Player_Classify

Chenjie Li

11/11/2019

```
# firts, let's connect R with our Postgres databse:
library(RPostgreSQL)
```

```
## Loading required package: DBI
```

```
# create connection
con <- dbConnect(PostgreSQL(), user= "lchenjie", dbname="csp571")

# query to fetch players' stats

Q = "select a.*,s.*,r.*,i.*
from player_assists a, player_scoring s, player_rebounds r, player_info i
where a.player_name = s.player_name and a.season_name=s.season_name and a.team_name = s.team_name and
s.player_name = r.player_name and s.season_name=r.season_name and s.team_name = r.team_name and
r.player_name = i.player_name; "

# return results
player_raw <- dbGetQuery(con,Q)
```

```
# remove duplicate cols
players <- player_raw[, !duplicated(colnames(player_raw))]
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(zoo)
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
players_18 = players[(players$season_name=='2018-19'|players$season_name=='2017-18'|players$season_name=
players_18 = players_18[players_18$gamesplayed>40,]
players_18 = players_18[players_18$minutes/players_18$gamesplayed > 10,]
sapply(players_18, class)
```

```
##                season_name                season_type
##                "character"                "character"
##                player_name                  team_name
```

```
##               "character"               "character"
##             gamesplayed                   minutes
##               "numeric"                 "numeric"
##                 assists              assistpoints
##               "numeric"                 "numeric"
##              assists2pt               assists3pt
##               "numeric"                 "numeric"
##            atrimassists      shortmidrangeassists
##               "numeric"                 "numeric"
##      longmidrangeassists           corner3assists
##               "numeric"                 "numeric"
##            arc3assists                   offposs
##               "numeric"                 "numeric"
##                  points                      fg2m
##               "numeric"                 "numeric"
##                    fg2a                    fg2pct
##               "numeric"                 "numeric"
##                    fg3m                      fg3a
##               "numeric"                 "numeric"
##                  fg3pct           nonheavefg3pct
##               "numeric"                 "numeric"
##                ftsmade            ptsassisted2s
##               "numeric"                 "numeric"
##          ptsunassisted2s            ptsassisted3s
##               "numeric"                 "numeric"
##          ptsunassisted3s            assisted2spct
##               "numeric"                 "numeric"
##  nonputbacksassisted2spct            assisted3spct
##               "numeric"                 "numeric"
##                 fg3apct           shotqualityavg
##               "numeric"                 "numeric"
##                  efgpct                    tspct
##               "numeric"                 "numeric"
##              ptsputbacks              fg2ablocked
##               "numeric"                 "numeric"
##           fg2apctblocked              fg3ablocked
##               "numeric"                 "numeric"
##           fg3apctblocked                   usage
##               "numeric"                 "numeric"
##                rebounds              defrebounds
##               "numeric"                 "numeric"
##            ftdefrebounds          defftreboundpct
##               "numeric"                 "numeric"
##           def2ptrebounds       def2ptreboundpct
##               "numeric"                 "numeric"
##           def3ptrebounds       def3ptreboundpct
##               "numeric"                 "numeric"
##          deffgreboundpct              offrebounds
##               "numeric"                 "numeric"
##            ftoffrebounds          offftreboundpct
##               "numeric"                 "numeric"
##           off2ptrebounds       off2ptreboundpct
##               "numeric"                 "numeric"
##           off3ptrebounds       off3ptreboundpct
```

```
##                  "numeric"                  "numeric"
##            offfgreboundpct         defatrimreboundpct
##                  "numeric"                  "numeric"
## defshortmidrangereboundpct  deflongmidrangereboundpct
##                  "numeric"                  "numeric"
##          defarc3reboundpct       defcorner3reboundpct
##                  "numeric"                  "numeric"
##         offatrimreboundpct offshortmidrangereboundpct
##                  "numeric"                  "numeric"
##  offlongmidrangereboundpct          offarc3reboundpct
##                  "numeric"                  "numeric"
##       offcorner3reboundpct                   position
##                  "numeric"                "character"
##                     height                     weight
##                "character"                  "integer"
```

```
# ---------------- Data Cleaning and Transformation--------------------------#

# convert height to meters
players_18$height <- sapply(strsplit(as.character(players_18$height),"-"),
        function(x){0.3048*(as.numeric(x[1]) + 0.1*as.numeric(x[2]))})

# replacing NA's with group mean's (e.g.: G = 1.9, F=2.05 etc)
players_18$height <-na.aggregate(players_18$height,by=players_18$position)
players_18$weight <-na.aggregate(players_18$weight,by=players_18$position)
```

```
# get numerical data to perform correlation analysis
num_players_18 <- players_18[, sapply(players_18, class) != "character"]
```

Find heavily correlated cols and remove some

```
# define a function to extract correlated pairs
cor_extract <-function(df,thre){
  cor_mat <- cor(df)
  for (i in 1:nrow(cor_mat)){
  correlations <-  which((cor_mat[i,] > thre) & (cor_mat[i,] != 1))

  if(length(correlations)> 0){
    print(colnames(df)[i])
    print(correlations)
  }
}
}

cor_extract(num_players_18,0.8)
```

```
## [1] "minutes"
## offposs
##      12
## [1] "assists"
##         assistpoints           assists2pt           assists3pt
##                    4                    5                    6
##         atrimassists shortmidrangeassists       corner3assists
##                    7                    8                   10
```

```
##          arc3assists
##                   11
## [1] "assistpoints"
##              assists           assists2pt           assists3pt
##                    3                    5                    6
##         atrimassists shortmidrangeassists       corner3assists
##                    7                    8                   10
##          arc3assists
##                   11
## [1] "assists2pt"
##              assists         assistpoints           assists3pt
##                    3                    4                    6
##         atrimassists shortmidrangeassists          arc3assists
##                    7                    8                   11
## [1] "assists3pt"
##        assists   assistpoints     assists2pt   atrimassists corner3assists
##              3              4              5              7             10
##    arc3assists
##             11
## [1] "atrimassists"
##              assists         assistpoints           assists2pt
##                    3                    4                    5
##           assists3pt shortmidrangeassists       corner3assists
##                    6                    8                   10
##          arc3assists
##                   11
## [1] "shortmidrangeassists"
##      assists assistpoints   assists2pt atrimassists
##            3            4            5            7
## [1] "corner3assists"
##      assists assistpoints   assists3pt atrimassists  arc3assists
##            3            4            6            7           11
## [1] "arc3assists"
##        assists   assistpoints     assists2pt     assists3pt   atrimassists
##              3              4              5              6              7
## corner3assists
##             10
## [1] "offposs"
## minutes
##       2
## [1] "points"
##            fg2m            fg2a         ftsmade ptsunassisted2s           usage
##              14              15              21              23              38
## [1] "fg2m"
##          points            fg2a ptsunassisted2s
##              13              15              23
## [1] "fg2a"
##          points            fg2m ptsunassisted2s
##              13              14              23
## [1] "fg3m"
##          fg3a ptsassisted3s
##            18            24
## [1] "fg3a"
##          fg3m ptsassisted3s
```

```
##            17            24
## [1] "fg3pct"
## nonheavefg3pct
##             20
## [1] "nonheavefg3pct"
## fg3pct
##     19
## [1] "ftsmade"
## points
##     13
## [1] "ptsunassisted2s"
## points   fg2m   fg2a
##     13     14     15
## [1] "ptsassisted3s"
## fg3m fg3a
##   17   18
## [1] "assisted2spct"
## nonputbacksassisted2spct
##                       27
## [1] "nonputbacksassisted2spct"
## assisted2spct
##            26
## [1] "efgpct"
## tspct
##    32
## [1] "tspct"
## efgpct
##     31
## [1] "ptsputbacks"
##                    rebounds                offrebounds
##                          39                         48
##             off2ptrebounds           off2ptreboundpct
##                          51                         52
##            offfgreboundpct         offatrimreboundpct
##                          55                         61
## offshortmidrangereboundpct
##                          62
## [1] "usage"
## points
##     13
## [1] "rebounds"
##      ptsputbacks      defrebounds    ftdefrebounds   def2ptrebounds
##               33               40               41               43
## def2ptreboundpct   def3ptrebounds  deffgreboundpct      offrebounds
##               44               45               47               48
##   off2ptrebounds   off3ptrebounds
##               51               53
## [1] "defrebounds"
##       rebounds  ftdefrebounds def2ptrebounds def3ptrebounds
##             39             41             43             45
## [1] "ftdefrebounds"
##        rebounds     defrebounds defftreboundpct
##              39              40              42
## [1] "defftreboundpct"
```

```
##    ftdefrebounds def2ptreboundpct  deffgreboundpct
##               41               44               47
## [1] "def2ptrebounds"
##       rebounds    defrebounds def3ptrebounds
##             39             40             45
## [1] "def2ptreboundpct"
##                   rebounds            defftreboundpct
##                         39                         42
##           def3ptreboundpct            deffgreboundpct
##                         46                         47
##         defatrimreboundpct defshortmidrangereboundpct
##                         56                         57
##  deflongmidrangereboundpct          defarc3reboundpct
##                         58                         59
## [1] "def3ptrebounds"
##       rebounds    defrebounds def2ptrebounds
##             39             40             43
## [1] "def3ptreboundpct"
##           def2ptreboundpct            deffgreboundpct
##                         44                         47
## defshortmidrangereboundpct  deflongmidrangereboundpct
##                         57                         58
##          defarc3reboundpct       defcorner3reboundpct
##                         59                         60
## [1] "deffgreboundpct"
##                   rebounds            defftreboundpct
##                         39                         42
##           def2ptreboundpct           def3ptreboundpct
##                         44                         46
##         defatrimreboundpct defshortmidrangereboundpct
##                         56                         57
##  deflongmidrangereboundpct          defarc3reboundpct
##                         58                         59
## [1] "offrebounds"
##                ptsputbacks                    rebounds
##                         33                         39
##             off2ptrebounds           off2ptreboundpct
##                         51                         52
##             off3ptrebounds           off3ptreboundpct
##                         53                         54
##            offfgreboundpct         offatrimreboundpct
##                         55                         61
## offshortmidrangereboundpct  offlongmidrangereboundpct
##                         62                         63
##          offarc3reboundpct
##                         64
## [1] "ftoffrebounds"
## offfftreboundpct
##              50
## [1] "offfftreboundpct"
## ftoffrebounds
##            49
## [1] "off2ptrebounds"
##                ptsputbacks                    rebounds
```

```
##                         33                         39
##               offrebounds           off2ptreboundpct
##                         48                         52
##            off3ptrebounds            offfgreboundpct
##                         53                         55
##        offatrimreboundpct offshortmidrangereboundpct
##                         61                         62
##  offlongmidrangereboundpct
##                         63
## [1] "off2ptreboundpct"
##                ptsputbacks               offrebounds
##                         33                         48
##             off2ptrebounds           off3ptreboundpct
##                         51                         54
##            offfgreboundpct         offatrimreboundpct
##                         55                         61
## offshortmidrangereboundpct  offlongmidrangereboundpct
##                         62                         63
##          offarc3reboundpct
##                         64
## [1] "off3ptrebounds"
##          rebounds       offrebounds    off2ptrebounds  off3ptreboundpct
##                39                48                51                54
## offarc3reboundpct
##                64
## [1] "off3ptreboundpct"
##                offrebounds           off2ptreboundpct
##                         48                         52
##             off3ptrebounds            offfgreboundpct
##                         53                         55
##         offatrimreboundpct offshortmidrangereboundpct
##                         61                         62
##  offlongmidrangereboundpct          offarc3reboundpct
##                         63                         64
##       offcorner3reboundpct
##                         65
## [1] "offfgreboundpct"
##                ptsputbacks               offrebounds
##                         33                         48
##             off2ptrebounds           off2ptreboundpct
##                         51                         52
##           off3ptreboundpct         offatrimreboundpct
##                         54                         61
## offshortmidrangereboundpct  offlongmidrangereboundpct
##                         62                         63
##          offarc3reboundpct       offcorner3reboundpct
##                         64                         65
## [1] "defatrimreboundpct"
##           def2ptreboundpct            deffgreboundpct
##                         44                         47
## defshortmidrangereboundpct
##                         57
## [1] "defshortmidrangereboundpct"
##   def2ptreboundpct   def3ptreboundpct    deffgreboundpct defatrimreboundpct
```

```
##                 44                 46                 47                 56
##  defarc3reboundpct
##                 59
## [1] "deflongmidrangereboundpct"
##  def2ptreboundpct  def3ptreboundpct   deffgreboundpct defarc3reboundpct
##                44                46                47                59
## [1] "defarc3reboundpct"
##           def2ptreboundpct           def3ptreboundpct
##                         44                         46
##            deffgreboundpct defshortmidrangereboundpct
##                         47                         57
##  deflongmidrangereboundpct
##                         58
## [1] "defcorner3reboundpct"
## def3ptreboundpct
##               46
## [1] "offatrimreboundpct"
##                ptsputbacks                offrebounds
##                         33                         48
##             off2ptrebounds           off2ptreboundpct
##                         51                         52
##           off3ptreboundpct            offfgreboundpct
##                         54                         55
## offshortmidrangereboundpct  offlongmidrangereboundpct
##                         62                         63
##          offarc3reboundpct
##                         64
## [1] "offshortmidrangereboundpct"
##               ptsputbacks               offrebounds             off2ptrebounds
##                        33                        48                         51
##          off2ptreboundpct          off3ptreboundpct           offfgreboundpct
##                        52                        54                         55
##        offatrimreboundpct offlongmidrangereboundpct         offarc3reboundpct
##                        61                        63                         64
## [1] "offlongmidrangereboundpct"
##                offrebounds             off2ptrebounds
##                         48                         51
##           off2ptreboundpct           off3ptreboundpct
##                         52                         54
##            offfgreboundpct         offatrimreboundpct
##                         55                         61
## offshortmidrangereboundpct         offarc3reboundpct
##                         62                         64
## [1] "offarc3reboundpct"
##                offrebounds           off2ptreboundpct
##                         48                         52
##             off3ptrebounds           off3ptreboundpct
##                         53                         54
##            offfgreboundpct         offatrimreboundpct
##                         55                         61
## offshortmidrangereboundpct  offlongmidrangereboundpct
##                         62                         63
## [1] "offcorner3reboundpct"
## off3ptreboundpct  offfgreboundpct
```

```
##               54               55
## [1] "height"
## weight
##     67
## [1] "weight"
## height
##     66
```

```
# based on the observations of the results above, delete some highly correlated cols

excluding_cols = c("points","minutes","offposs","assists","assistpoints","assists2pt","assists3pt","fg2
subset_players_18 <- players_18[ , -which(names(players_18) %in% excluding_cols)]

subset_num_players_18 <- subset_players_18[, sapply(subset_players_18, class) != "character"]
cor_extract(subset_num_players_18,0.8)
```

```
## [1] "atrimassists"
## shortmidrangeassists       corner3assists          arc3assists
##                    3                    5                    6
## [1] "shortmidrangeassists"
## atrimassists
##            2
## [1] "corner3assists"
## atrimassists  arc3assists
##            2            6
## [1] "arc3assists"
##   atrimassists corner3assists
##              2              5
## [1] "fg3pct"
## nonheavefg3pct
##              9
## [1] "nonheavefg3pct"
## fg3pct
##      8
## [1] "ptsputbacks"
##           off2ptreboundpct         offatrimreboundpct
##                         30                         37
## offshortmidrangereboundpct
##                         38
## [1] "defftreboundpct"
## def2ptreboundpct
##               27
## [1] "def2ptreboundpct"
##            defftreboundpct           def3ptreboundpct
##                         26                         28
##         defatrimreboundpct defshortmidrangereboundpct
##                         32                         33
##  deflongmidrangereboundpct          defarc3reboundpct
##                         34                         35
## [1] "def3ptreboundpct"
##           def2ptreboundpct defshortmidrangereboundpct
##                         27                         33
##  deflongmidrangereboundpct          defarc3reboundpct
##                         34                         35
##       defcorner3reboundpct
```

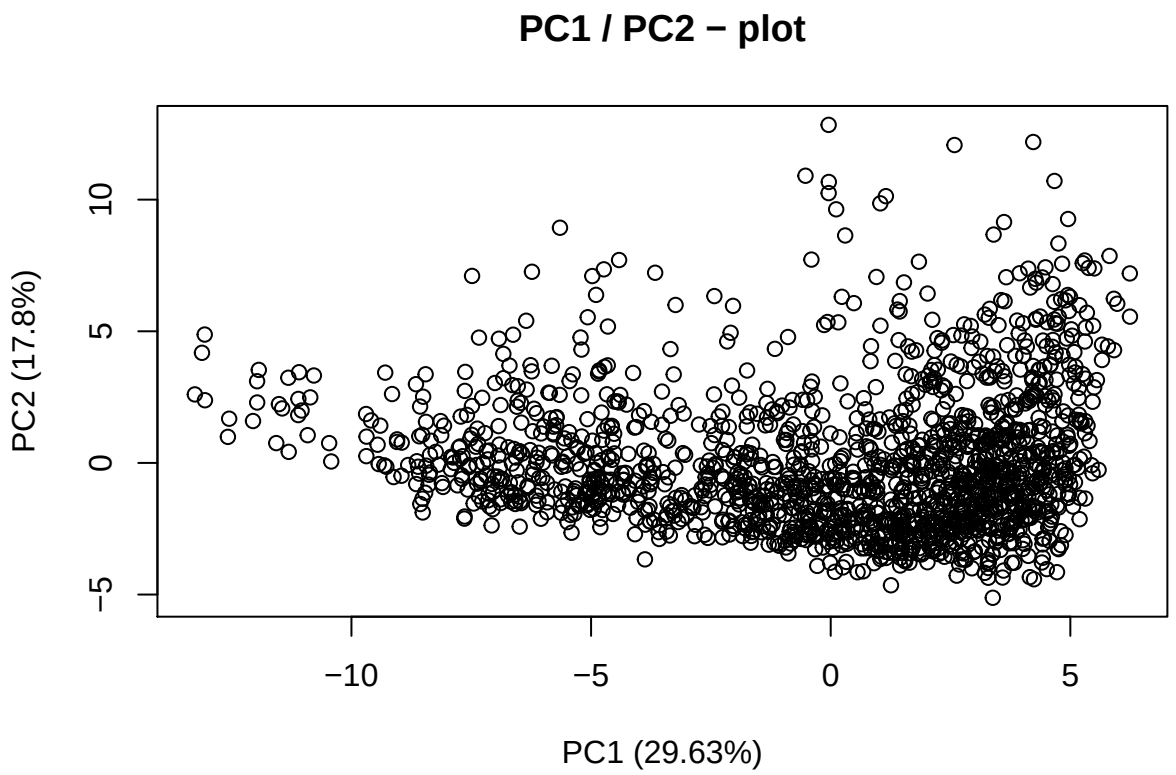
```
##                          36
## [1] "off2ptreboundpct"
##                ptsputbacks           off3ptreboundpct
##                          20                         31
##         offatrimreboundpct offshortmidrangereboundpct
##                          37                         38
##  offlongmidrangereboundpct          offarc3reboundpct
##                          39                         40
## [1] "off3ptreboundpct"
##           off2ptreboundpct         offatrimreboundpct
##                          30                         37
## offshortmidrangereboundpct  offlongmidrangereboundpct
##                          38                         39
##          offarc3reboundpct       offcorner3reboundpct
##                          40                         41
## [1] "defatrimreboundpct"
##           def2ptreboundpct defshortmidrangereboundpct
##                          27                         33
## [1] "defshortmidrangereboundpct"
##   def2ptreboundpct   def3ptreboundpct defatrimreboundpct  defarc3reboundpct
##                27                 28                 32                 35
## [1] "deflongmidrangereboundpct"
##  def2ptreboundpct  def3ptreboundpct defarc3reboundpct
##                27                 28                 35
## [1] "defarc3reboundpct"
##           def2ptreboundpct           def3ptreboundpct
##                          27                         28
## defshortmidrangereboundpct  deflongmidrangereboundpct
##                          33                         34
## [1] "defcorner3reboundpct"
## def3ptreboundpct
##               28
## [1] "offatrimreboundpct"
##                ptsputbacks           off2ptreboundpct
##                          20                         30
##           off3ptreboundpct offshortmidrangereboundpct
##                          31                         38
##  offlongmidrangereboundpct          offarc3reboundpct
##                          39                         40
## [1] "offshortmidrangereboundpct"
##                ptsputbacks          off2ptreboundpct          off3ptreboundpct
##                          20                        30                        31
##        offatrimreboundpct offlongmidrangereboundpct         offarc3reboundpct
##                          37                        39                        40
## [1] "offlongmidrangereboundpct"
##           off2ptreboundpct           off3ptreboundpct
##                          30                         31
##         offatrimreboundpct offshortmidrangereboundpct
##                          37                         38
##          offarc3reboundpct
##                          40
## [1] "offarc3reboundpct"
##           off2ptreboundpct           off3ptreboundpct
##                          30                         31
```

```
##         offatrimreboundpct offshortmidrangereboundpct
##                         37                         38
##  offlongmidrangereboundpct
##                         39
## [1] "offcorner3reboundpct"
## off3ptreboundpct
##               31
## [1] "height"
## weight
##     43
## [1] "weight"
## height
##     42
```

```
#PCA for the data from season 2018-19
# players_18$position <- as.factor(players_18$position)

for(i in 1:ncol(subset_num_players_18)){
  subset_num_players_18[is.na(subset_num_players_18[,i]), i] <- mean(subset_num_players_18[,i], na.rm =
}

players_18.pr <-prcomp(subset_num_players_18, center = TRUE, scale = TRUE)

summary(players_18.pr)
```

```
## Importance of components:
##                           PC1    PC2     PC3     PC4     PC5     PC6     PC7
## Standard deviation     4.0849 2.6201 1.71177 1.42274 1.26160 1.22080 1.11968
## Proportion of Variance 0.3881 0.1596 0.06814 0.04707 0.03702 0.03466 0.02916
## Cumulative Proportion  0.3881 0.5477 0.61585 0.66293 0.69994 0.73460 0.76376
##                            PC8     PC9   PC10    PC11    PC12    PC13   PC14
## Standard deviation     1.01005 0.96793 0.9456 0.86713 0.78340 0.74897 0.6939
## Proportion of Variance 0.02373 0.02179 0.0208 0.01749 0.01427 0.01305 0.0112
## Cumulative Proportion  0.78748 0.80927 0.8301 0.84755 0.86183 0.87487 0.8861
##                           PC15    PC16    PC17    PC18    PC19    PC20   PC21
## Standard deviation     0.66465 0.64646 0.60323 0.59476 0.56971 0.52791 0.4951
## Proportion of Variance 0.01027 0.00972 0.00846 0.00823 0.00755 0.00648 0.0057
## Cumulative Proportion  0.89634 0.90606 0.91452 0.92275 0.93030 0.93678 0.9425
##                           PC22    PC23    PC24    PC25    PC26    PC27    PC28
## Standard deviation     0.48294 0.48239 0.47201 0.44988 0.43786 0.42892 0.41268
## Proportion of Variance 0.00542 0.00541 0.00518 0.00471 0.00446 0.00428 0.00396
## Cumulative Proportion  0.94790 0.95331 0.95849 0.96320 0.96766 0.97194 0.97590
##                           PC29    PC30    PC31    PC32    PC33    PC34    PC35
## Standard deviation     0.40375 0.37260 0.34429 0.33512 0.33375 0.30583 0.29428
## Proportion of Variance 0.00379 0.00323 0.00276 0.00261 0.00259 0.00218 0.00201
## Cumulative Proportion  0.97969 0.98292 0.98568 0.98829 0.99088 0.99305 0.99507
##                           PC36    PC37    PC38    PC39    PC40    PC41    PC42
## Standard deviation     0.2862 0.23009 0.21089 0.16494 0.05216 0.03603 0.03387
## Proportion of Variance 0.0019 0.00123 0.00103 0.00063 0.00006 0.00003 0.00003
## Cumulative Proportion  0.9970 0.99820 0.99924 0.99987 0.99993 0.99996 0.99999
##                           PC43
## Standard deviation     0.02112
## Proportion of Variance 0.00001
## Cumulative Proportion  1.00000
```

```
plot(players_18.pr$x[,1],players_18.pr$x[,2], xlab="PC1 (29.63%)", ylab = "PC2 (17.8%)", main = "PC1 / P
```




```
library("factoextra")
```

```
## Loading required package: ggplot2
```

```
## Welcome! Related Books: `Practical Guide To Cluster Analysis in R` at https://goo.gl/13EFCZ
```

```
jpeg("/home/chenjie/Desktop/CSP571/Clustering/Figs/2018-19_pca.jpg")
fviz_pca_ind(players_18.pr, geom.ind = "point", pointshape = 21,
             pointsize = 2,
             fill.ind = players_18$position,
             col.ind = "black",
             palette = "jco",
             addEllipses = TRUE,
             label = "var",
             col.var = "black",
             repel = TRUE,
             legend.title = "Diagnosis") +
  ggtitle("2D PCA-plot from 66 feature dataset") +
  theme(plot.title = element_text(hjust = 0.5))
dev.off()
```

```
## pdf
##   2
```

```
set.seed(123)
# LDAfor the data from season 2018-19
library(caret)
```

```
## Loading required package: lattice
```

```r
# split traning/testing data 4:1
training_18_index <- createDataPartition(subset_players_18$position,p=0.8,list=FALSE)
train_18 <- subset_players_18[training_18_index,]
test_18 <- subset_players_18[-training_18_index,]
train_num_18 <- train_18[, sapply(train_18, class) != "character"]
test_num_18 <- test_18[,sapply(test_18, class) != "character"]

char_names <-names(train_18[, sapply(train_18, class) != "character"])
```

```r
# run model
library(MASS)
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```r
f <- paste("position ~", paste(char_names, collapse=" + "))
lda_18 <- lda(as.formula(paste(f)), data = train_18)
```

```r
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
## Attaching package: 'pROC'
```

```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```r
library(e1071)
# lda_18.predict
lda_18.predict <- predict(lda_18, newdata = test_18)
```

```r
# Confusion Matrix
confusionMatrix(table(lda_18.predict$class,test_18$position),mode = "everything")
```

```
## Confusion Matrix and Statistics
##
##
##          C C-F   F F-G   G
##   C     20   5   3   0   0
##   C-F    9  30   9   2   0
##   F      2  13  64  15   1
##   F-G    0   0   7  23  15
##   G      0   0   0   5 107
##
## Overall Statistics
##
##                Accuracy : 0.7394
##                  95% CI : (0.6885, 0.7859)
##     No Information Rate : 0.3727
##     P-Value [Acc > NIR] : < 2.2e-16
##
```

```
##                   Kappa : 0.6537
##
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: C Class: C-F Class: F Class: F-G Class: G
## Sensitivity           0.64516    0.62500   0.7711     0.5111   0.8699
## Specificity           0.97324    0.92908   0.8745     0.9228   0.9758
## Pos Pred Value        0.71429    0.60000   0.6737     0.5111   0.9554
## Neg Pred Value        0.96358    0.93571   0.9191     0.9228   0.9266
## Precision             0.71429    0.60000   0.6737     0.5111   0.9554
## Recall                0.64516    0.62500   0.7711     0.5111   0.8699
## F1                    0.67797    0.61224   0.7191     0.5111   0.9106
## Prevalence            0.09394    0.14545   0.2515     0.1364   0.3727
## Detection Rate        0.06061    0.09091   0.1939     0.0697   0.3242
## Detection Prevalence  0.08485    0.15152   0.2879     0.1364   0.3394
## Balanced Accuracy     0.80920    0.77704   0.8228     0.7170   0.9229
```

```
# visualize roc for each class
roc.multi <- multiclass.roc(predictor=lda_18.predict$posterior[,1], response=test_18$position)
```

```
## Setting direction: controls > cases
## Setting direction: controls > cases
## Setting direction: controls > cases
## Setting direction: controls > cases
## Setting direction: controls > cases
## Setting direction: controls > cases
## Setting direction: controls > cases
## Setting direction: controls > cases
## Setting direction: controls > cases
## Setting direction: controls > cases
```

```
cat("the AUC value for LDA with raw variables is ",roc.multi$auc)
```

```
## the AUC value for LDA with raw variables is  0.943386
```

```
# now trying LDA using PCA variables
pca_df <- players_18.pr$x[,1:18]
pca_df <- cbind(pca_df, players_18$position)
pca_df <- as.data.frame(pca_df)
```

```
set.seed(123)
colnames(pca_df)[19] <- "position"
pca_train_index <- createDataPartition(pca_df$position, p=0.8, list = FALSE)

cols = c(seq(1,18,by=1));
pca_df[,cols] = apply(pca_df[,cols], 2, function(x) as.numeric(as.character(x)));

pca_train_df <- pca_df[pca_train_index,]
pca_test_df <- pca_df[-pca_train_index,]

pca_lda <- lda(position~., data = pca_train_df)
```

```
pca_lda.predict <- predict(pca_lda, newdata = pca_test_df)
confusionMatrix(table(pca_lda.predict$class,pca_test_df$position),mode = "everything")
```

```
## Confusion Matrix and Statistics
##
##
##          C C-F   F F-G   G
##   C     20   6   3   0   0
##   C-F   11  28  11   0   0
##   F      0  14  60  18   1
##   F-G    0   0   9  23  14
##   G      0   0   0   4 108
##
## Overall Statistics
##
##                Accuracy : 0.7242
##                  95% CI : (0.6726, 0.7718)
##     No Information Rate : 0.3727
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.6339
##
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: C Class: C-F Class: F Class: F-G Class: G
## Sensitivity           0.64516    0.58333   0.7229     0.5111   0.8780
## Specificity           0.96990    0.92199   0.8664     0.9193   0.9807
## Pos Pred Value        0.68966    0.56000   0.6452     0.5000   0.9643
## Neg Pred Value        0.96346    0.92857   0.9030     0.9225   0.9312
## Precision             0.68966    0.56000   0.6452     0.5000   0.9643
## Recall                0.64516    0.58333   0.7229     0.5111   0.8780
## F1                    0.66667    0.57143   0.6818     0.5055   0.9191
## Prevalence            0.09394    0.14545   0.2515     0.1364   0.3727
## Detection Rate        0.06061    0.08485   0.1818     0.0697   0.3273
## Detection Prevalence  0.08788    0.15152   0.2818     0.1394   0.3394
## Balanced Accuracy     0.80753    0.75266   0.7946     0.7152   0.9294
```

```
library(pROC)
pca_roc.multi <- multiclass.roc(predictor=pca_lda.predict$posterior[,1], response=pca_test_df$position)
```

```
## Setting direction: controls > cases
## Setting direction: controls > cases
## Setting direction: controls > cases
## Setting direction: controls > cases
## Setting direction: controls > cases
## Setting direction: controls > cases
## Setting direction: controls > cases
## Setting direction: controls > cases
## Setting direction: controls > cases
## Setting direction: controls > cases
```

```
cat("the AUC value for LDA with raw variables is ",pca_roc.multi$auc)
```

```
## the AUC value for LDA with raw variables is  0.9445637
```

```
# cross validation to compare 2 models (LDA_all_variables VS PCA_LDA)
library(caret)
```

```r
# LDA_all_variables

set.seed(1234)


ctrl <- trainControl(method = "cv",
                     number = 10,
                     returnResamp = "all")

boot_mod <- train(as.formula(paste(f)), data = train_18,
         method = "lda",
         trControl = ctrl)

boot_mod$results$Accuracy
```

```
## [1] 0.7395254
```

```r
library(caret)

# PCA_LDA

set.seed(1234)


ctrl <- trainControl(method = "cv",
                     number = 10,
                     returnResamp = "all")

boot_mod <- train(position~., data = pca_train_df,
         method = "lda",
         trControl = ctrl)

boot_mod
```

```
## Linear Discriminant Analysis
##
## 1332 samples
##   18 predictor
##    5 classes: 'C', 'C-F', 'F', 'F-G', 'G'
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 1199, 1199, 1199, 1198, 1197, 1199, ...
## Resampling results:
##
##   Accuracy   Kappa
##   0.7380845  0.6521624
```

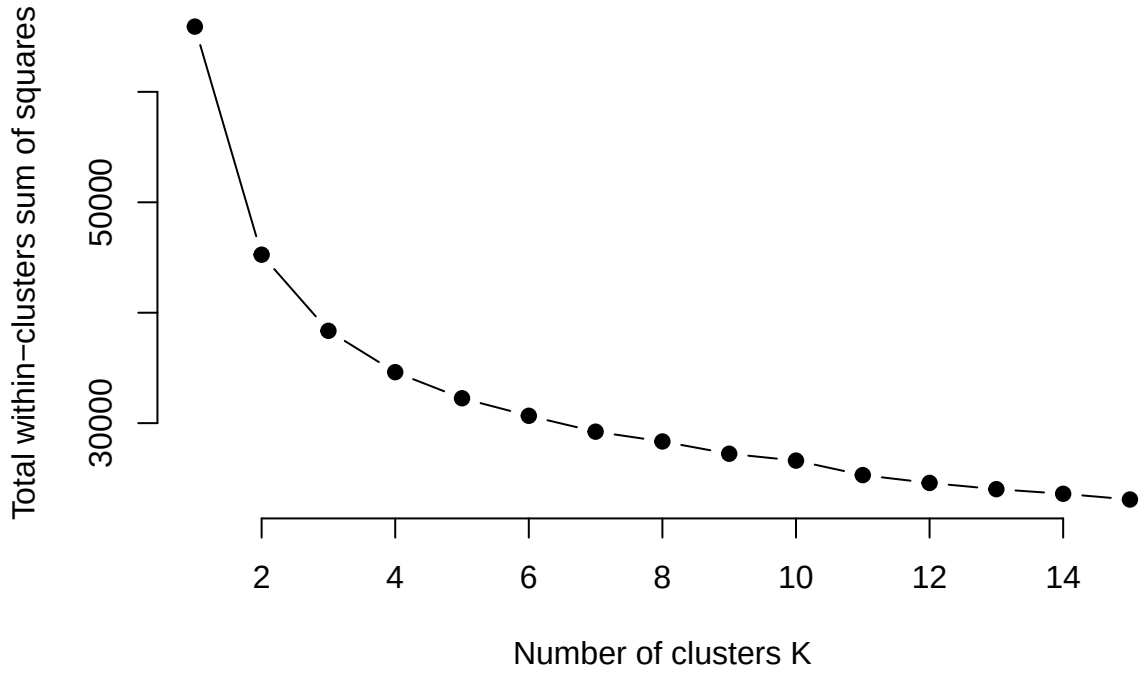
from the results above, we could conclude that using "PCA_LDA" model is slightly better, thus we will use this model to preceed to do k-means clustering.

```r
library(cluster)
#Elbow Method for finding the optimal number of clusters
set.seed(123)
# Compute and plot wss for k = 2 to k = 15.
```

```r
k.max <- 15
# drop position
pca_df_num <- pca_df [ , !(names(pca_df) %in% "position")]
data <- pca_df_num
wss <- sapply(1:k.max,
              function(k){kmeans(data, k, nstart=20,iter.max = 15 )$tot.withinss})
wss
```

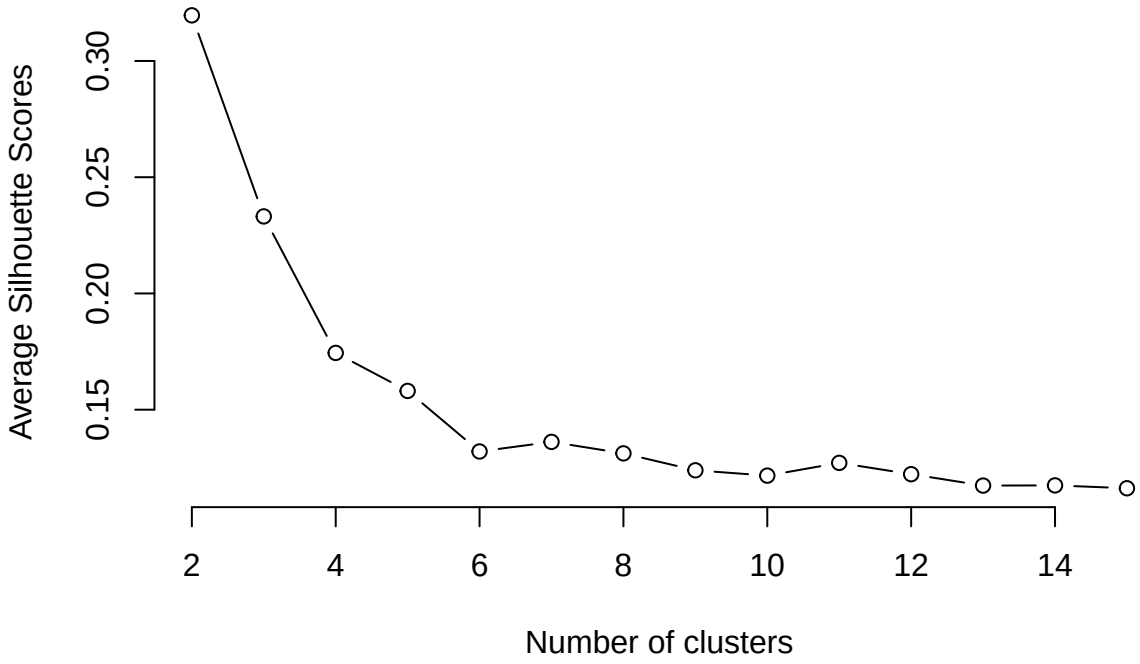
```
##  [1] 65905.45 45255.19 38360.89 34617.91 32253.10 30667.24 29229.32 28342.37
##  [9] 27230.89 26609.86 25297.38 24586.77 24023.43 23605.44 23086.33
```

```r
plot(1:k.max, wss,
     type="b", pch = 19, frame = FALSE,
     xlab="Number of clusters K",
     ylab="Total within-clusters sum of squares")
```

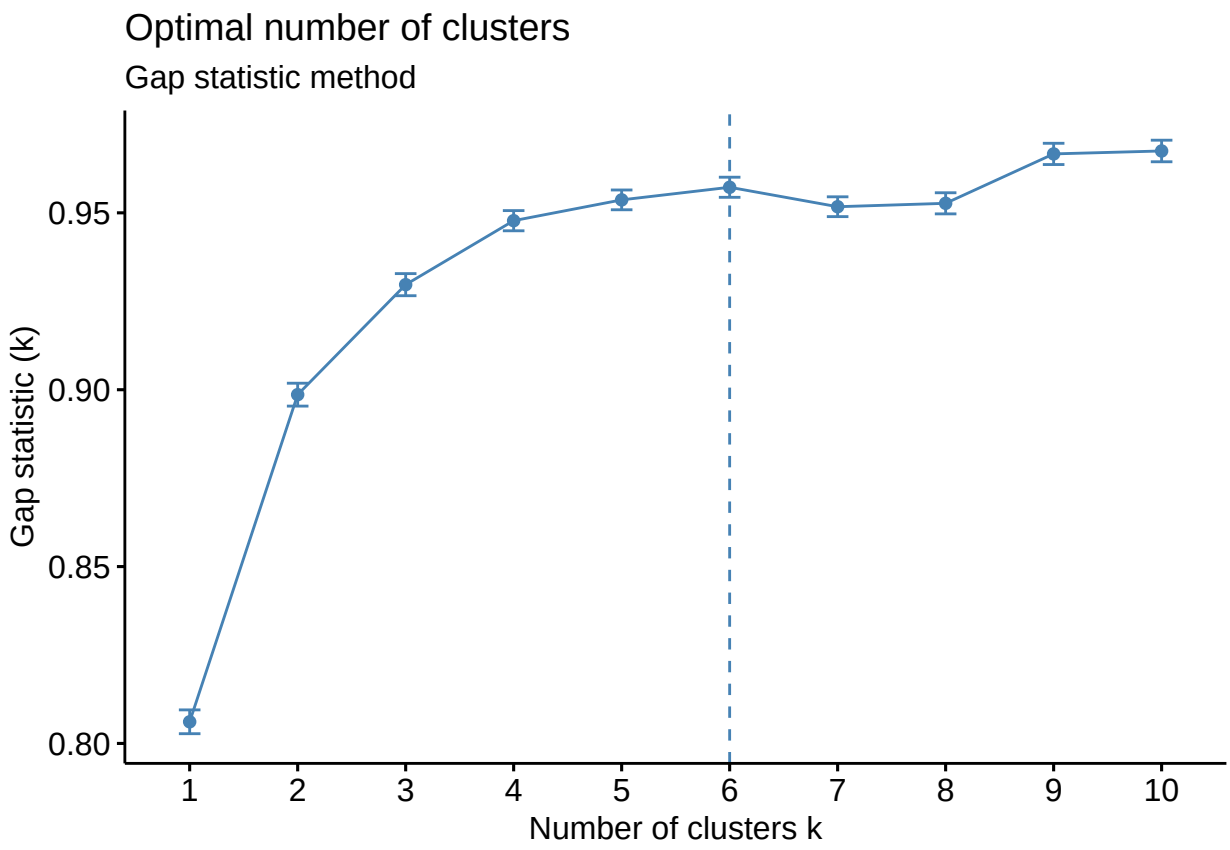



```r
silhouette_score <- function(k){
  km <- kmeans(pca_df_num, centers = k, nstart=25)
  ss <- silhouette(km$cluster, dist(pca_df_num))
  mean(ss[, 3])
}
```

```r
k <- 2:15
avg_sil <- sapply(k, silhouette_score)
plot(k, type='b', avg_sil, xlab='Number of clusters', ylab='Average Silhouette Scores', frame=FALSE)
```

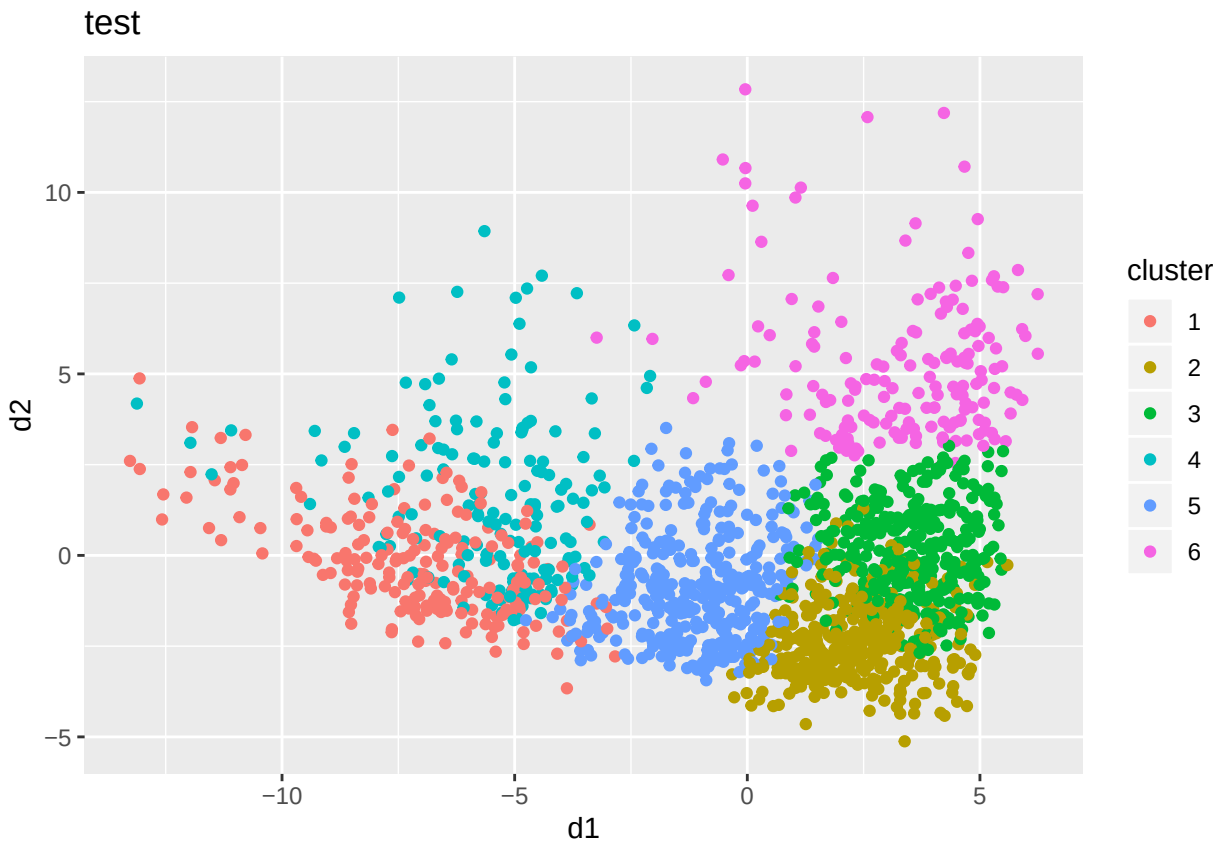



```
# Gap statistic
# nboot = 50 to keep the function speedy.
# recommended value: nboot= 500 for your analysis.
# Use verbose = FALSE to hide computing progression.
set.seed(123)
fviz_nbclust(pca_df_num, kmeans, nstart = 25,  method = "gap_stat", nboot = 50)+
  labs(subtitle = "Gap statistic method")
```




```
set.seed(123)

kcluster = clusGap(pca_df_num,FUNcluster = kmeans, nstart = 25, K.max = 15, B = 50)
k_chosen <- maxSE(f = kcluster$Tab[, "gap"], SE.f = kcluster$Tab[, "SE.sim"])
```

```
km <- kmeans(pca_df_num, centers = k_chosen, nstart=25)
subset_players_18$cluster <-km$cluster

library(ggplot2)
subset_players_18$cluster <- as.factor(subset_players_18$cluster)
subset_players_18$d1 <- pca_df_num$PC1
subset_players_18$d2 <- pca_df_num$PC2


ggplot(subset_players_18, aes(x= d1, y= d2, colour= cluster, label=player_name))+
  geom_point() + ggtitle("test")
```


test
d2
d1
10
5
0
−5
−10
−5
0
5
cluster
1
2
3
4
5
6