miniProject

```
# Save your input data file into your Project directory
fna.data <- "WisconsinCancer.csv"

# Complete the following code to input the data and store as wisc.df
wisc.df <-read.csv(fna.data, row.names=1)
wisc.data <- wisc.df[,-1]
diagnosis<-as.factor(wisc.df$diagnosis)
```

```
dim(wisc.data)
```

```
[1] 569  30
```

Q1 There are 569 observations in the data set

```
length(grep("M",diagnosis,value="TRUE"))
```

```
[1] 212
```

Q2 There are 212 observations of malignancy

```
length(grep("_mean",names(wisc.data),value="TRUE"))
```

```
[1] 10
```

Q3) There are 10 variables with _mean as their suffix

2. Principal Component Analysis

```
colMeans(wisc.data)
```

```
          radius_mean          texture_mean        perimeter_mean
         1.412729e+01          1.928965e+01          9.196903e+01
            area_mean       smoothness_mean      compactness_mean
         6.548891e+02          9.636028e-02          1.043410e-01
       concavity_mean    concave.points_mean         symmetry_mean
         8.879932e-02          4.891915e-02          1.811619e-01
fractal_dimension_mean             radius_se            texture_se
         6.279761e-02          4.051721e-01          1.216853e+00
       perimeter_se               area_se         smoothness_se
         2.866059e+00          4.033708e+01          7.040979e-03
      compactness_se         concavity_se     concave.points_se
         2.547814e-02          3.189372e-02          1.179614e-02
        symmetry_se    fractal_dimension_se          radius_worst
         2.054230e-02          3.794904e-03          1.626919e+01
       texture_worst       perimeter_worst            area_worst
         2.567722e+01          1.072612e+02          8.805831e+02
    smoothness_worst     compactness_worst       concavity_worst
         1.323686e-01          2.542650e-01          2.721885e-01
 concave.points_worst        symmetry_worst fractal_dimension_worst
         1.146062e-01          2.900756e-01          8.394582e-02
```

```
apply(wisc.data,2,sd)
```

```
          radius_mean          texture_mean        perimeter_mean
         3.524049e+00          4.301036e+00          2.429898e+01
            area_mean       smoothness_mean      compactness_mean
         3.519141e+02          1.406413e-02          5.281276e-02
       concavity_mean    concave.points_mean         symmetry_mean
         7.971981e-02          3.880284e-02          2.741428e-02
fractal_dimension_mean             radius_se            texture_se
         7.060363e-03          2.773127e-01          5.516484e-01
       perimeter_se               area_se         smoothness_se
         2.021855e+00          4.549101e+01          3.002518e-03
      compactness_se         concavity_se     concave.points_se
         1.790818e-02          3.018606e-02          6.170285e-03
        symmetry_se    fractal_dimension_se          radius_worst
         8.266372e-03          2.646071e-03          4.833242e+00
       texture_worst       perimeter_worst            area_worst
```

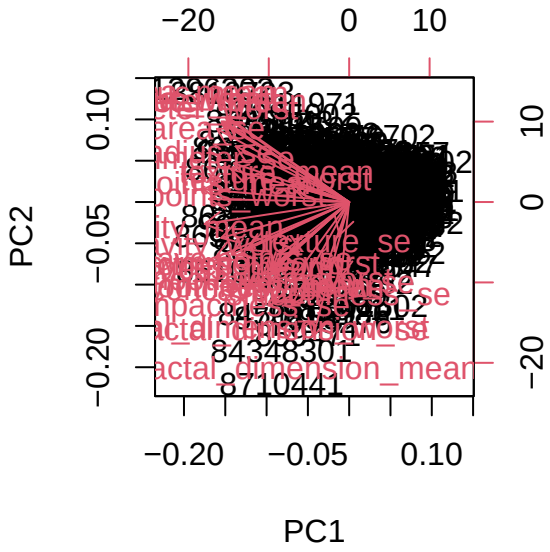
```
          6.146258e+00            3.360254e+01            5.693570e+02
     smoothness_worst       compactness_worst         concavity_worst
          2.283243e-02            1.573365e-01            2.086243e-01
 concave.points_worst          symmetry_worst fractal_dimension_worst
          6.573234e-02            6.186747e-02            1.806127e-02
```

```
wisc.pr <- prcomp(wisc.data, scale=TRUE)
summary(wisc.pr)
```

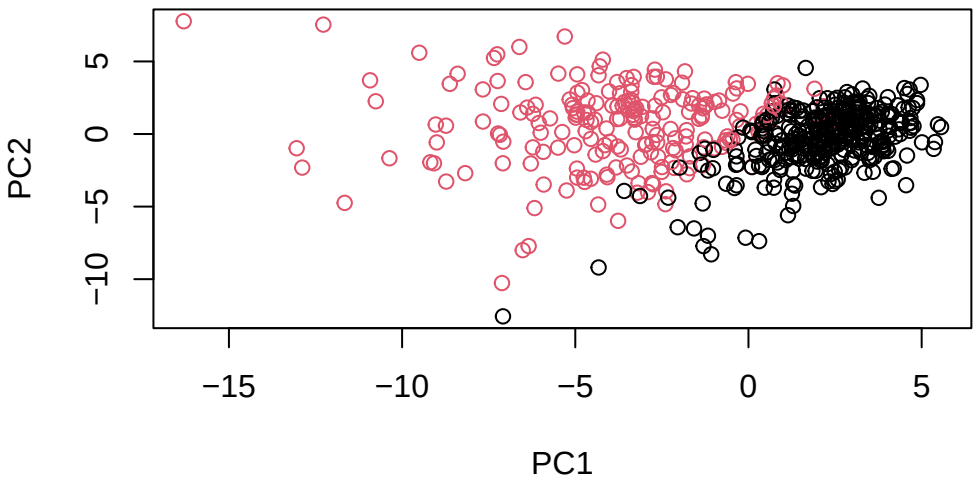
```
Importance of components:
                          PC1    PC2     PC3     PC4     PC5     PC6     PC7
Standard deviation     3.6444 2.3857 1.67867 1.40735 1.28403 1.09880 0.82172
Proportion of Variance 0.4427 0.1897 0.09393 0.06602 0.05496 0.04025 0.02251
Cumulative Proportion  0.4427 0.6324 0.72636 0.79239 0.84734 0.88759 0.91010
                           PC8    PC9    PC10   PC11    PC12    PC13    PC14
Standard deviation     0.69037 0.6457 0.59219 0.5421 0.51104 0.49128 0.39624
Proportion of Variance 0.01589 0.0139 0.01169 0.0098 0.00871 0.00805 0.00523
Cumulative Proportion  0.92598 0.9399 0.95157 0.9614 0.97007 0.97812 0.98335
                          PC15    PC16    PC17    PC18    PC19    PC20   PC21
Standard deviation     0.30681 0.28260 0.24372 0.22939 0.22244 0.17652 0.1731
Proportion of Variance 0.00314 0.00266 0.00198 0.00175 0.00165 0.00104 0.0010
Cumulative Proportion  0.98649 0.98915 0.99113 0.99288 0.99453 0.99557 0.9966
                          PC22    PC23   PC24    PC25    PC26    PC27    PC28
Standard deviation     0.16565 0.15602 0.1344 0.12442 0.09043 0.08307 0.03987
Proportion of Variance 0.00091 0.00081 0.0006 0.00052 0.00027 0.00023 0.00005
Cumulative Proportion  0.99749 0.99830 0.9989 0.99942 0.99969 0.99992 0.99997
                          PC29    PC30
Standard deviation     0.02736 0.01153
Proportion of Variance 0.00002 0.00000
Cumulative Proportion  1.00000 1.00000
```

Q4 44% Q5 4 Q6 0.4427+0.1897+0.09393+0.06602+0.05496+0.04025+0.02251, You need 7 PC

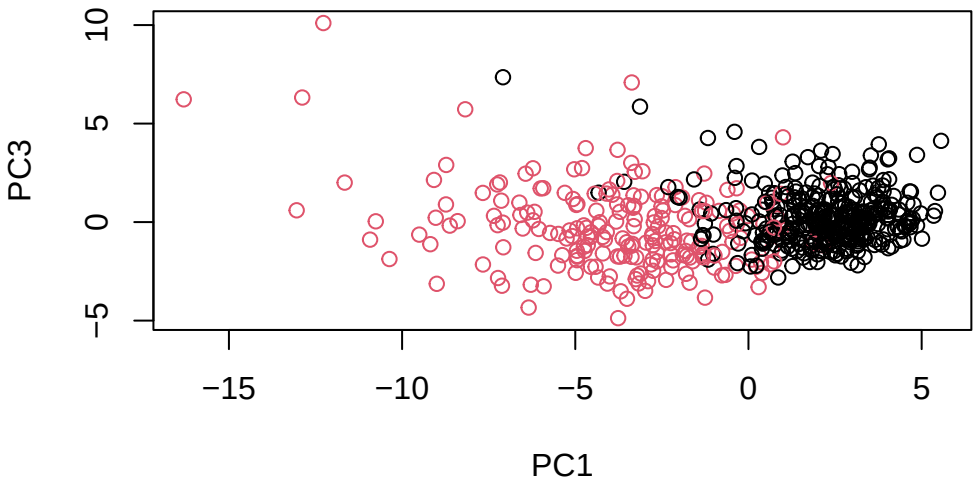
```
biplot(wisc.pr)
```




```
# Scatter plot observations by components 1 and 2
plot(wisc.pr$x[,1],wisc.pr$x[,2], col = diagnosis ,
     xlab = "PC1", ylab = "PC2")
```




```
plot(wisc.pr$x[,1],wisc.pr$x[,3], col = diagnosis,
     xlab = "PC1", ylab = "PC3")
```

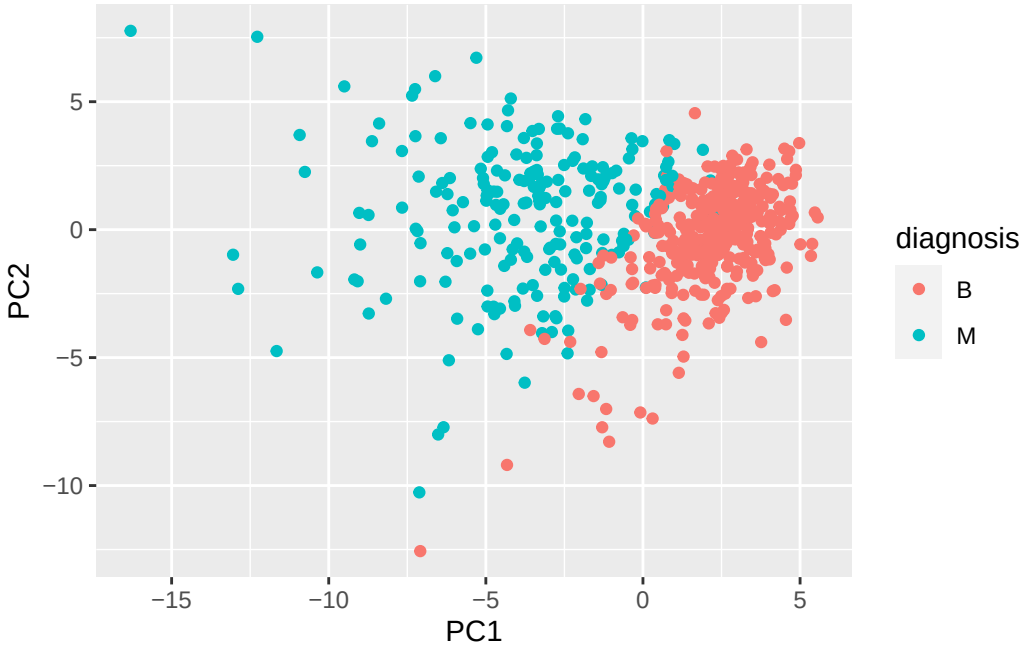


Q8 I notice that the separation of the groups are pretty similar with both plots suggesting that PC1 does a pretty good job

```
# Create a data.frame for ggplot
df <- as.data.frame(wisc.pr$x)
df$diagnosis <- diagnosis

# Load the ggplot2 package
library(ggplot2)

# Make a scatter plot colored by diagnosis
ggplot(df) +
  aes(PC1, PC2, col=diagnosis) +
  geom_point()
```

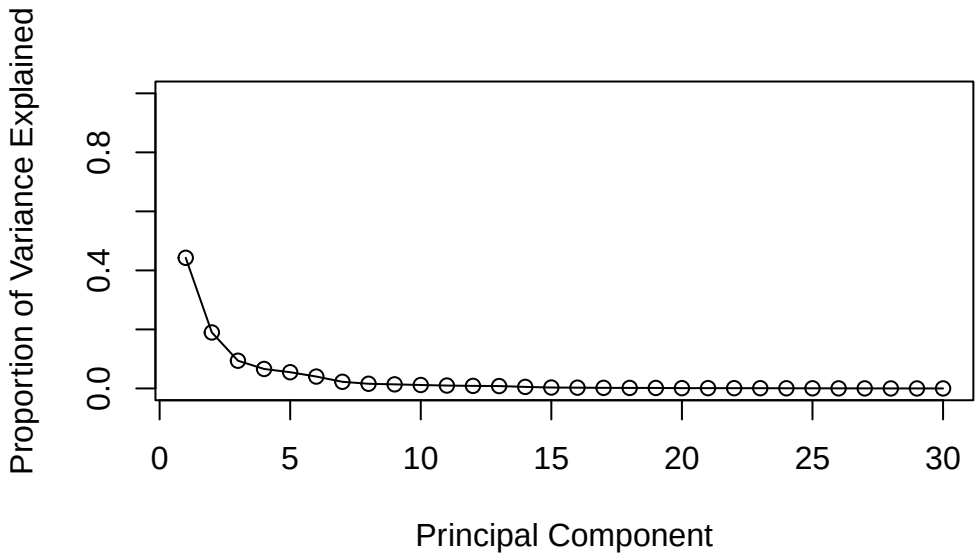



```
# Calculate variance of each component
pr.var <- wisc.pr$sdev^2
head(pr.var)
```

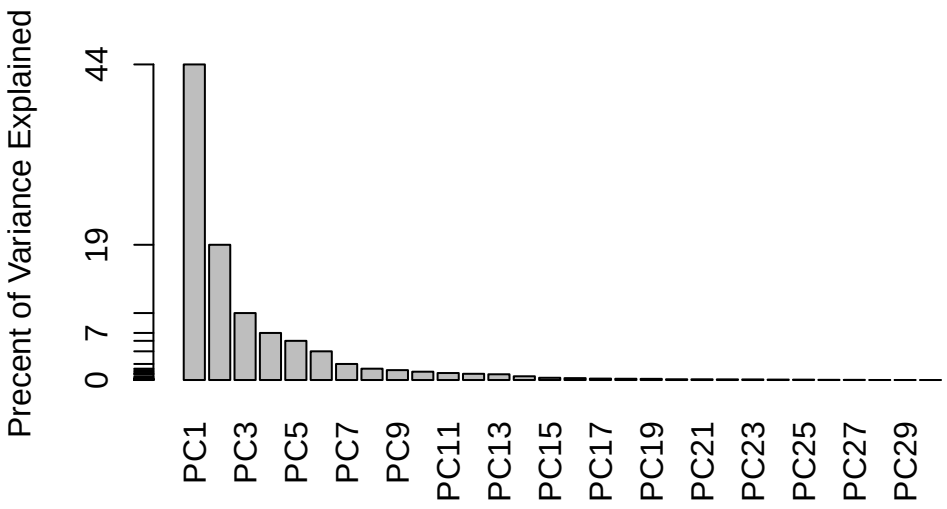
```
[1] 13.281608  5.691355  2.817949  1.980640  1.648731  1.207357
```

```r
# Variance explained by each principal component: pve
pve <- pr.var/sum(pr.var)

# Plot variance explained for each principal component
plot(pve, xlab = "Principal Component",
     ylab = "Proportion of Variance Explained",
     ylim = c(0, 1), type = "o")
```



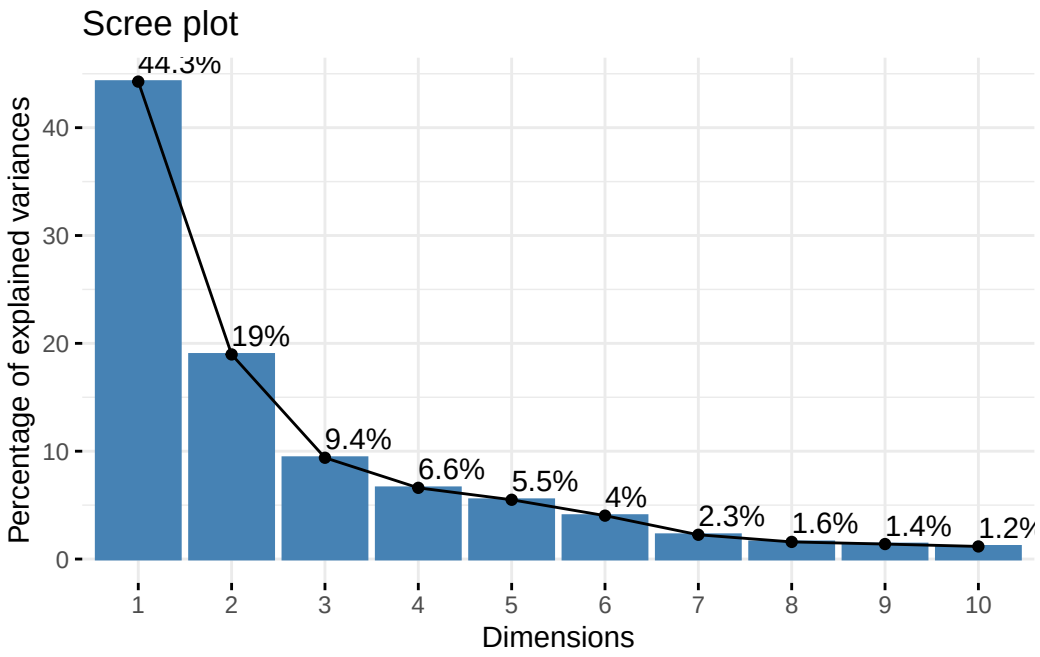

```r
# Alternative scree plot of the same data, note data driven y-axis
barplot(pve, ylab = "Precent of Variance Explained",
     names.arg=paste0("PC",1:length(pve)), las=2, axes = FALSE)
axis(2, at=pve, labels=round(pve,2)*100 )
```




```
## ggplot based graph
##install.packages("factoextra")
library(factoextra)
```

```
Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

```
fviz_eig(wisc.pr, addlabels = TRUE)
```



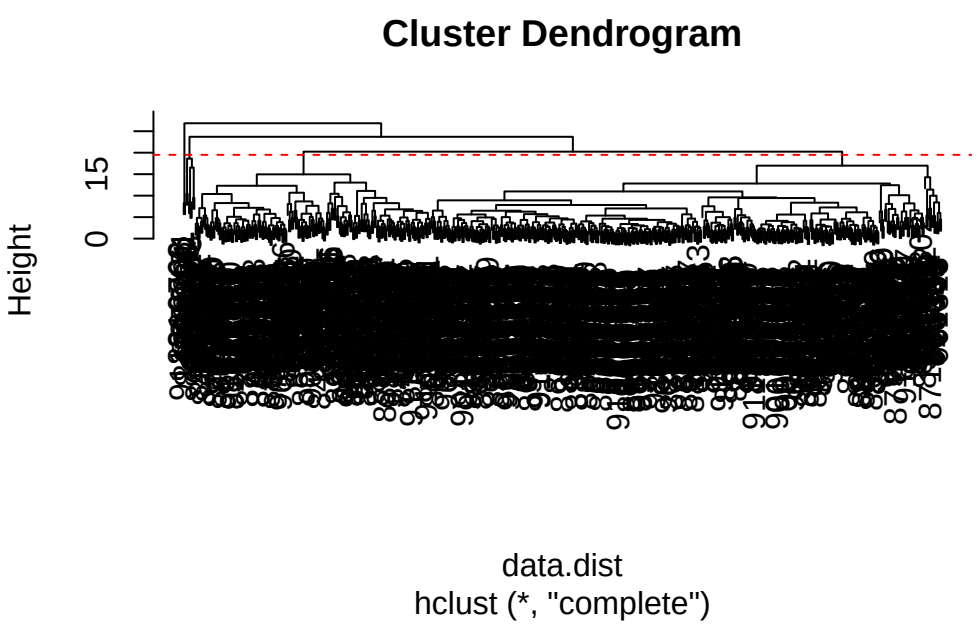

```
wisc.pr$rotation["concave.points_mean",1]
```

```
[1] -0.2608538
```

Q9) -0.2608538 Q10)Based on the screeplot, YOu would need 5 PCA

3. Hierarchal Clustering

```
data.scaled <-scale(wisc.data)
data.dist <- dist(data.scaled)
wisc.hclust <- hclust(data.dist,"complete")
plot(wisc.hclust)
abline(h=19.5, col="red", lty=2)
```



Q11) Between 19 and 20 height there are 4 clusters

```
wisc.hclust.clusters <- cutree(wisc.hclust,k=4)
table(wisc.hclust.clusters, diagnosis)
```

```
                    diagnosis
wisc.hclust.clusters   B   M
                   1  12 165
                   2   2   5
                   3 343  40
                   4   0   2
```

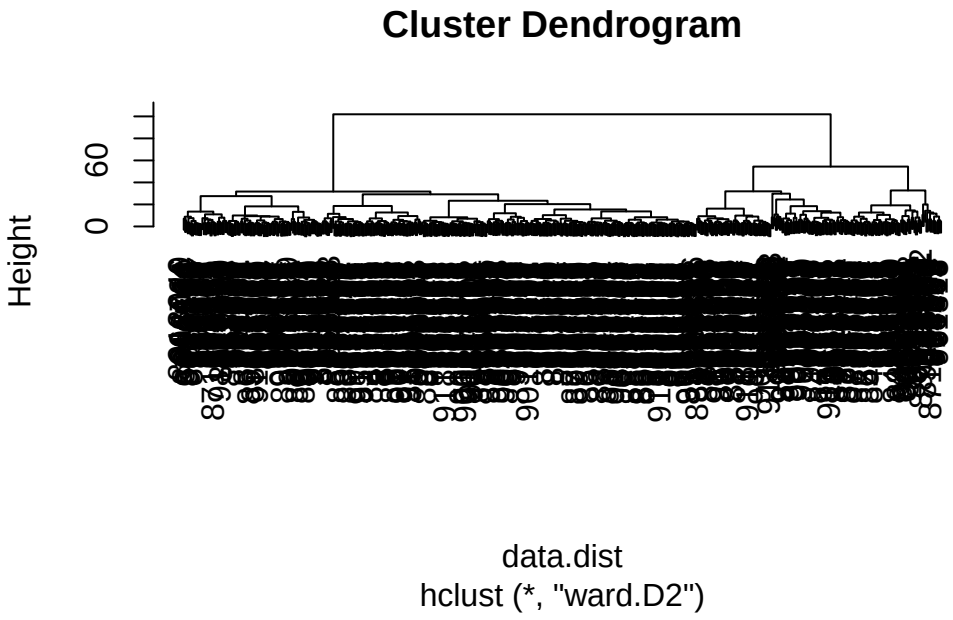
```
wisc.hclust.clusters <- cutree(wisc.hclust,k=6)
table(wisc.hclust.clusters, diagnosis)
```

```
                    diagnosis
wisc.hclust.clusters   B   M
                   1  12 165
                   2   0   5
                   3 331  39
```

```
4   2   0
5  12   1
6   0   2
```

Q12) 4 or 5 seems to be the best amount of clusters.At lower than 4, there is no separation while at above 5 and 6, the additional separation is so marginal and doesn't seem to show any improved impact

```
  wisc.hclust <- hclust(data.dist,"ward.D2")
plot(wisc.hclust)
```




```
wisc.hclust <- hclust(data.dist,"complete")
```

Q13) Ward.D2 seems to produce the best results as it clearly splits the groups into two while the others can't clearly distinguish into two visually obviously separate groups

```
wisc.km <- kmeans(data.scaled, centers=2, nstart= 20)
table(wisc.km$cluster,diagnosis)
```

```
diagnosis
   B   M
```

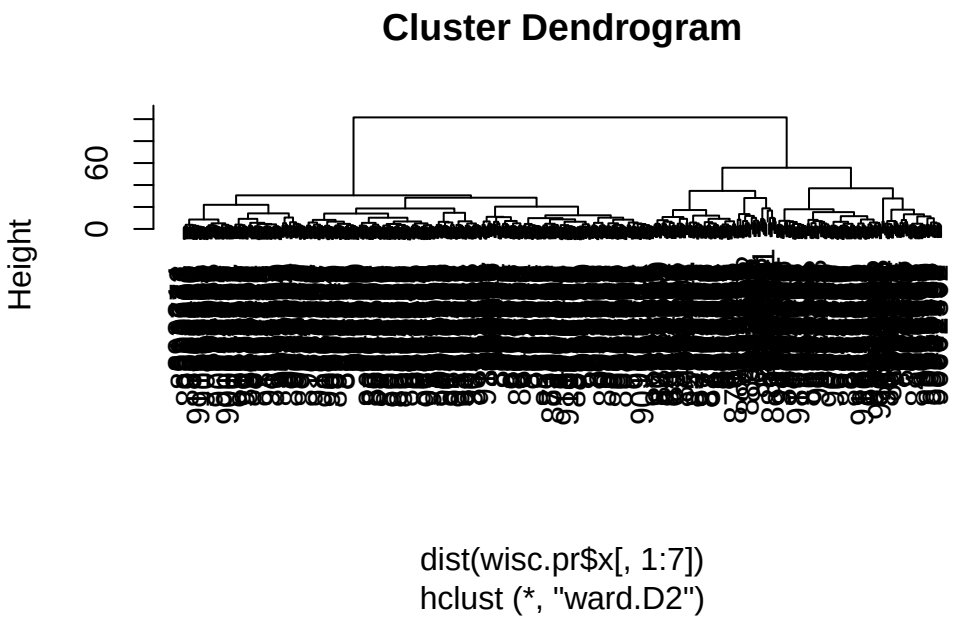
```
  1 343  37
  2  14 175
```

Q14, it separates it pretty well as group 1 seems to be skewed bengign while group 2 seems to be skewed malignant. These numbers are pretty comparable to the hclust results.

```
table(wisc.hclust.clusters,wisc.km$cluster)
```

```
wisc.hclust.clusters   1   2
                   1  17 160
                   2   0   5
                   3 358  12
                   4   0   2
                   5   5   8
                   6   0   2
```

```
wisc.pr.hclust<-hclust(dist(wisc.pr$x[,1:7]),"ward.D2")
plot(wisc.pr.hclust)
```



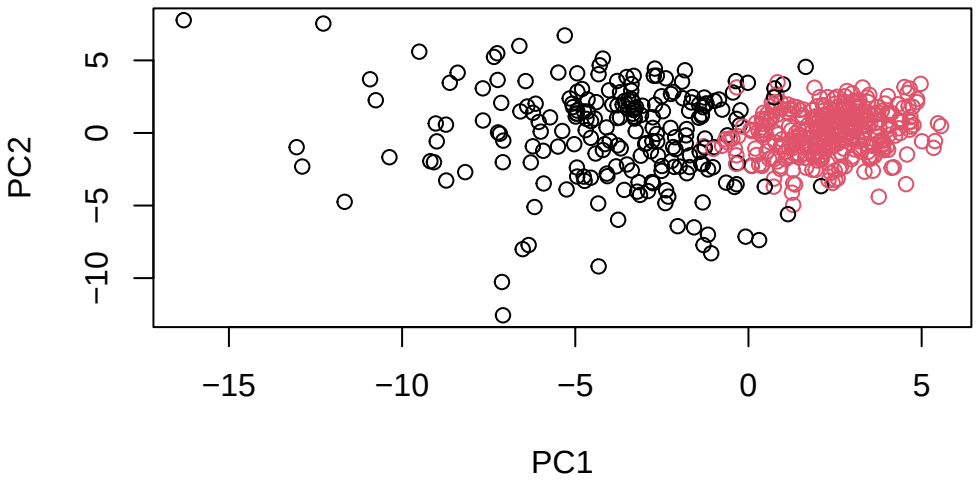
```
grps <- cutree(wisc.pr.hclust, k=2)
table(grps)
```

```
grps
  1   2
216 353
```

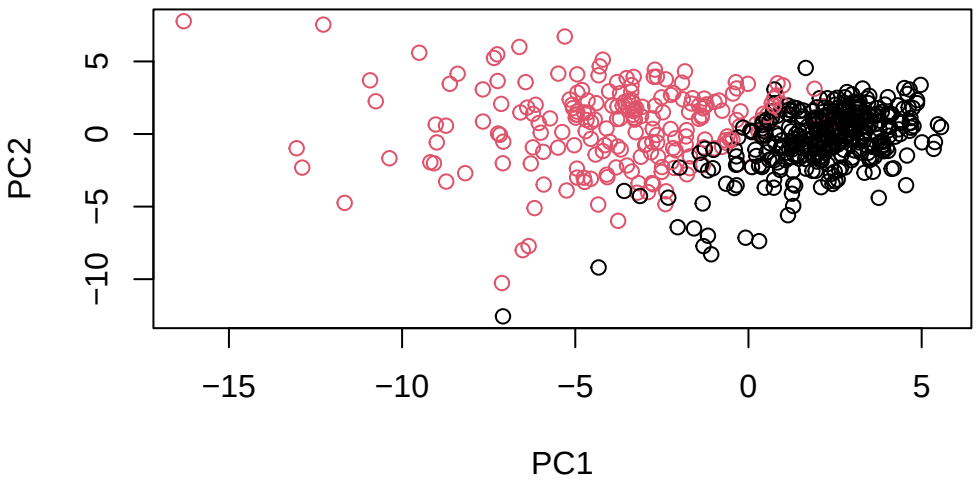
```
table(grps, diagnosis)
```

```
    diagnosis
grps   B   M
   1  28 188
   2 329  24
```

```
plot(wisc.pr$x[,1:2], col=grps)
```




```
plot(wisc.pr$x[,1:2], col=diagnosis)
```



Q15 The new model clearly splits it into two groups despite the 20 outliers. When compared to the hclust.clusters model, it does very comparably.

```
table(wisc.km$cluster, diagnosis)
```

```
   diagnosis
      B   M
  1 343  37
  2  14 175
```

```
table(wisc.hclust.clusters, diagnosis)
```

```
                    diagnosis
wisc.hclust.clusters   B   M
                   1  12 165
                   2   0   5
                   3 331  39
                   4   2   0
                   5  12   1
                   6   0   2
```

Q16) the kmeans and hclust alone do fairly similar in that they can separate them into cleaer two groups but each group will have a fair amount of the other diagnosis. The pca hclust does better as within each group there are less "false positives"

Q17

```
# for Sensitivity
#hclust alone
HclustSens<-165/(165+12)
#kmeans alone
kMeansSens<-175/(175+14)
#pca hclust
pcaHclustSens<-188/(188+28)



# specificity
#hclust alone
HclustSpec<-343/(343+40)
#kmeans alone
kMeansSpec<-343/(343+37)
#pca hclust
pcaHclustSpec<-329/(329+24)
```

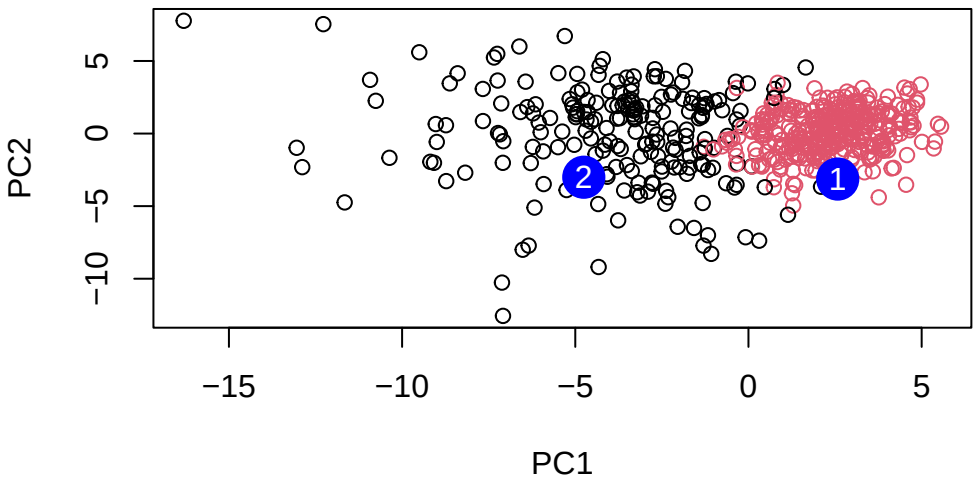
The hclust alone has the greatest sensitivity while the pca hclust has the greatest specificity

```
#url <- "new_samples.csv"
url <- "https://tinyurl.com/new-samples-CSV"
new <- read.csv(url)
npc <- predict(wisc.pr, newdata=new)
npc
```

```
          PC1       PC2        PC3        PC4       PC5        PC6        PC7
[1,]  2.576616 -3.135913  1.3990492 -0.7631950  2.781648 -0.8150185 -0.3959098
[2,] -4.754928 -3.009033 -0.1660946 -0.6052952 -1.140698 -1.2189945  0.8193031
           PC8       PC9       PC10      PC11      PC12      PC13     PC14
[1,] -0.2307350 0.1029569 -0.9272861 0.3411457  0.375921 0.1610764 1.187882
[2,] -0.3307423 0.5281896 -0.4855301 0.7173233 -1.185917 0.5893856 0.303029
         PC15       PC16        PC17        PC18        PC19       PC20
[1,] 0.3216974 -0.1743616 -0.07875393 -0.11207028 -0.08802955 -0.2495216
[2,] 0.1299153  0.1448061 -0.40509706  0.06565549  0.25591230 -0.4289500
          PC21       PC22       PC23       PC24        PC25         PC26
```

```
[1,]  0.1228233 0.09358453 0.08347651  0.1223396  0.02124121  0.078884581
[2,] -0.1224776 0.01732146 0.06316631 -0.2338618 -0.20755948 -0.009833238
             PC27        PC28         PC29         PC30
[1,]  0.220199544 -0.02946023 -0.015620933  0.005269029
[2,] -0.001134152  0.09638361  0.002795349 -0.019015820
```

```r
plot(wisc.pr$x[,1:2], col=grps)
points(npc[,1], npc[,2], col="blue", pch=16, cex=3)
text(npc[,1], npc[,2], c(1,2), col="white")
```



Q18) The groups coloring is produced from cutting the pca hclust. From the previous question we know that the pca hclust has low sensitivity but high specificity. This means that it is worse at identifying ill patients compared to its ability to reject healthy patients. This would mean we should be more worried about patient 2 as the patients in the black group represent the ones the pca hclust assigned as malignant.