# MVA_Assignment_4

Aman

10/08/2020

## Assignment 4 - PCA

This document does a PCA (Principal component analysis) on the Heart Failure Prediction dataset

## Let us load libraries and data

```
# clear environment
rm(list = ls())

# defining libraries

library(ggplot2)
library(dplyr)
library(PerformanceAnalytics)
library(data.table)
library(sqldf)
library(nortest)
library(tidyverse)
library(MASS)
library(rpart)
library(class)
library(ISLR)
library(scales)
library(ClustOfVar)
library(GGally)
library(reticulate)
library(ggthemes)
library(RColorBrewer)
library(gridExtra)
library(kableExtra)
library(Hmisc)
library(corrplot)
library(energy)
library(nnet)
library(Hotelling)
library(car)
library(devtools)
library(ggbiplot)
library(factoextra)
```

```r
library(rgl)
library(FactoMineR)
```

```r
# reading data
data <- read.csv('/Users/mac/Downloads/heart_failure_clinical_records_dataset.csv')
str(data)
```

```
## 'data.frame':    299 obs. of  13 variables:
##  $ age                     : num  75 55 65 50 65 90 75 60 65 80 ...
##  $ anaemia                 : int  0 0 0 1 1 1 1 1 0 1 ...
##  $ creatinine_phosphokinase: int  582 7861 146 111 160 47 246 315 157 123 ...
##  $ diabetes                : int  0 0 0 0 1 0 0 1 0 0 ...
##  $ ejection_fraction       : int  20 38 20 20 20 40 15 60 65 35 ...
##  $ high_blood_pressure     : int  1 0 0 0 0 1 0 0 0 1 ...
##  $ platelets               : num  265000 263358 162000 210000 327000 ...
##  $ serum_creatinine        : num  1.9 1.1 1.3 1.9 2.7 2.1 1.2 1.1 1.5 9.4 ...
##  $ serum_sodium            : int  130 136 129 137 116 132 137 131 138 133 ...
##  $ sex                     : int  1 1 1 1 0 1 1 1 0 1 ...
##  $ smoking                 : int  0 0 1 0 0 1 0 1 0 1 ...
##  $ time                    : int  4 6 7 7 8 8 10 10 10 10 ...
##  $ DEATH_EVENT             : int  1 1 1 1 1 1 1 1 1 1 ...
```

We check to see if we have categorical variables
However we see all our variables are numeric
Even the categorical ones are binary and already have 1/0 as values

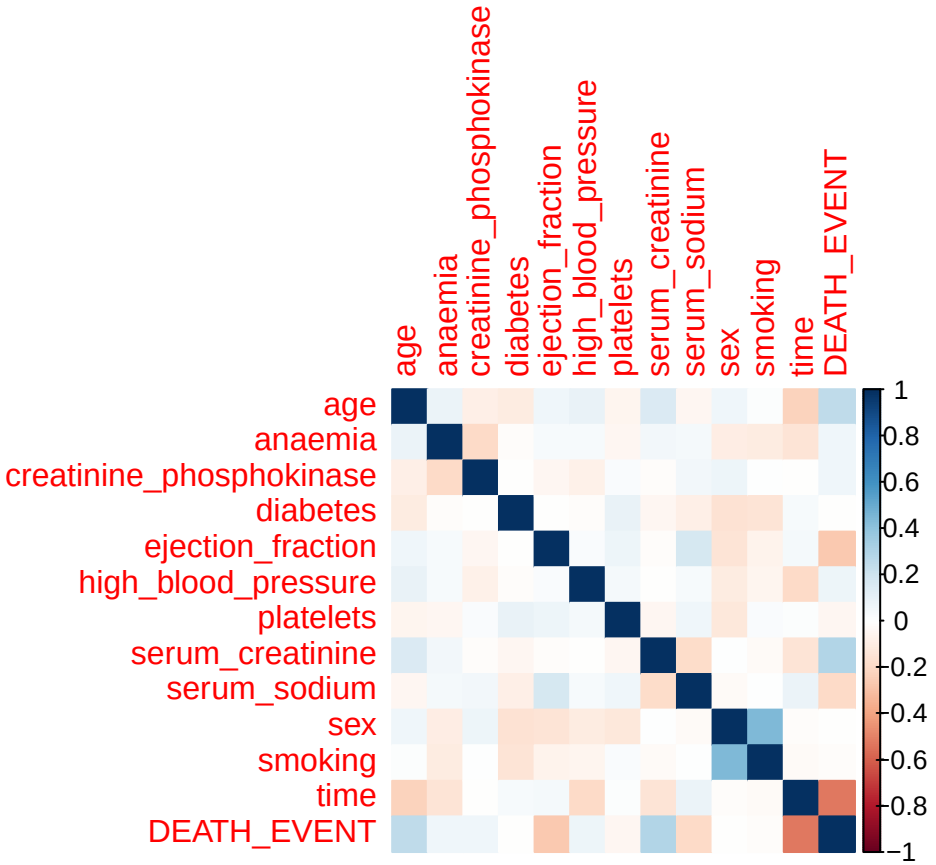
## Let's quickly revise our correlation plot

```r
# Correlation plot
M<-cor(data)
head(round(M,2))
```

```
##                            age anaemia creatinine_phosphokinase diabetes
## age                       1.00    0.09                   -0.08    -0.10
## anaemia                   0.09    1.00                   -0.19    -0.01
## creatinine_phosphokinase -0.08   -0.19                    1.00    -0.01
## diabetes                 -0.10   -0.01                   -0.01     1.00
## ejection_fraction         0.06    0.03                   -0.04     0.00
## high_blood_pressure       0.09    0.04                   -0.07    -0.01
##                          ejection_fraction high_blood_pressure platelets
## age                                   0.06               0.09     -0.05
## anaemia                               0.03               0.04     -0.04
## creatinine_phosphokinase             -0.04              -0.07      0.02
## diabetes                              0.00              -0.01      0.09
## ejection_fraction                     1.00               0.02      0.07
## high_blood_pressure                   0.02               1.00      0.05
##                          serum_creatinine serum_sodium   sex smoking  time
## age                                 0.16        -0.05  0.07    0.02 -0.22
## anaemia                             0.05         0.04 -0.09   -0.11 -0.14
## creatinine_phosphokinase           -0.02         0.06  0.08    0.00 -0.01
## diabetes                           -0.05        -0.09 -0.16   -0.15  0.03
## ejection_fraction                  -0.01         0.18 -0.15   -0.07  0.04
```

```
## high_blood_pressure                0.00         0.04 -0.10   -0.06 -0.20
##                            DEATH_EVENT
## age                               0.25
## anaemia                           0.07
## creatinine_phosphokinase          0.06
## diabetes                          0.00
## ejection_fraction                -0.27
## high_blood_pressure               0.08
```

```
corrplot(M, method="color")
```




Since most of the correlations are low (Pearson's r $< 0.25$) ), we don't particularly see a need for PCA
We use PCA to reduce the dimensionality of the dataset as PCA accomplishes this by capturing the variance in the dataset. It get the components such that the are in the direction of the highest variance.
We also saw from EDA in last exercise that our VIF was quite low indicating absence of multi-collinearity. So, reducing dimensionality may lead to loss of variance for our project. However, for exposition, we will try PCA and analyse results

## Let us perform PCA on our dataset

```
pca <- prcomp(data[,1:12],scale=TRUE)
summary(pca)
```

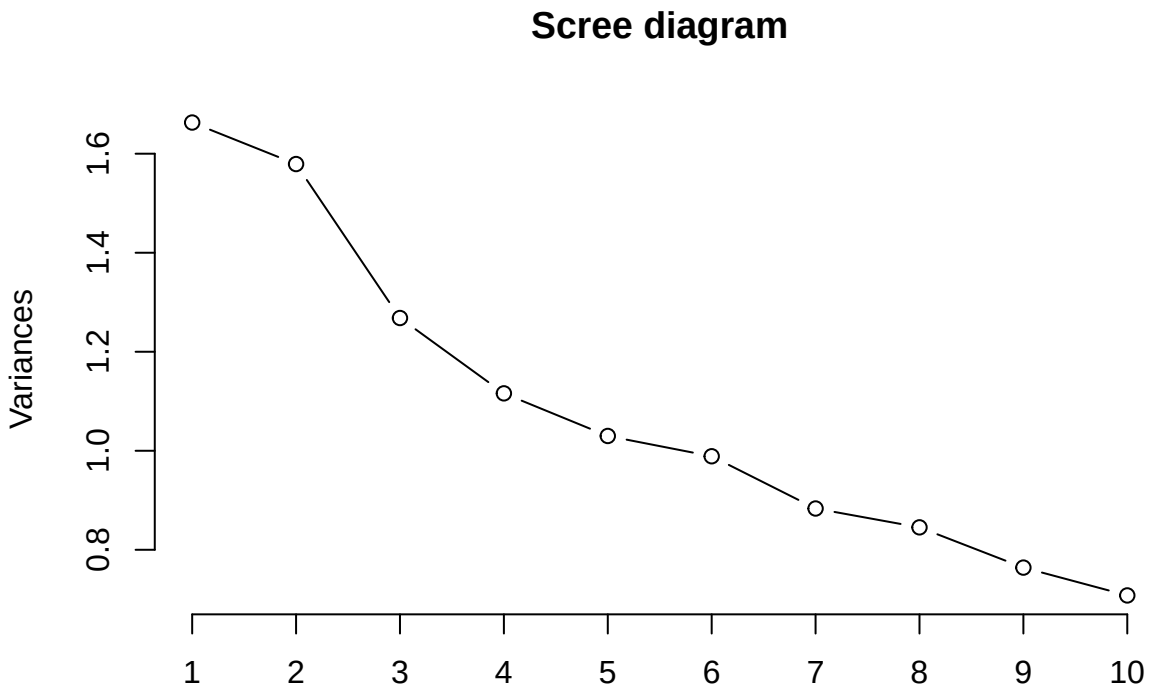
```
## Importance of components:
##                           PC1    PC2    PC3     PC4     PC5     PC6
```

```
## Standard deviation     1.2896 1.2566 1.1261 1.05638 1.01483 0.99442
## Proportion of Variance 0.1386 0.1316 0.1057 0.09299 0.08582 0.08241
## Cumulative Proportion  0.1386 0.2702 0.3759 0.46885 0.55467 0.63708
##                            PC7     PC8     PC9    PC10    PC11    PC12
## Standard deviation     0.93987 0.91940 0.87408 0.84132 0.80250 0.71457
## Proportion of Variance 0.07361 0.07044 0.06367 0.05898 0.05367 0.04255
## Cumulative Proportion  0.71069 0.78113 0.84480 0.90378 0.95745 1.00000
```

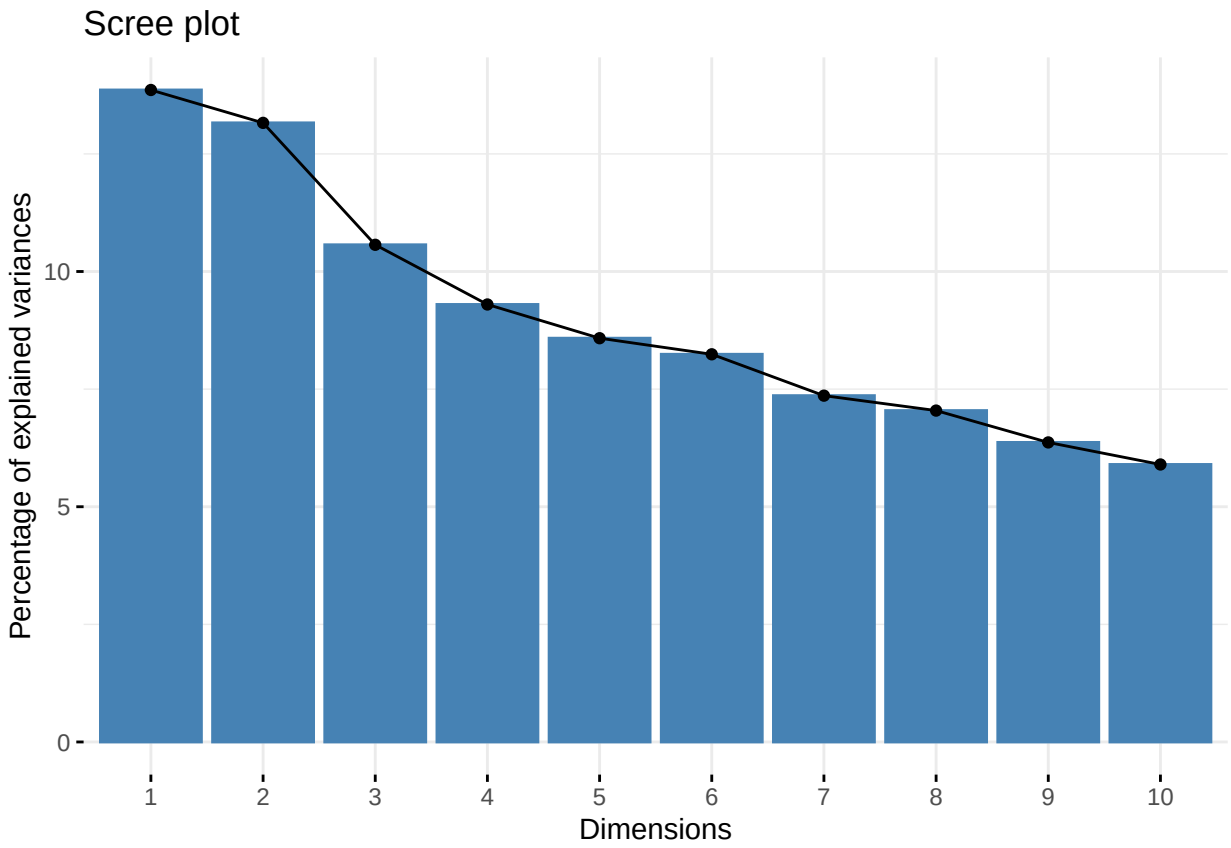
Here, we see that we need 8 components to get cumulative proportion of variance equivalent to 0.78.
For convention, we would consider as many components as required to get in the range of 0.75-0.95
Let us then consider 10 components (Cum prop. ~90%) instead of 12 reducing our dimensions from 12 to 10

## Let's plot the Scree diagrams

```
plot(pca, type="lines", main = "Scree diagram")
```

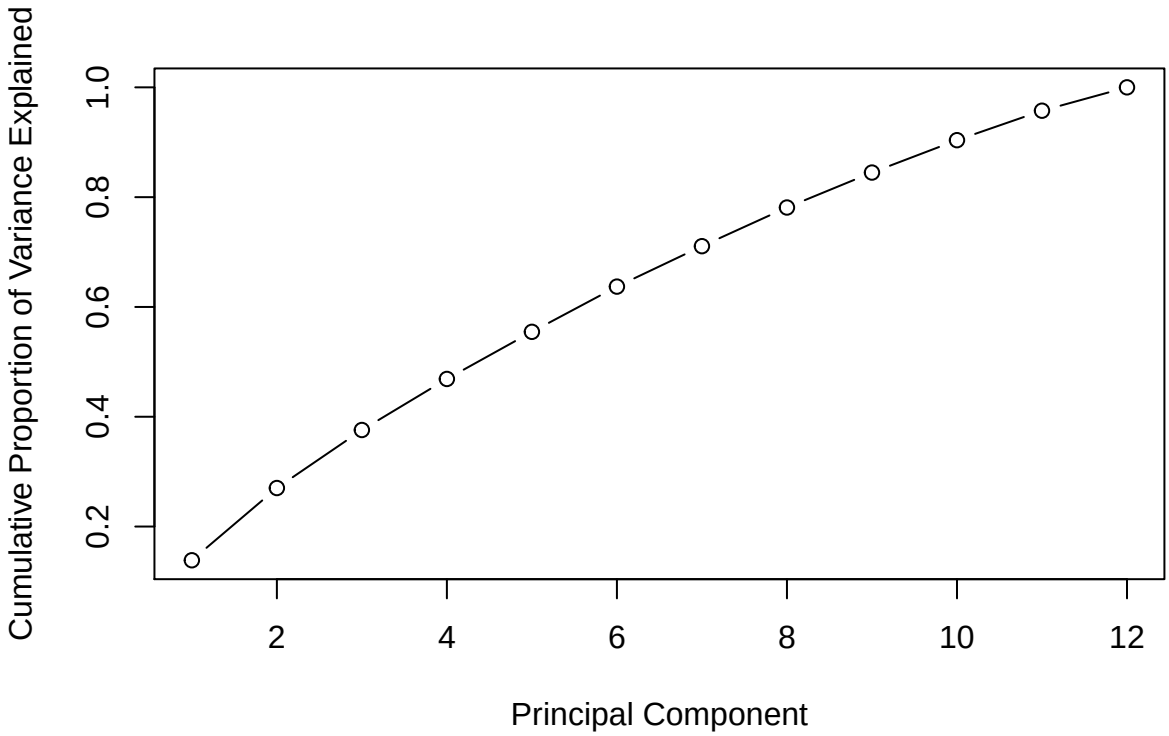



```
fviz_eig(pca)
```




## We can also see a cumulative plot

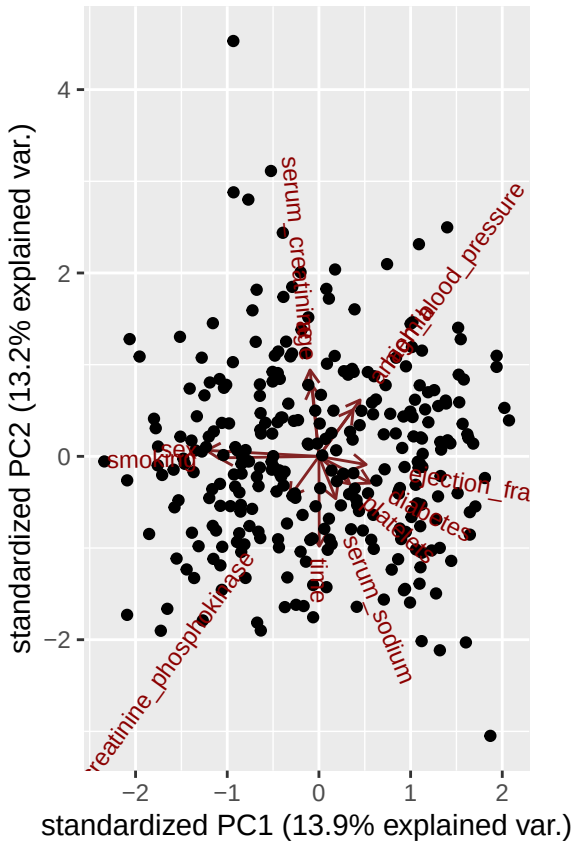
```
std_dev <- pca$sdev
pr_var <- std_dev^2
prop_varex <- pr_var/sum(pr_var)
plot(cumsum(prop_varex), xlab = "Principal Component",
     ylab = "Cumulative Proportion of Variance Explained",
     type = "b")
```




Both of the above plots (variance and cum. variance) show that we need atleast 10 components for 90% variance and since we don't see a taper down in graph of cum. variance or a steep decline in scree diagram, we can note that this isnt ideal.

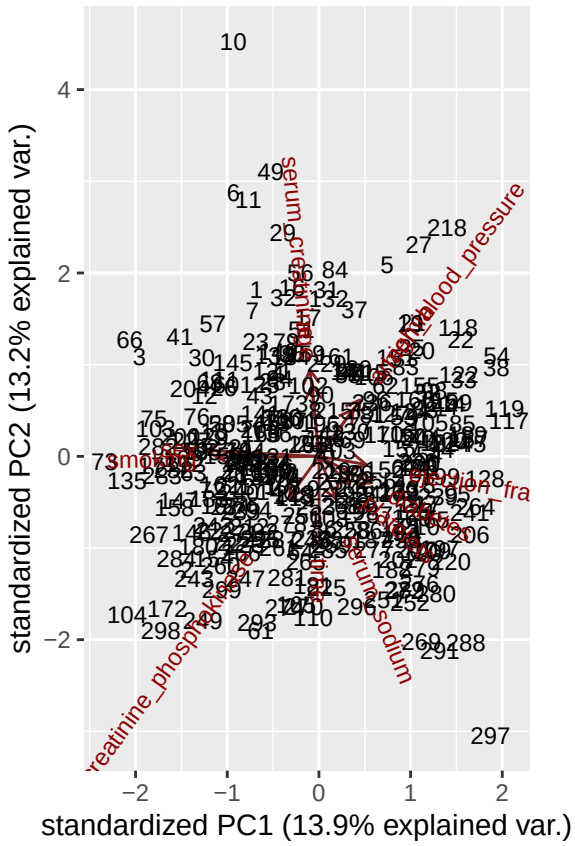
## Plotting PCA

```
# bi-plot which will use PC1 and PC2
ggbiplot(pca)
```




Here, we can tell that ejection_fraction, diabetes, platelets all contribute to PC1 with higher values in these features moving the samples to the right
Similarly we can tell that age, serum_creatinine contributes more towards PC2
In PC1, we can see sex, smoking towards negative side of PC1
In PC2, we can time towards negative side of PC2

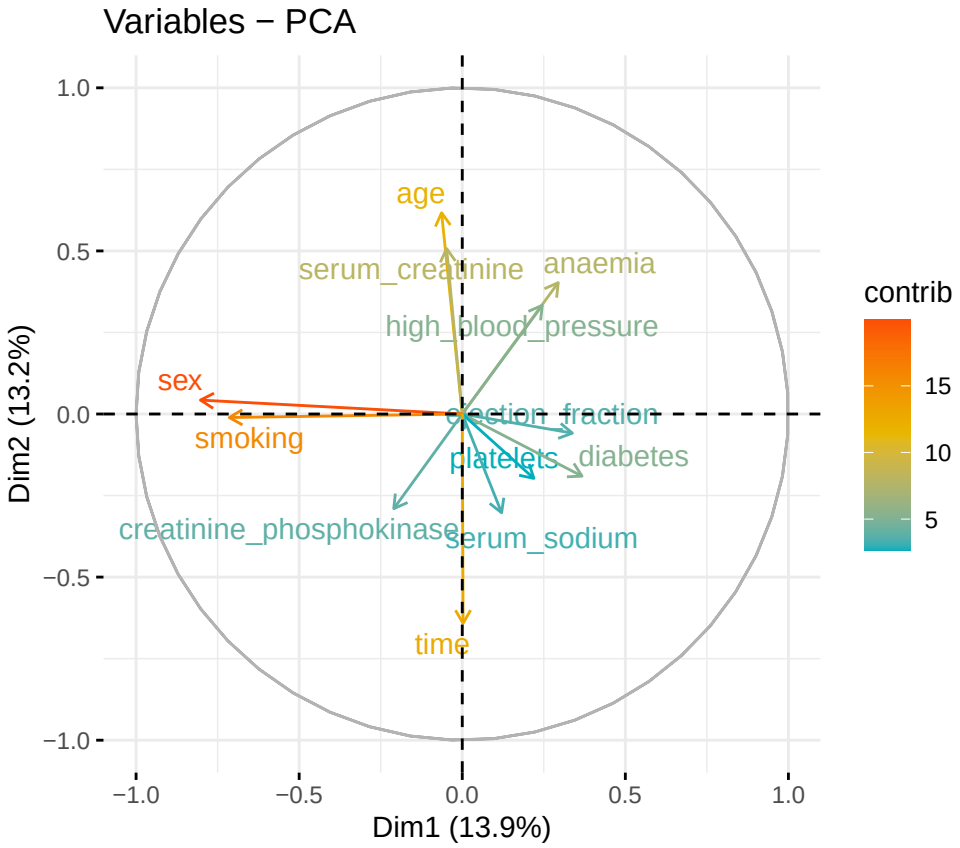
```
# We can also tell which patients are similar to one other
# by adding rownames
# Let's use each row as a patient identifier, then,

ggbiplot(pca, labels=rownames(data))
```




This tells us that patient IDs-16,32,56 are similar as they cluster together This would ideally be helpful with more meaningful identifiers

## Let's also look at contribution by variables

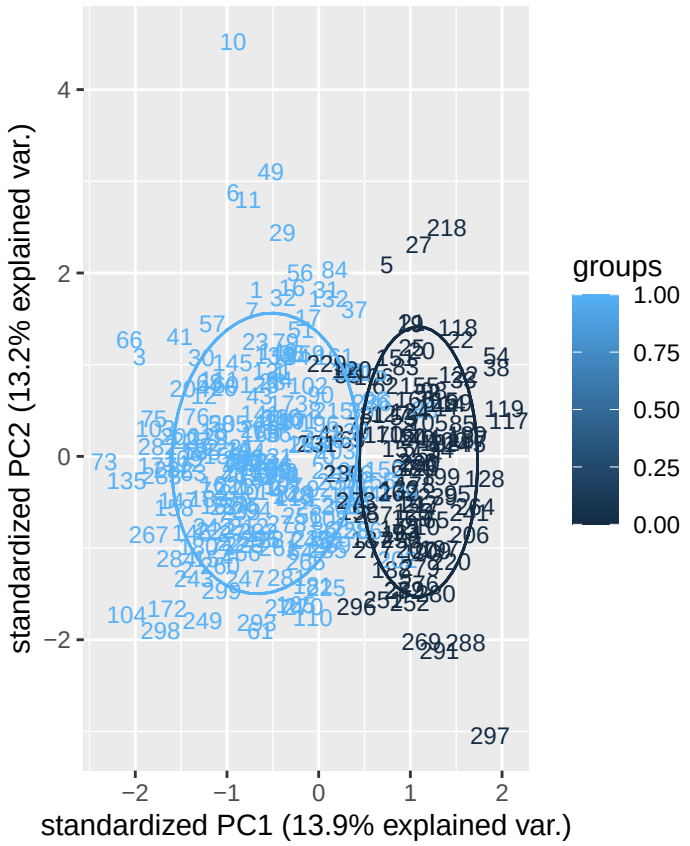
```
fviz_pca_var(pca,
             col.var = "contrib", # Color by contributions to the PC
             gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),
             repel = TRUE     # Avoid text overlapping
)
```




We see that age, sex, smoking contribute more to PC1 and PC2 so we can try and visualize this in more detail by bi-plots with these groups.

## Let's plot the bi-plot with gender

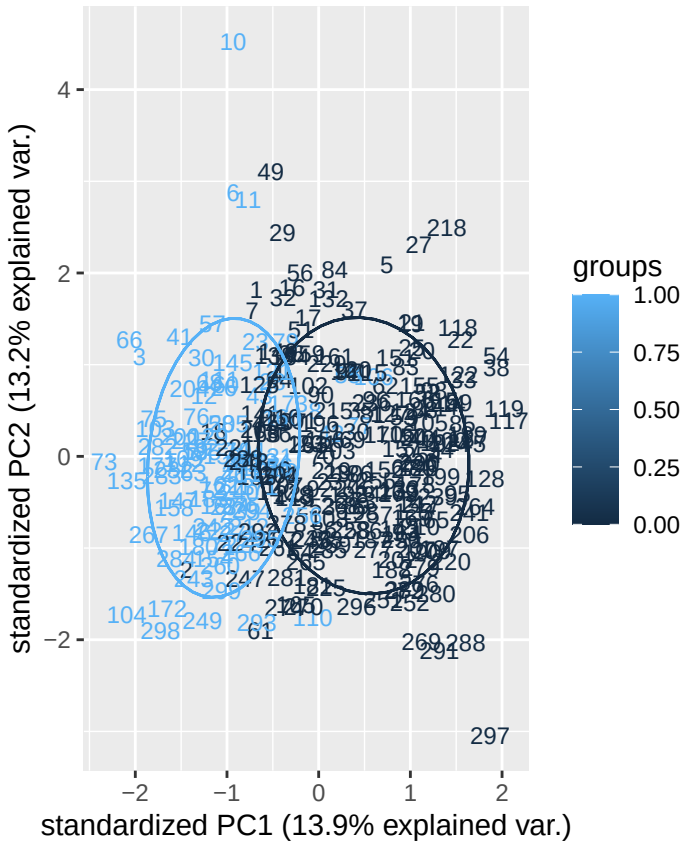
```
ggbiplot(pca,ellipse=TRUE, var.axes=FALSE, labels=rownames(data), groups=data$sex)
```




A clear indicator that males indicated by 1 have more breadth in PC1 as opposed to Females indicated by 0 which are more narrow along with that we see +ve indication for females along PC1 and negative for males

## Let's plot the bi-plot with smoking

```
ggbiplot(pca,ellipse=TRUE, var.axes=FALSE, labels=rownames(data), groups=data$smoking)
```



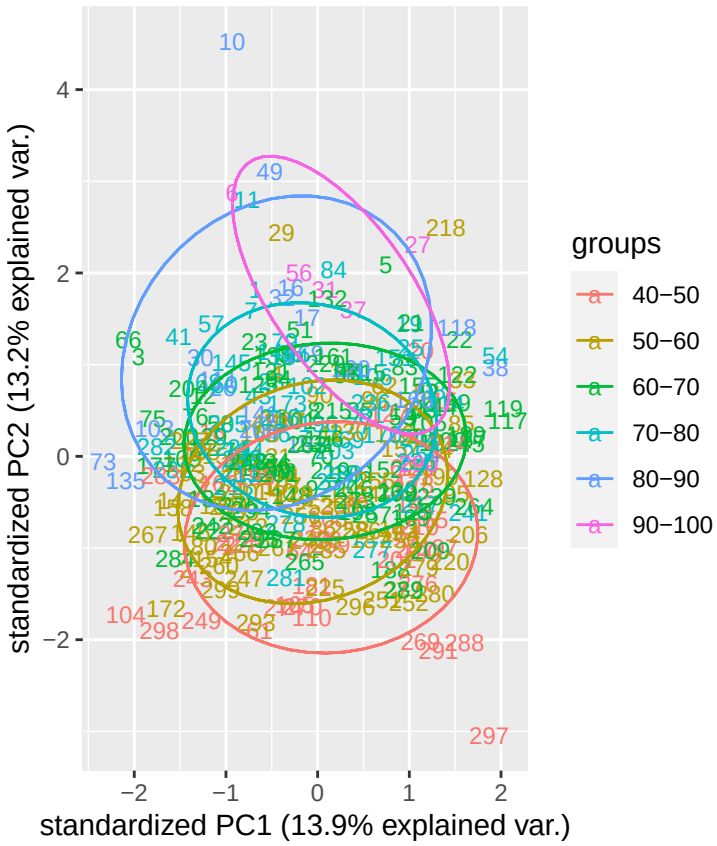

A clear indicator that smokers indicated by 1 have less breadth in PC1 as opposed to non-smokers indicated by 0 which are more wider and to the positive side along with that we see +ve indication for non-smokers for PC1 and negative for smokers

## We will create an age range variable and do the same as well

```
data$age_tr[data$age < 50 & data$age >= 40]="40-50"
data$age_tr[data$age < 60 & data$age >= 50]="50-60"
data$age_tr[data$age < 70 & data$age >= 60]="60-70"
data$age_tr[data$age < 80 & data$age >= 70]="70-80"
data$age_tr[data$age < 90 & data$age >= 80]="80-90"
data$age_tr[data$age < 100 & data$age >= 90]="90-100"
```

## And then plot the same result with

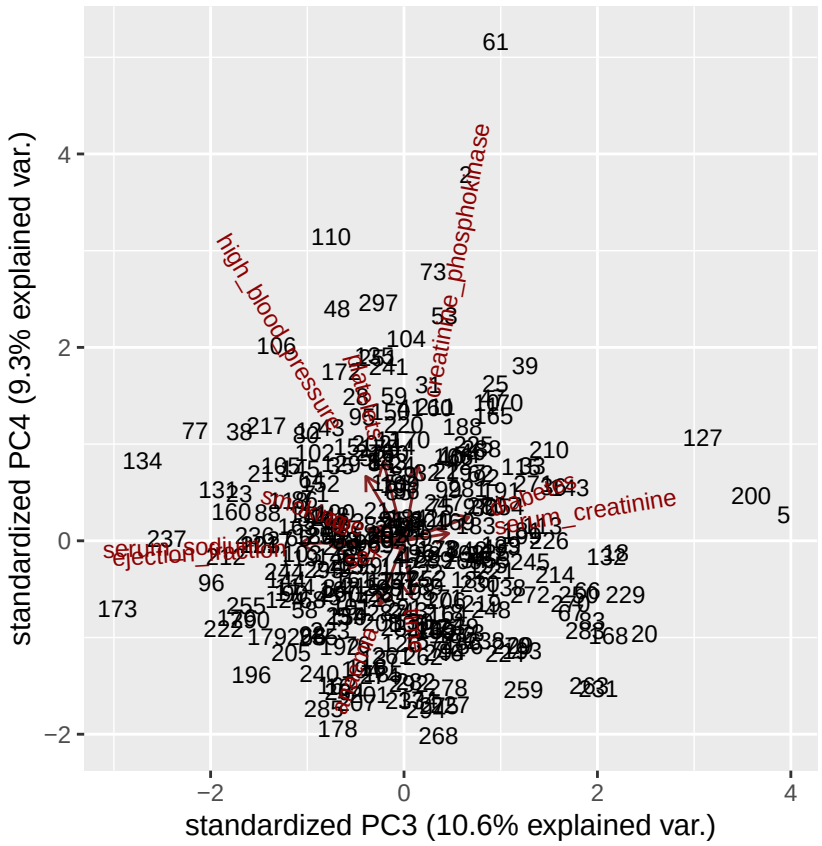
```
ggbiplot(pca,ellipse=TRUE, var.axes=FALSE, labels=rownames(data), groups=data$age_tr)
```




Not much indication here other than higher age groups tend to be more spread out in PC2

## We can also look at PC3 and PC4

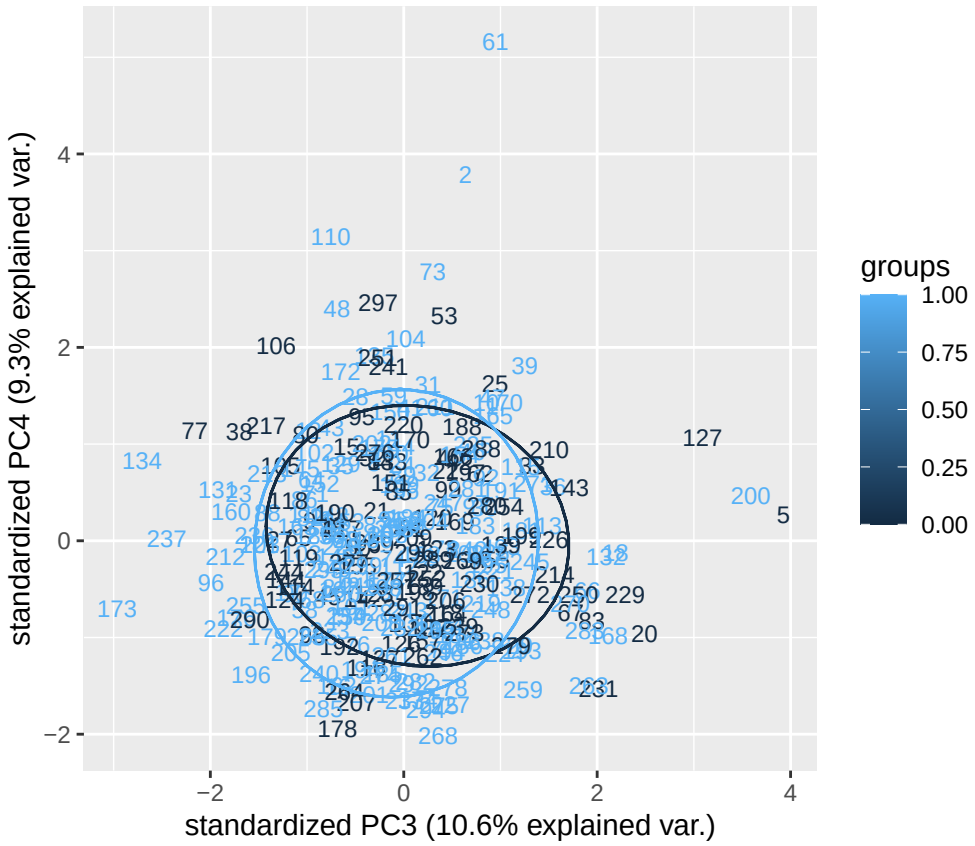
```
ggbiplot(pca,ellipse=TRUE,choices=c(3,4), labels=rownames(data))
```




serum_creatinine, diabetes more towards PC3
Platelets, creatinine_phosphokinase, and high bp more towards PC4

## By gender

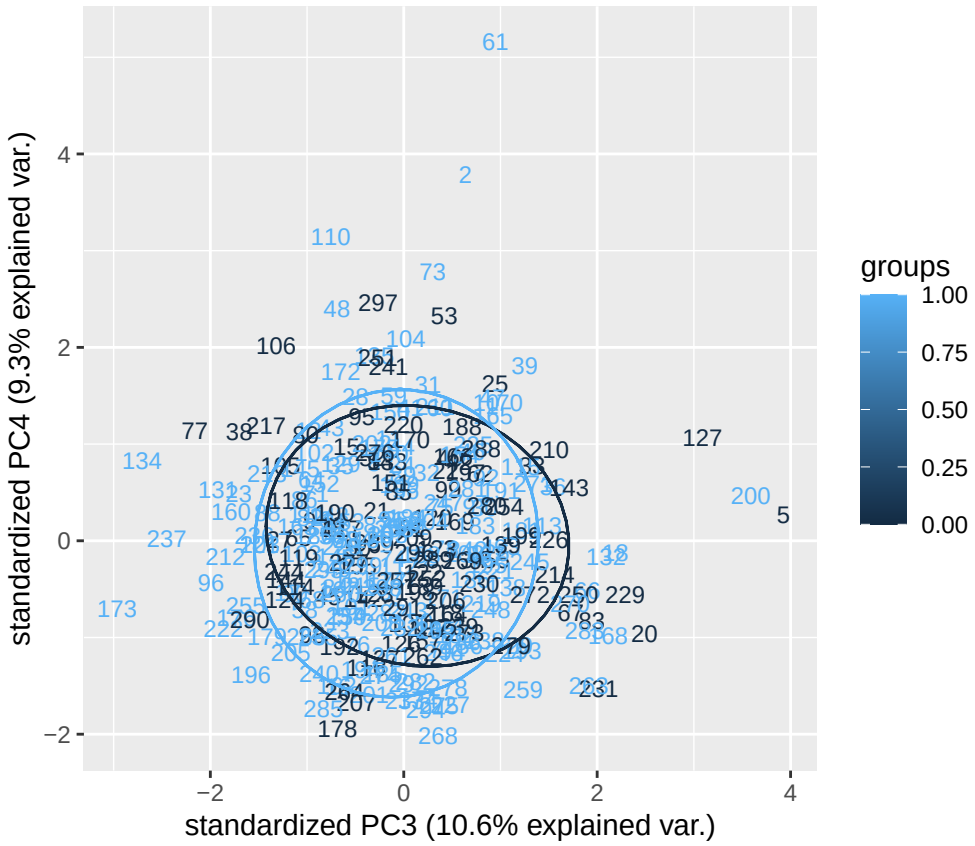
```
ggbiplot(pca,ellipse=TRUE, var.axes=FALSE,choices=c(3,4), labels=rownames(data), groups=data$sex)
```




We note even spread in PC3, PC4 for gender

## By smoking

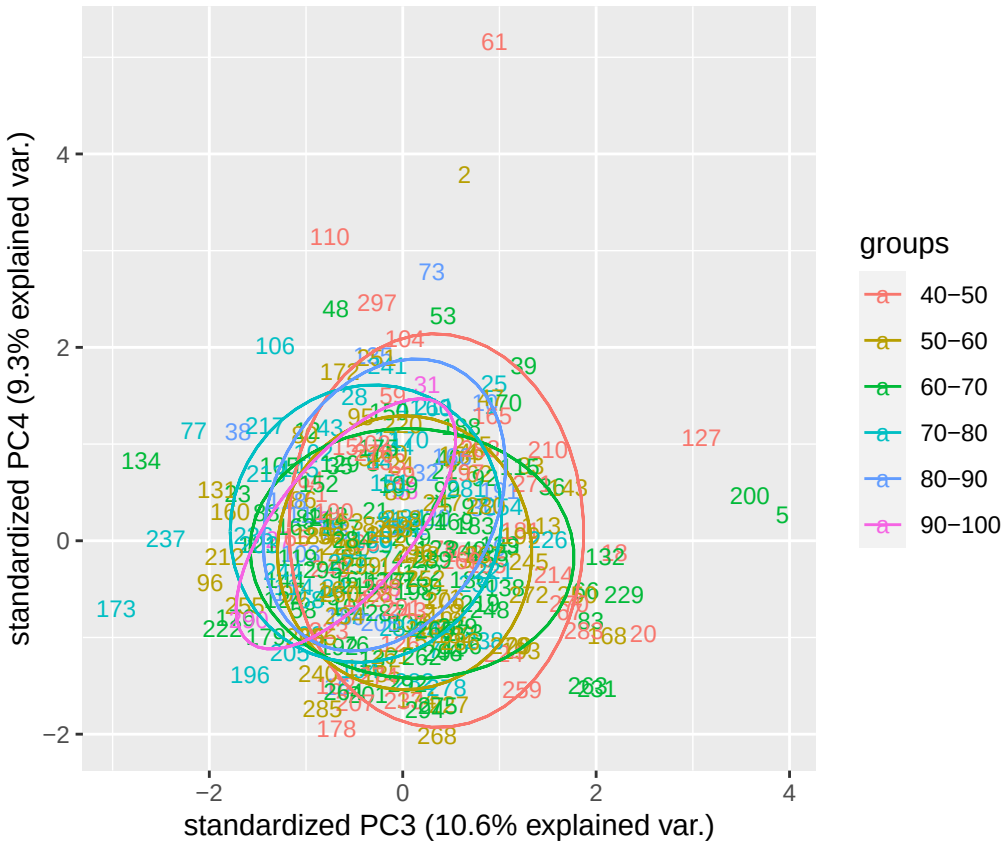
```
ggbiplot(pca,ellipse=TRUE, var.axes=FALSE,choices=c(3,4), labels=rownames(data), groups=data$sex)
```




We note even spread in PC3, PC4 for smoking

## By age groups

```
ggbiplot(pca,ellipse=TRUE, var.axes=FALSE, choices=c(3,4), labels=rownames(data), groups=data$age_tr)
```




We note age group 40-50 with most spread in PC4

## Let us do a visualizations to see how much of each variable is present in each component

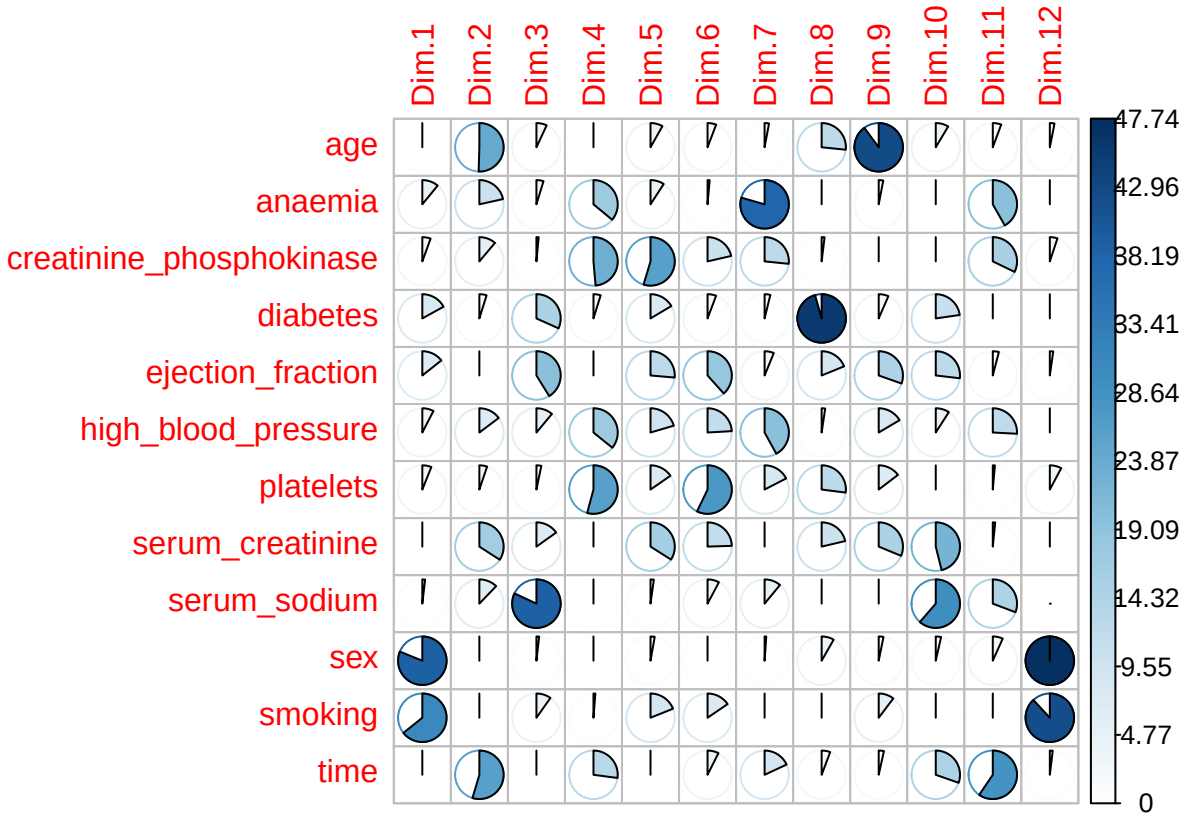
**We use factoextra and factominer for this**

```
pca_viz <- PCA(data[,1:12], graph = FALSE,ncp =12)
var <- get_pca_var(pca_viz)

# We can now use the contrib function to get contribution of each variable
# to the PCs
var$contrib
```

```
##                              Dim.1        Dim.2      Dim.3       Dim.4
## age                     0.241090384 24.101434995  3.3163296  0.39280771
## anaemia                 5.206735608 10.269724031  2.2631127 17.17630734
## creatinine_phosphokinase  2.649236139  5.332487239  0.7053467 23.20438561
## diabetes                8.105226639  2.297576996 15.1507483  2.29189172
## ejection_fraction       6.854530236  0.217736177 19.6527955  0.07657524
## high_blood_pressure     3.619721576  7.032538446  5.2796886 17.09003399
## platelets               2.897262990  2.465399361  1.5514524 25.86579597
## serum_creatinine        0.138380266 16.289311870  7.2362369  0.22159879
## serum_sodium            0.874941443  5.801880835 39.1285510  0.03348228
## sex                    38.735237509  0.115390255  0.8641004  0.12270511
## smoking                30.677244573  0.007309816  4.6786480  0.59294946
## time                    0.000392639 26.069209979  0.1729899 12.93146679
##                              Dim.5      Dim.6      Dim.7      Dim.8
## age                     3.9011856  2.7555641  1.4007020 12.6976634
## anaemia                 4.3790165  0.6542563 37.9390831  0.2779042
## creatinine_phosphokinase 26.0814913 10.2180018 12.7005328  0.9270554
## diabetes                7.8596708  2.7390564  1.8247821 45.6451006
## ejection_fraction      12.5914655 18.3098685  2.9668747  8.9975332
## high_blood_pressure     9.7651512 11.4973519 20.0374382  1.1248799
## platelets               7.3305815 27.2882869  8.3116040 12.8857440
## serum_creatinine       16.2258154 11.6967660  0.3065137 10.1791449
## serum_sodium            1.0585428  3.7109241  5.2334965  0.1588060
## sex                     1.3302428  0.2030701  0.5239019  3.9385503
## smoking                 9.0334496  7.3891019  0.1218295  0.4260566
## time                    0.4433871  3.5377519  8.6332413  2.7415614
##                               Dim.9      Dim.10       Dim.11       Dim.12
## age                     42.91511507  4.09657622 2.704958e+00 1.476573e+00
## anaemia                  1.44340862  0.03280328 1.994331e+01 4.143354e-01
## creatinine_phosphokinase  0.11229815  0.28937299 1.534936e+01 2.430432e+00
## diabetes                 3.12016545 10.84650145 1.409905e-02 1.051806e-01
## ejection_fraction       14.50809463 12.80508851 1.943706e+00 1.075731e+00
## high_blood_pressure      7.87271788  4.34111365 1.231598e+01 2.338126e-02
## platelets                6.98624451  0.02931458 6.671889e-01 3.721125e+00
## serum_creatinine        14.90486479 22.00615253 7.600028e-01 3.521202e-02
## serum_sodium             0.03335447 29.28118472 1.468480e+01 3.794302e-05
## sex                      1.51876722  1.69988788 3.212703e+00 4.773544e+01
## smoking                  4.92532200  0.08076016 1.832524e-04 4.206715e+01
## time                     1.65964721 14.49124402 2.840370e+01 9.154034e-01
```

## Let's plot this -

```
corrplot(var$contrib, is.corr=FALSE, method="pie")
```




**Key Observations**

1. Sex and Smoking are dominant in PC1
2. Age and time are dominant in PC2
3. Serum_Sodium is dominant in PC3
4. Platelets and creatinine_phosphokinase are dominant in PC4
5. creatinine_phosphokinase is dominant in PC5
6. Platelets and ejection_fraction are dominant in PC6
7. Anaemia is dominant in PC7
8. Diabetes is dominant in PC8
9. Age is dominant in PC9
10. Serum_Sodium, Serum_creatinine is dominant in PC10

11. Time, anaemia is dominant in PC11
12. Sex and smoking are dominant in PC12

**Note**

We don't see a good combination of variables in any component and PC12 is redundant as PC1 and gives same information

## Let us now combine the pca with dataset

```
data_pca <- cbind(data,pca$x)
```

The new dataset now has 26 variables with PC1-PC12 added

## Now Let us check the means by death events

```
meansPC <- aggregate(data_pca[,15:26],by=list(DEATH_EVENT=data$DEATH_EVENT),mean)
meansPC
```

```
##   DEATH_EVENT         PC1        PC2       PC3        PC4         PC5
## 1           0  0.06334102 -0.4172118 -0.163244 -0.1645663  0.02037653
## 2           1 -0.13393986  0.8822291  0.345193  0.3479892 -0.04308787
##            PC6        PC7          PC8         PC9         PC10       PC11
## 1 -0.07937634  0.1065350  0.002886629  0.03360757 -0.006010037 -0.1031019
## 2  0.16784789 -0.2252772 -0.006104018 -0.07106601  0.012708725  0.2180175
##          PC12
## 1  0.02108797
## 2 -0.04459228
```

## Let us check stddev by death events

```
sdsPC <- aggregate(data_pca[,15:26],by=list(DEATH_EVENT=data$DEATH_EVENT),sd)
sdsPC
```

```
##   DEATH_EVENT      PC1      PC2      PC3       PC4       PC5       PC6
## 1           0 1.290014 1.003036 1.066432 0.9419239 0.8557085 0.9279099
## 2           1 1.285022 1.286723 1.175754 1.1974011 1.2926138 1.1087023
##         PC7       PC8       PC9      PC10      PC11      PC12
## 1 0.8975874 0.8653738 0.7849849 0.7825581 0.7830979 0.7022593
## 2 0.9911225 1.0291470 1.0386759 0.9580715 0.8034110 0.7416994
```

We notice a clear difference in means (note the different signs) however not much in std. deviation
This may indicate that PCs aren't doing a good job in segregating the death events from non-death events

## Let us perform t-tests

```
t.test(PC1~data_pca$DEATH_EVENT,data=data_pca)
```

```
##
##  Welch Two Sample t-test
##
## data:  PC1 by data_pca$DEATH_EVENT
## t = 1.2379, df = 187.14, p-value = 0.2173
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.1171103  0.5116720
## sample estimates:
## mean in group 0 mean in group 1
##      0.06334102     -0.13393986
```

```
t.test(PC2~data_pca$DEATH_EVENT,data=data_pca)
```

```
##
##  Welch Two Sample t-test
##
## data:  PC2 by data_pca$DEATH_EVENT
## t = -8.7208, df = 151.56, p-value = 4.57e-15
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.593836 -1.005046
## sample estimates:
## mean in group 0 mean in group 1
##      -0.4172118       0.8822291
```

```
t.test(PC3~data_pca$DEATH_EVENT,data=data_pca)
```

```
##
##  Welch Two Sample t-test
##
## data:  PC3 by data_pca$DEATH_EVENT
## t = -3.595, df = 171.12, p-value = 0.0004241
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.7876082 -0.2292657
## sample estimates:
## mean in group 0 mean in group 1
##       -0.163244        0.345193
```

```
t.test(PC4~data_pca$DEATH_EVENT,data=data_pca)
```

```
##
##  Welch Two Sample t-test
##
## data:  PC4 by data_pca$DEATH_EVENT
## t = -3.6889, df = 152.59, p-value = 0.0003127
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.7870593 -0.2380518
## sample estimates:
```

```
## mean in group 0 mean in group 1
##      -0.1645663       0.3479892
```

```
t.test(PC5~data_pca$DEATH_EVENT,data=data_pca)
```

```
##
##  Welch Two Sample t-test
##
## data:  PC5 by data_pca$DEATH_EVENT
## t = 0.43782, df = 135.72, p-value = 0.6622
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.2231971  0.3501259
## sample estimates:
## mean in group 0 mean in group 1
##      0.02037653     -0.04308787
```

```
t.test(PC6~data_pca$DEATH_EVENT,data=data_pca)
```

```
##
##  Welch Two Sample t-test
##
## data:  PC6 by data_pca$DEATH_EVENT
## t = -1.8936, df = 160.1, p-value = 0.06009
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.50506564  0.01061717
## sample estimates:
## mean in group 0 mean in group 1
##     -0.07937634      0.16784789
```

```
t.test(PC7~data_pca$DEATH_EVENT,data=data_pca)
```

```
##
##  Welch Two Sample t-test
##
## data:  PC7 by data_pca$DEATH_EVENT
## t = 2.7844, df = 170.89, p-value = 0.005968
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.09657887 0.56704564
## sample estimates:
## mean in group 0 mean in group 1
##       0.1065350      -0.2252772
```

```
t.test(PC8~data_pca$DEATH_EVENT,data=data_pca)
```

```
##
##  Welch Two Sample t-test
##
## data:  PC8 by data_pca$DEATH_EVENT
## t = 0.074099, df = 160.7, p-value = 0.941
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.2306227  0.2486040
## sample estimates:
## mean in group 0 mean in group 1
```

```
##     0.002886629    -0.006104018
```

```
t.test(PC9~data_pca$DEATH_EVENT,data=data_pca)
```

```
##
##  Welch Two Sample t-test
##
## data:  PC9 by data_pca$DEATH_EVENT
## t = 0.87614, df = 148.17, p-value = 0.3824
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.1314148  0.3407619
## sample estimates:
## mean in group 0 mean in group 1
##      0.03360757     -0.07106601
```

```
t.test(PC10~data_pca$DEATH_EVENT,data=data_pca)
```

```
##
##  Welch Two Sample t-test
##
## data:  PC10 by data_pca$DEATH_EVENT
## t = -0.1669, df = 157.05, p-value = 0.8677
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.2402408  0.2028033
## sample estimates:
## mean in group 0 mean in group 1
##    -0.006010037     0.012708725
```

```
t.test(PC11~data_pca$DEATH_EVENT,data=data_pca)
```

```
##
##  Welch Two Sample t-test
##
## data:  PC11 by data_pca$DEATH_EVENT
## t = -3.253, df = 182.24, p-value = 0.001361
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.5158895 -0.1263492
## sample estimates:
## mean in group 0 mean in group 1
##      -0.1031019       0.2180175
```

```
t.test(PC12~data_pca$DEATH_EVENT,data=data_pca)
```

```
##
##  Welch Two Sample t-test
##
## data:  PC12 by data_pca$DEATH_EVENT
## t = 0.7271, df = 177.61, p-value = 0.4681
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.1125810  0.2439415
## sample estimates:
## mean in group 0 mean in group 1
##      0.02108797     -0.04459228
```

We notice signifcant results in PC2, PC3, PC4, and PC11 at alpha=0.5

## Let us also perform F-ratio tests

```
var.test(PC1~data_pca$DEATH_EVENT,data=data_pca)
```

```
##
##  F test to compare two variances
##
## data:  PC1 by data_pca$DEATH_EVENT
## F = 1.0078, num df = 202, denom df = 95, p-value = 0.9818
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.7049109 1.4096115
## sample estimates:
## ratio of variances
##           1.007784
```

```
var.test(PC2~data_pca$DEATH_EVENT,data=data_pca)
```

```
##
##  F test to compare two variances
##
## data:  PC2 by data_pca$DEATH_EVENT
## F = 0.60766, num df = 202, denom df = 95, p-value = 0.003495
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.4250391 0.8499515
## sample estimates:
## ratio of variances
##          0.6076623
```

```
var.test(PC3~data_pca$DEATH_EVENT,data=data_pca)
```

```
##
##  F test to compare two variances
##
## data:  PC3 by data_pca$DEATH_EVENT
## F = 0.82268, num df = 202, denom df = 95, p-value = 0.2539
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.5754393 1.1507071
## sample estimates:
## ratio of variances
##          0.8226839
```

```
var.test(PC4~data_pca$DEATH_EVENT,data=data_pca)
```

```
##
##  F test to compare two variances
##
## data:  PC4 by data_pca$DEATH_EVENT
## F = 0.6188, num df = 202, denom df = 95, p-value = 0.004915
```

```
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.4328316 0.8655341
## sample estimates:
## ratio of variances
##          0.6188029
```

```
var.test(PC5~data_pca$DEATH_EVENT,data=data_pca)
```

```
##
##  F test to compare two variances
##
## data:  PC5 by data_pca$DEATH_EVENT
## F = 0.43824, num df = 202, denom df = 95, p-value = 1.071e-06
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.3065355 0.6129796
## sample estimates:
## ratio of variances
##          0.4382422
```

```
var.test(PC6~data_pca$DEATH_EVENT,data=data_pca)
```

```
##
##  F test to compare two variances
##
## data:  PC6 by data_pca$DEATH_EVENT
## F = 0.70046, num df = 202, denom df = 95, p-value = 0.03751
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.4899461 0.9797461
## sample estimates:
## ratio of variances
##          0.7004574
```

```
var.test(PC7~data_pca$DEATH_EVENT,data=data_pca)
```

```
##
##  F test to compare two variances
##
## data:  PC7 by data_pca$DEATH_EVENT
## F = 0.82016, num df = 202, denom df = 95, p-value = 0.2466
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.5736743 1.1471775
## sample estimates:
## ratio of variances
##          0.8201604
```

```
var.test(PC8~data_pca$DEATH_EVENT,data=data_pca)
```

```
##
##  F test to compare two variances
##
## data:  PC8 by data_pca$DEATH_EVENT
## F = 0.70705, num df = 202, denom df = 95, p-value = 0.04286
## alternative hypothesis: true ratio of variances is not equal to 1
```

```
## 95 percent confidence interval:
##  0.4945603 0.9889732
## sample estimates:
## ratio of variances
##          0.7070542
```

```
var.test(PC9~data_pca$DEATH_EVENT,data=data_pca)
```

```
##
##  F test to compare two variances
##
## data:  PC9 by data_pca$DEATH_EVENT
## F = 0.57117, num df = 202, denom df = 95, p-value = 0.001005
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.3995113 0.7989035
## sample estimates:
## ratio of variances
##          0.5711662
```

```
var.test(PC10~data_pca$DEATH_EVENT,data=data_pca)
```

```
##
##  F test to compare two variances
##
## data:  PC10 by data_pca$DEATH_EVENT
## F = 0.66717, num df = 202, denom df = 95, p-value = 0.01793
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.4666635 0.9331880
## sample estimates:
## ratio of variances
##          0.6671712
```

```
var.test(PC11~data_pca$DEATH_EVENT,data=data_pca)
```

```
##
##  F test to compare two variances
##
## data:  PC11 by data_pca$DEATH_EVENT
## F = 0.95007, num df = 202, denom df = 95, p-value = 0.7546
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.6645431 1.3288880
## sample estimates:
## ratio of variances
##          0.9500721
```

```
var.test(PC12~data_pca$DEATH_EVENT,data=data_pca)
```

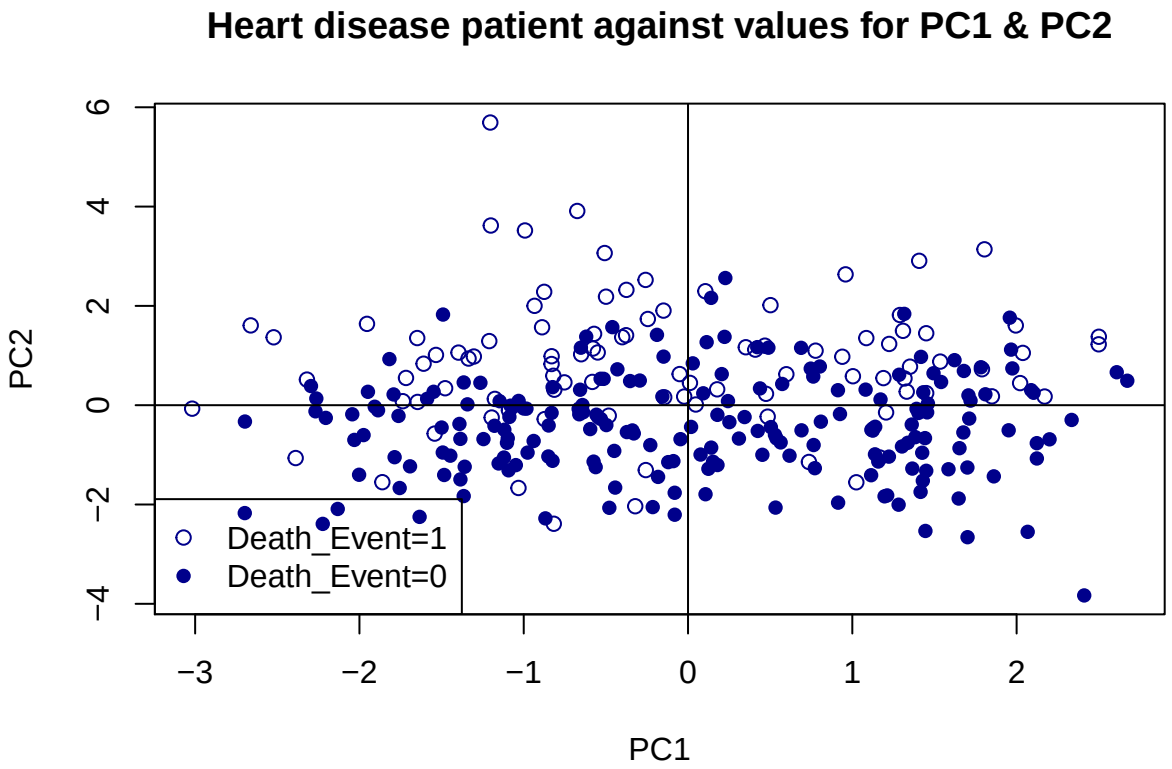
```
##
##  F test to compare two variances
##
## data:  PC12 by data_pca$DEATH_EVENT
## F = 0.89648, num df = 202, denom df = 95, p-value = 0.5188
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
```

```
##  0.6270551 1.2539233
## sample estimates:
## ratio of variances
##           0.896477
```

We notice signifcant results in PC2, PC4, PC5, PC6, PC8, PC9, and PC10

## Plotting the scores for the first and second components

```
plot(data_pca$PC1, data_pca$PC2,
pch=ifelse(data_pca$DEATH_EVENT == "1",1,16),xlab="PC1", ylab="PC2",col="dark blue",
main="Heart disease patient against values for PC1 & PC2")
abline(h=0)
abline(v=0)
legend("bottomleft", legend=c("Death_Event=1","Death_Event=0") ,col="dark blue", pch=c(1,16))
```




We do note that survivors seem to be closer to average than those who died
Also recall the definition of PC1 and PC2 -
PC1 was sex, smoking dominant
PC2 was age, time dominant
This also tells us that non-survivors were on the extremes of ages and follow-up period

## PCA - prediction

## We can try a prediction with pca by splitting our data into train and test and finding the PCs on train and validating on test data

```
data <- read.csv('/Users/mac/Downloads/heart_failure_clinical_records_dataset.csv')
# Split data into 2 parts for pca training (75%) and prediction (25%)
set.seed(1)
samp <- sample(nrow(data), nrow(data)*0.75)
data.train <- data[samp,]
data.valid <- data[-samp,]
dim(data.train)
```

```
## [1] 224  13
```

```
dim(data.valid)
```

```
## [1] 75 13
```

We split our data into 224 rows, and 75 rows into sets of train and valid.

## conduct PCA on training dataset

```
pca <- prcomp(data.train[,1:12], retx=TRUE, center=TRUE, scale=TRUE)
expl.var <- round(pca$sdev^2/sum(pca$sdev^2)*100) # percent explained variance
expl.var
```

```
##  [1] 14 13 11 10  9  8  7  7  6  6  5  4
```

The explained variance in components is same as before

## prediction of PCs for validation dataset

```
pred <- predict(pca, newdata=data.valid[,1:12])
head(pred,5)
```

```
##          PC1        PC2        PC3        PC4         PC5        PC6
## 1 -0.5954087 -2.4147312 -1.2869417 -1.4633921 -0.44669243  0.2404997
## 2 -1.4825899  1.4230934 -1.5406500 -2.2850123  4.91526790  4.3237734
## 5  1.1123863 -1.8537831 -4.1490234 -0.2757868 -1.61057161 -1.1684945
## 7 -0.5305120 -2.2444212 -0.5039181  1.2599211 -0.41993785  1.6722533
## 8 -0.5164002 -0.3656315  0.3163589 -0.8577554  0.02733162 -1.5838050
##         PC7        PC8         PC9       PC10         PC11       PC12
## 1 0.2009025  0.4756267  1.42931330 0.1608863 -0.425019536  1.0143996
## 2 0.1018572  0.3428343  1.00825598 3.2063081  1.214114217  0.7872561
## 5 1.7270738 -0.1946348  0.02891085 2.5410132 -0.114196476 -0.4949330
## 7 0.3511343 -0.9511788  0.97400380 0.2458546 -1.242363953  0.2571025
## 8 1.1867301 -1.2780996 -2.52022626 2.0190596 -0.009443437  0.2478751
```

We print the first 5 rows to see the predicted values in our validation set.

## Let us take first 10 components that explain 90% variance in data and do the same

```
train.data <- data.frame(DEATH_EVENT=data.train$DEATH_EVENT, pca$x)
train.data <- train.data[,1:11]

test.data <- predict(pca, newdata = data.valid)
test.data <- as.data.frame(test.data)
test.data <- test.data[,1:10]
head(test.data,5)
```

```
##          PC1        PC2        PC3        PC4         PC5        PC6
## 1 -0.5954087 -2.4147312 -1.2869417 -1.4633921 -0.44669243  0.2404997
## 2 -1.4825899  1.4230934 -1.5406500 -2.2850123  4.91526790  4.3237734
## 5  1.1123863 -1.8537831 -4.1490234 -0.2757868 -1.61057161 -1.1684945
## 7 -0.5305120 -2.2444212 -0.5039181  1.2599211 -0.41993785  1.6722533
## 8 -0.5164002 -0.3656315  0.3163589 -0.8577554  0.02733162 -1.5838050
##         PC7        PC8         PC9      PC10
## 1 0.2009025  0.4756267  1.42931330 0.1608863
## 2 0.1018572  0.3428343  1.00825598 3.2063081
## 5 1.7270738 -0.1946348  0.02891085 2.5410132
## 7 0.3511343 -0.9511788  0.97400380 0.2458546
## 8 1.1867301 -1.2780996 -2.52022626 2.0190596
```

This finally gives us the test data with PC1-10
Our final conclusion however remains the same that PCA isn't ideal for modeling purpose in our project

**This concludes our analysis of PCA in our dataset**