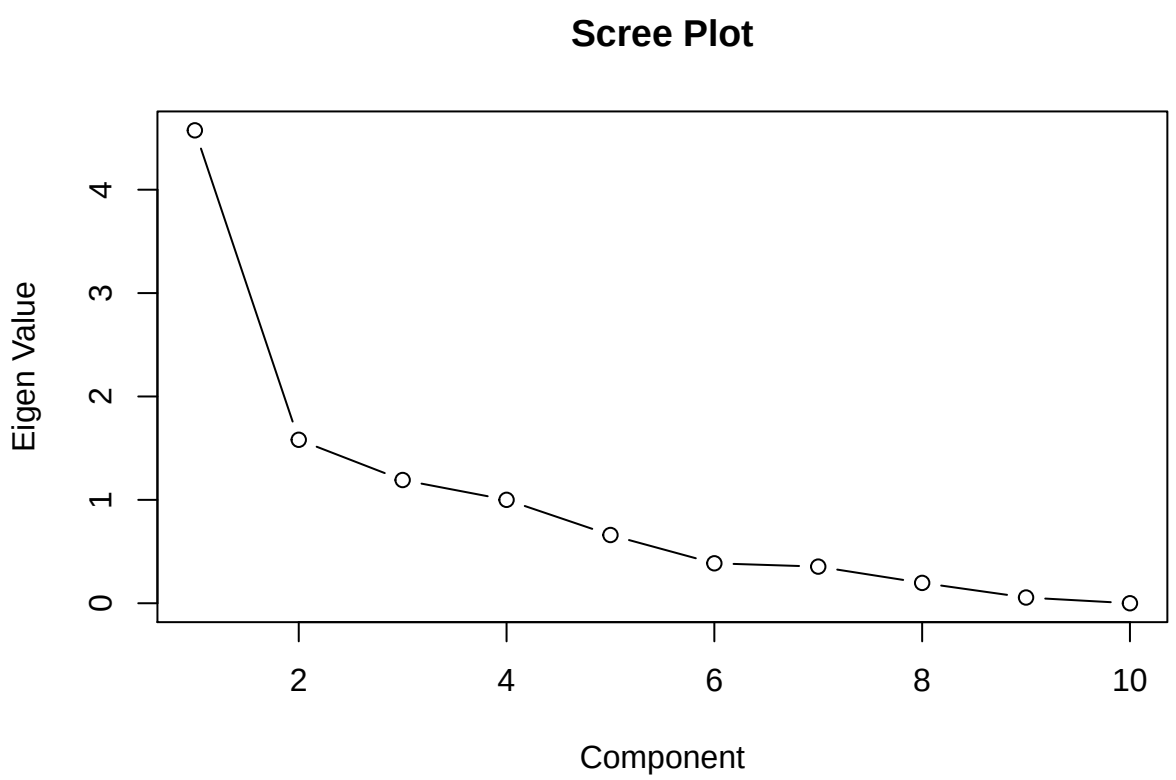
6.2) Eigen Values

```
eigen.percent <- 100 * eigen.values/eigen.sum
eigen.cum <- cumsum(eigen.percent)

# a) su,,ary table of eigen values
eigen.table <- data.frame(eigenvalue = eigen.values, percentage = eigen.percent, 'Cumulative Percentage
print(round(eigen.table, 4), print.gap=2)
```

```
##     eigenvalue  percentage  Cumulative.Percentage
## 1       4.5741     45.7409                45.7409
## 2       1.5812     15.8117                61.5526
## 3       1.1918     11.9180                73.4706
## 4       1.0002     10.0016                83.4723
## 5       0.6601      6.6007                90.0730
## 6       0.3863      3.8627                93.9357
## 7       0.3543      3.5431                97.4789
## 8       0.1968      1.9681                99.4469
## 9       0.0553      0.5531               100.0000
## 10      0.0000      0.0000               100.0000
```

From the table, it can be seen that the eigenvalues with the largest values represent the component that capture the most of the variance present in the original data set.
For instance, the largest value of 4.5741 represents the first Principal Component that captures about 45.7% of the variation present in the data set.
On the other hand, the last eigenvalue (associated with PC10), correct to 4 decimal places, captures 0% (1.83e-32 %) of the variation present in the data - meaning that this PC can be very easily discarded without loss of much information.

```
# b) scree-plot
plot(eigen.values, type='b', main='Scree Plot', xlab='Component', ylab='Eigen Value')
```



In a scree plot, one can see Principal Components on the X axis, and eigenvalues for each of the PCs. A scree plot is usuall a downward sloping line or bar graph representing each of the PCs with their respective eigenvalues.
A scree plot is often used as a diagnostic tool to check whether PCA is a viable tool of getting rid of dimensions. The plot starts with the PCs that capture most of the variation in the data - representing the most important dimension to be conserved.

c)

In the table, the last eigenvalue for PC10 is outputted 0. However, this is because of the 4 decimal place limit I placed on the values while creating the table.
However, this extremely small value (1.83×10^{-32}) or 0 as outputted means that an excruciatingly small value of variance of the full dataset is what is captured by PC10. This hints at the possible discarding of this PC in lieu of dimension-reduction.

d)

Based on the several different criteria discussed to choose how many components should be retained, I decided to retain **3** PCs.
I used the rule that states to include the PCs that capture 70% of the variation in the data. As can be seen in the table, the cumulative percentage for the first 2 PCs is 61.55%, and that of the first 3 PCs is 73.47% implying that the first 3 PCs satisfied this criterion.

6.3) Interpreting PCs

```
# a) matrix of correlations between variables and PCs
pc_cors <- round(cor(data, pc_scores), 4)
pc_cors
```

```
##          PC1     PC2     PC3     PC4     PC5     PC6     PC7     PC8
## W     0.8683  0.3759 -0.1557 -0.0264 -0.1400  0.0321 -0.2251  0.0895
## L    -0.8683 -0.3759  0.1557  0.0264  0.1400 -0.0321  0.2251 -0.0895
## PTS   0.8710 -0.0133  0.0893  0.0563  0.3790  0.0462  0.1687  0.1940
## FGM   0.7991 -0.2058  0.2244 -0.3556  0.3148  0.0358  0.1269 -0.0044
## X3PM  0.4672  0.3100 -0.3811  0.6543  0.1806 -0.1877  0.1536 -0.1364
## REB   0.6730  0.0923  0.5788  0.2138 -0.1088  0.2902 -0.0175 -0.2449
## AST   0.7410 -0.3853  0.0785 -0.2620 -0.0527 -0.4277 -0.1173 -0.1555
## TOV   0.0289 -0.7291  0.2755  0.5616  0.0291 -0.0271 -0.2320  0.1383
## STL   0.2262 -0.5598 -0.7035 -0.0940  0.1596  0.2764 -0.1164 -0.1250
## BLK   0.6298 -0.3977 -0.1670  0.0485 -0.5518  0.0366  0.3215  0.0749
##          PC9 PC10
## W     0.0219    0
## L    -0.0219    0
## PTS  -0.1350    0
## FGM   0.1605    0
## X3PM  0.0470    0
## REB  -0.0388    0
## AST  -0.0635    0
## TOV   0.0423    0
## STL  -0.0263    0
## BLK   0.0115    0
```

It should be noted that the absolute magnitude and the relative signs of each of the correlations of the variables with the Principal Component is of importance. For the case of discussion, an absolute value of **0.65 and above** will be considered a **strong correlation**.

PC1:
The first PC has the strongest correlation with the number of **Wins** (and equally strong a negative correlation with the number of Losses), and thus is understandably can be see with a comparable rise in **Points per Game** and **Rebounds**. It should be noted however that it has almost no correlation with the turnovers per game stat, and a low but postiive corelation with the steals per game stat. Thus, it can be concluded that PC1 can be associated with the likelihood of winning a game and thereby a higher number of points per game scored.

PC2:
PC2 has consdierably smaller correlations with the variables being analysed - which is in line with its importance being lower than PC1. The largest correlations are **negative** in nature and are attributed to the **Turnovers & Steals per Game**. This implies that a higher PC2 value will be attributed to a almost similarly lower Turnover or Steals per game value. This PC has almost no correlation with the number of points scored per game. Thus, it is seen that this *PC has largest variations on variables that were relatively quite under-represented in variation by PC1.*

PC3:
PC3 has even smaller correlations with the variables than there were in PC2 - as is expected. The most correlated variable, is the **Steals per Game** and this is a negative correlation. From this PC3, it can thus

be said that as PC3 increases, one can expect a considerably lower Steals per Game, and a slightly lesser correlated, but **lower 3 points per game** stat. This PC has almost no correlation (positive regardless) with the points per game statistic.

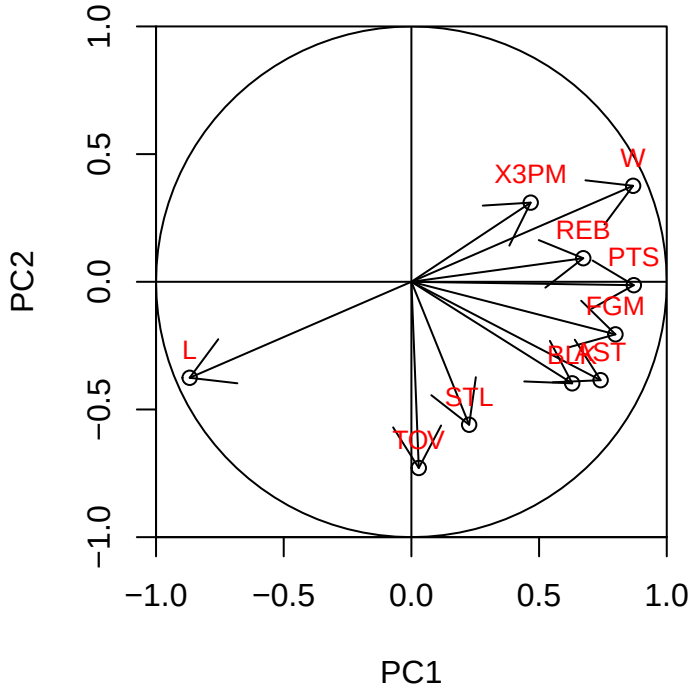
PC4:
PC4 is most heavily correlated with the **3 Pointers per Game** statistic than any other - and this is **positively correlated**. This would mean that higher the PC4 value, greater the number of 3 pointers made per game, and the negative correlation with the field goals made implying that this statistic will be lesser. It should be noted that there is an almost zero, but negative correlation of PC4 with number of **Wins**. This is important to note that this doesn't mean that a higher number of 3 points would mean a lesser or no correlation with games won, but it means that this Principal Component represents a dimension where the 3 points made per game vary the most, and the games won is not represented for most part.

b) Circle of Correlations

```
pc_cors_two <- pc_cors[, 1:2]
pc_cors_two
```

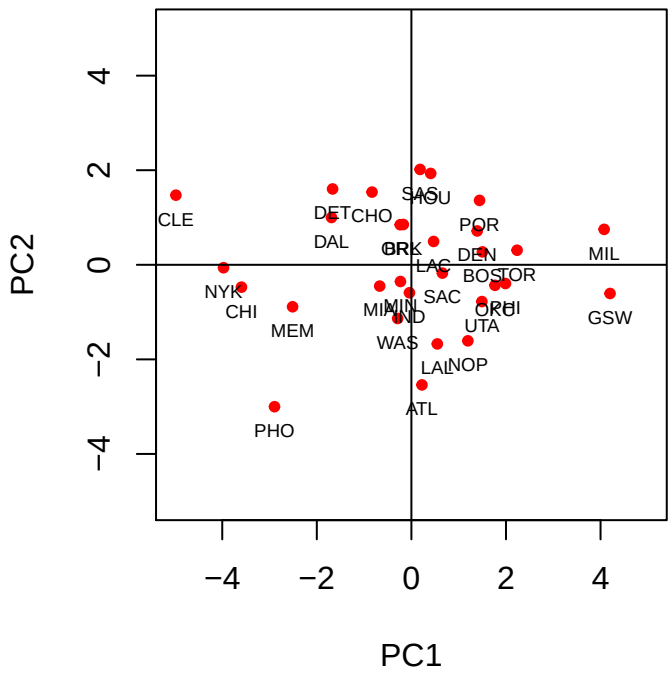
```
##          PC1     PC2
## W     0.8683  0.3759
## L    -0.8683 -0.3759
## PTS   0.8710 -0.0133
## FGM   0.7991 -0.2058
## X3PM  0.4672  0.3100
## REB   0.6730  0.0923
## AST   0.7410 -0.3853
## TOV   0.0289 -0.7291
## STL   0.2262 -0.5598
## BLK   0.6298 -0.3977
```

```
par(pty="s")
plot(pc_cors_two, xlim=c(-1,1), ylim=c(-1,1), xaxs='i', yaxs='i')
draw.circle(0,0, radius=1)
abline(h=0)
abline(v=0)
arrows(rep(0,10), rep(0,10), pc_cors_two[, 1], pc_cors_two[, 2])
text(pc_cors_two, labels=rownames(pc_cors_two), cex=0.8, pos=3, col='red')
```



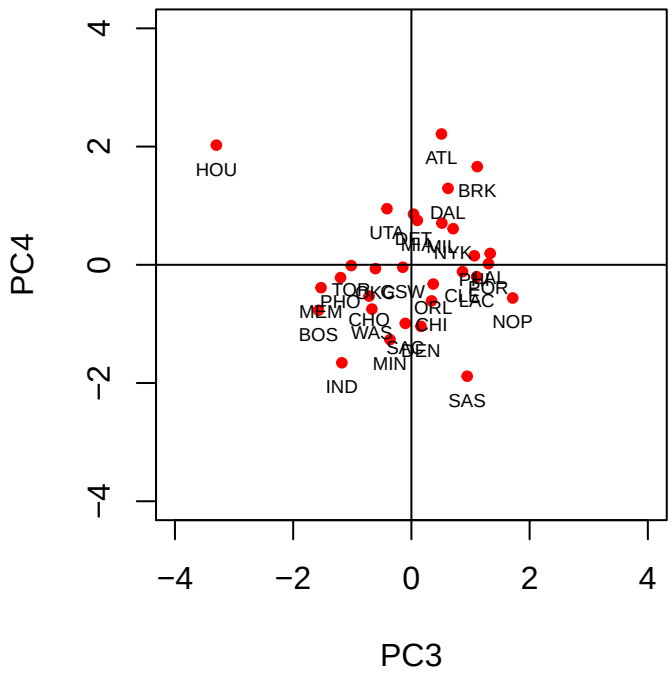
6.4) Cloud of Individuals

```
# a) Scatter Plot of Teams PC1 v/s PC2
pc_scores_two <- pc_scores[, 1:2]

par(pty='s')
plot(pc_scores_two, xlim=c(-5,5), ylim=c(-5,5), pch=20, col='red')
abline(h=0)
abline(v=0)
text(pc_scores_two, labels=named.data[, 2], cex=0.6, pos=1, col='black')
```



The general pattern from this score plot is that most of the teams are centred around the origin of the graph. However, there are a lot of insights to be found from the position of the certain outliers - GSW and MIL for instance. Both teams have an impeccable Wins number, however, their Turnovers and Steals per game mean that they are on different sides of the PC2 score. The negative PC2 score for the Bucks mean that they hinged a bit on their Turnover game (PC2) on track to their wins (PC1). However, both teams did not rely much on it - attributed by a low PC2 value. In addition, a very low PC2 score for the Hawks shed light on their strong Turnover and Steals game, whereas a very low PC1 score sheds light on a bad season dealt by both the Knicks and the Cavaliers.

```
# b) Scatter Plot of Teams PC3 v/s PC4
pc_scores_new <- pc_scores[, 3:4]

par(pty='s')
plot(pc_scores_new, xlim=c(-4,4), ylim=c(-4,4), pch=20, col='red')
abline(h=0)
abline(v=0)
text(pc_scores_new, labels=named.data[, 2], cex=0.6, pos=1, col='black')
```



Most of the data points for the PC3 v/s PC4 score plot are centred around the origin. Several extreme outliers like the Rockets and Spurs shed light on certain characteristics on their game.
A very negative PC3 value, and a very positive PC4 value for the Rockets sheds light on a team's strong Steals per Game statistic, and a strong 3 pointer game respectively. This sheds light on the Rockets fitting this profile. The Spurs and Bulls on the other hand have very low PC4 scores - attributed to their relatively sub-par 3 points per game statistic.

c)

Minnesota Timberwolves:
From the PCA analysis carried out on the given data set, and the PC1 v/s PC2 score plot in particular, it can be said that the Timberwolves have a very mediocre number of wins relative to the whole league - as represented by the almost 0 PC1 score. On the other hand, a negative PC2 value is attributed to the fairly above average Steals and Turnovers per game. In addition, a very low PC4 score (large absolute value) sheds light on the terrible 3 pointers per game record of the Timberwolves in the league.

GSW -vs- NYK:
The Golden State Warriors and the New York Knicks have a widely contrased PC1 v/s PC2 score plot. This can be attributed to GSW's impeccable win record, and the Knicks' terrible one - this is represented by the 2 teams being on either extremes on the plot. In addition, an almost 0 PC2 score for the Knicks highlights the very mediocre (average) steals and turnovers per game compared to the league, whereas Warriors' negative PC2 scores highlights the above-average statistic. The PC3 scores shed light on similar information for both the teams. It should be noted that the PC3 v/s PC4 score plot does not say much about both the teams because they are relatively closely placed on the plot - in addition, they understandably mean not as much as the PC1 and PC2 scores. **Based on these plots, the Warriors could be though of as the favourites in the matchup.**

PHO -vs- DET

The Phoenix Suns data point is an outlier on the PC1 v/s PC2 score plot. The large negative PC2 score sheds light on the fact the exceptional Turnovers and Steals per game statistic. On the other hand, the non-outlier but large negative PC1 score is attributed to the low number of wins. The negative PC1 and positive PC2 score for the Detriot Pistons means that they have both a low Win rate, and a low Steals/Turnover rate compared to the typical in the leauge. Although the PC3 v/s PC4 score plot does not represent as much of a contrast between the teams, it is safe to say that the Suns are more likely to have more steals in the matchup, and lesser 3 pointers than the Pistons. Based on the PCA, it can thus be said that the Pistons would edge the Suns in having a better shot at winning the matchup - however, with a lower Turnovers and Steals rate in the game.