class08

AUTHOR
Ziheng Huang

1. Exploratory data analysis

Preparing data:

```
fna.data <- "WisconsinCancer.csv"
wisc.df <- read.csv(fna.data, row.names=1)
wisc.data <- wisc.df[,-1]
diagnosis <- wisc.df$diagnosis
diagnosis <- as.factor(diagnosis)
```

Explore data:

Q1. How many observations are in this dataset? 569

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

```
[1] 569  30
```

Q2. How many of the observations have a malignant diagnosis? 212

```
table(diagnosis)
```

```
diagnosis
  B   M 
357 212 
```

```
sum(diagnosis == 'M')
```

```
[1] 212
```

Q3. How many variables/features in the data are suffixed with `_mean`? 10

```
length(grep('_mean',colnames(wisc.data)))
```

```
[1] 10
```

2. Principal Component Analysis

Performing PCA

```r
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 
```

```r
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 
```

```r
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. From your results, what proportion of the original variance is captured by the first principal components (PC1)?

```
0.4427
```

Q5. How many principal components (PCs) are required to describe at least 70% of the original variance in the data?

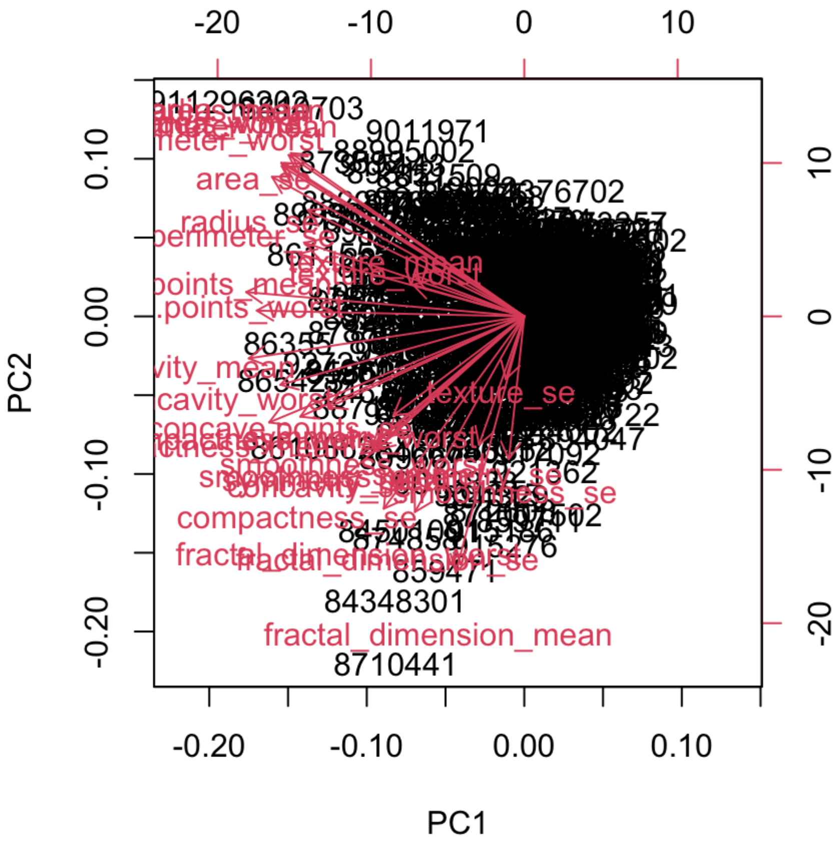
```
first 3 PCs explains 0.72636
```

Q6. How many principal components (PCs) are required to describe at least 90% of the original variance in the data?

```
first 7 PCs explains 0.91010
```

Interpreting PCA results

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

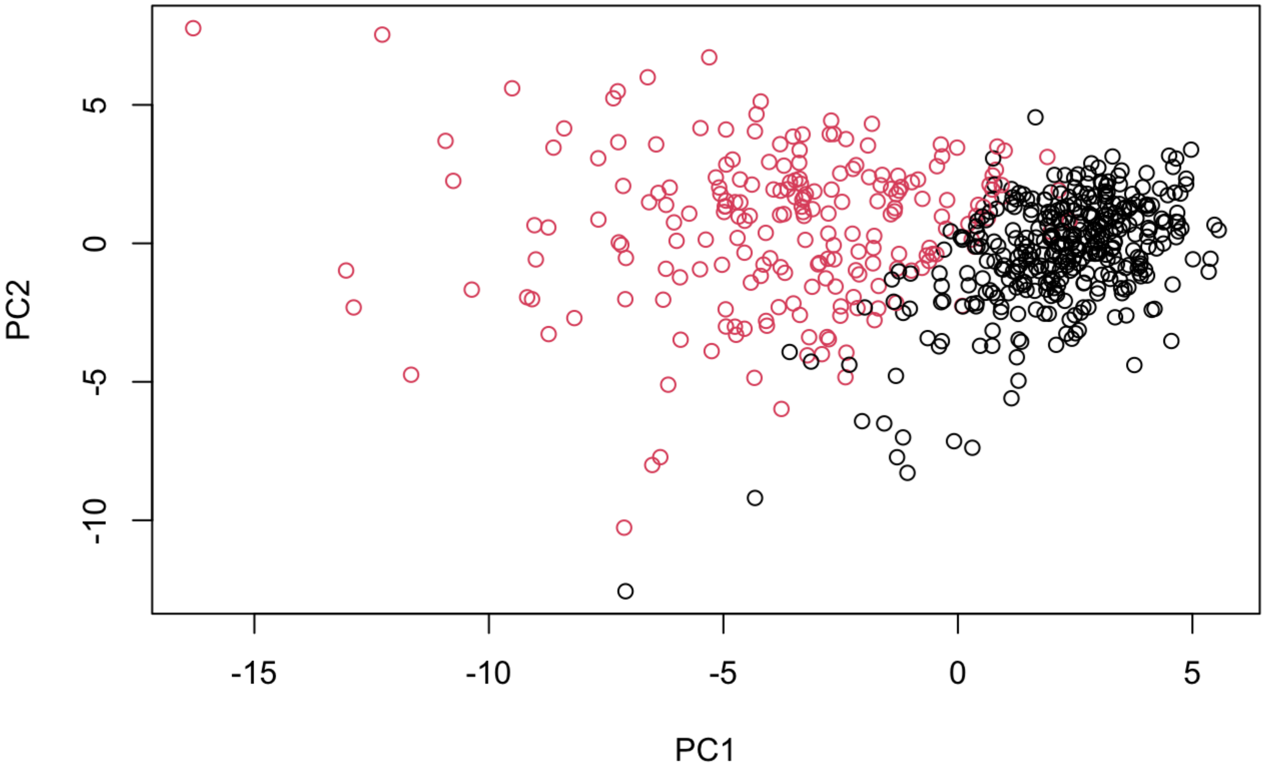


Q7. What stands out to you about this plot? Is it easy or difficult to understand? Why?

It’s a mess. very hard to see.

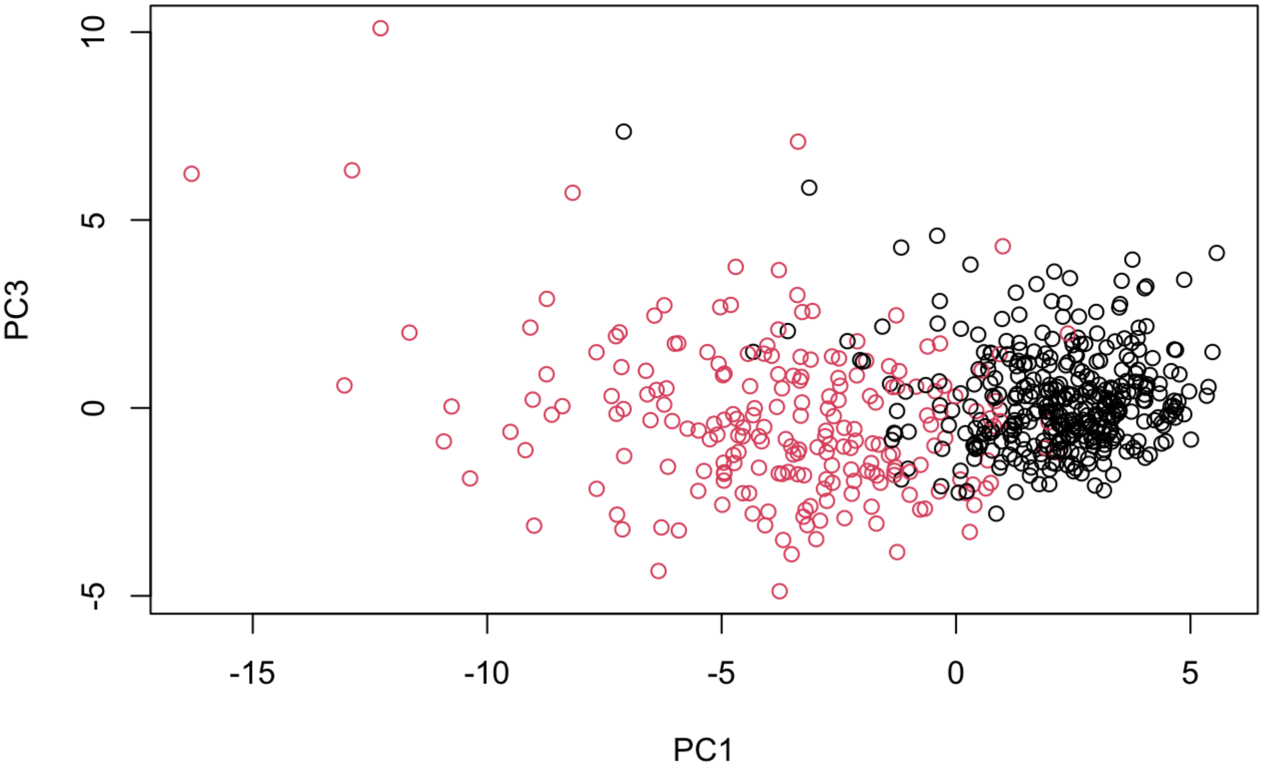
Plot: PC1 vs PC2

```
plot( wisc.pr$x[,1], wisc.pr$x[,2] , col =diagnosis , xlab = "PC1", ylab = "PC2")
```



Plot: PC1 vs PC3

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



Q8. Generate a similar plot for principal components 1 and 3. What do you notice about these plots?

There is pattern in the data that splits the patients.

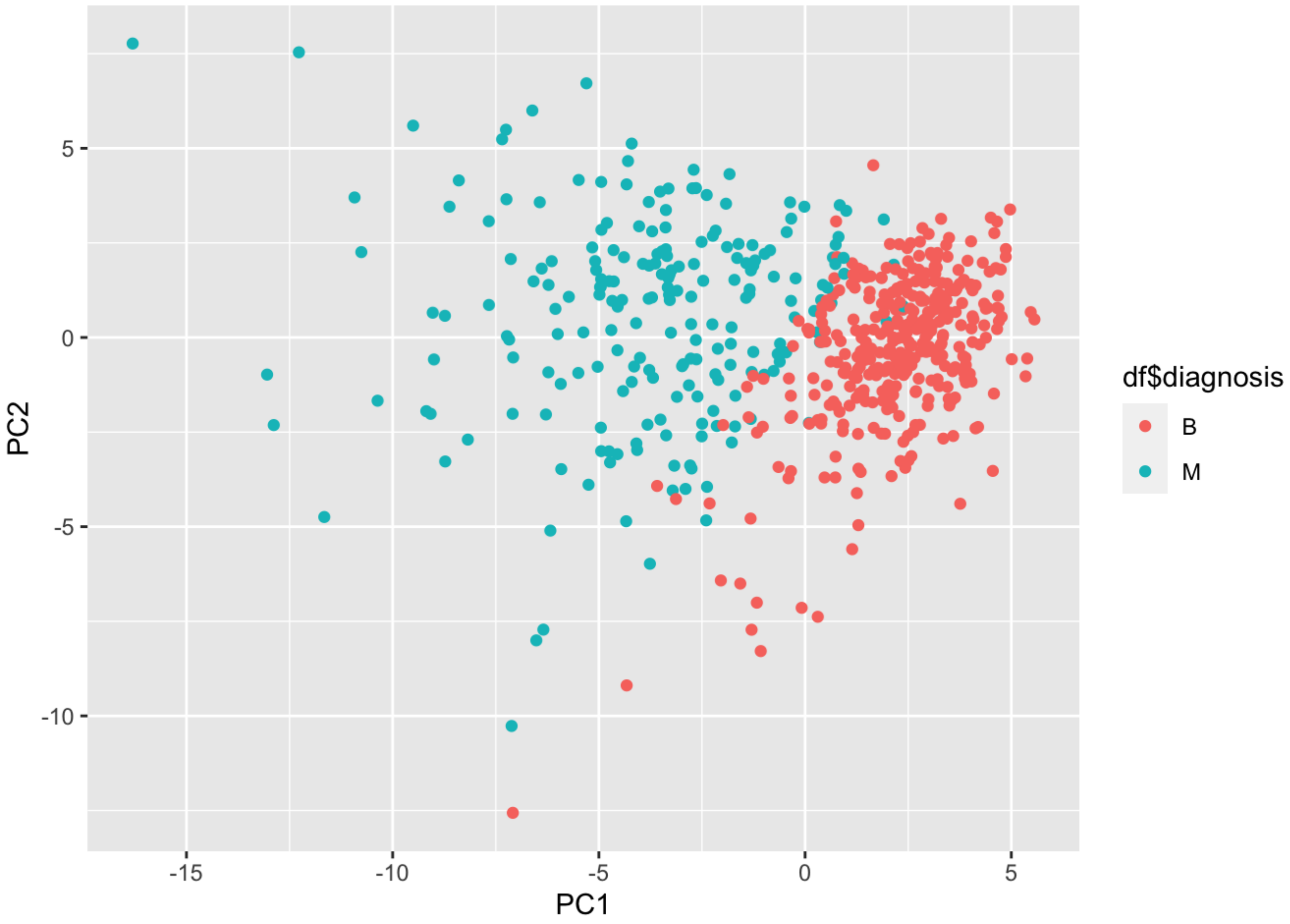
GGPLOT:

```
df <- as.data.frame(wisc.pr$x)
df$diagnosis <- diagnosis

library(ggplot2)

ggplot(df) +
  aes(PC1, PC2, col=df$diagnosis) +
  geom_point()
```

```
Warning: Use of `df$diagnosis` is discouraged. Use `diagnosis` instead.
```

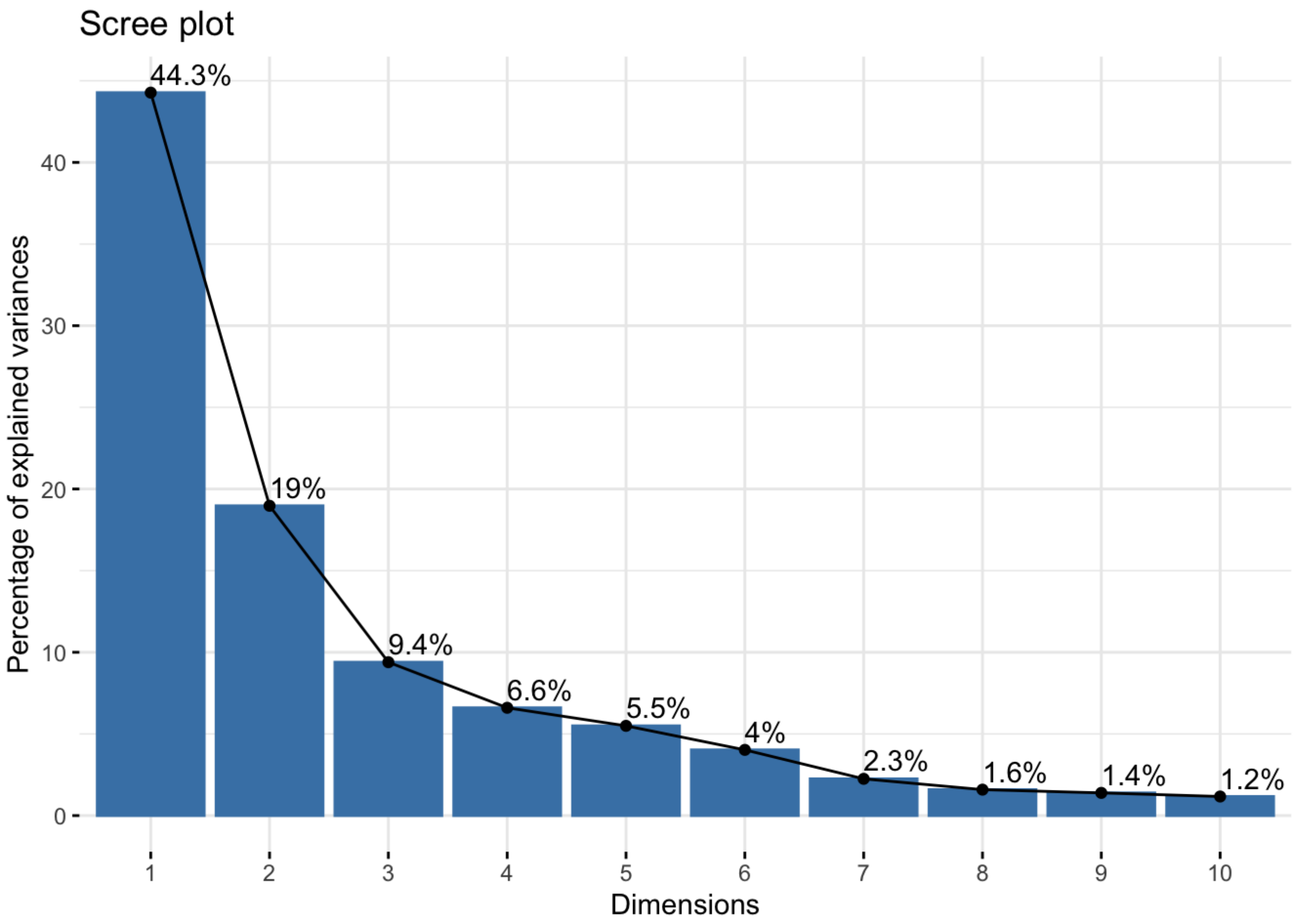


Variance explained

```r
library(factoextra)
```

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

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



Q9. For the first principal component, what is the component of the loading vector

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

```
[1] -0.2608538
```

Q10. What is the minimum number of principal components required to explain 80% of the variance of the data?

You need the first 5 PCs at least to explain 80% of the variance.

```
44.3 + 19 + 9.4 + 6.6 + 5.5
```

```
[1] 84.8
```

3. Hierarchical clustering

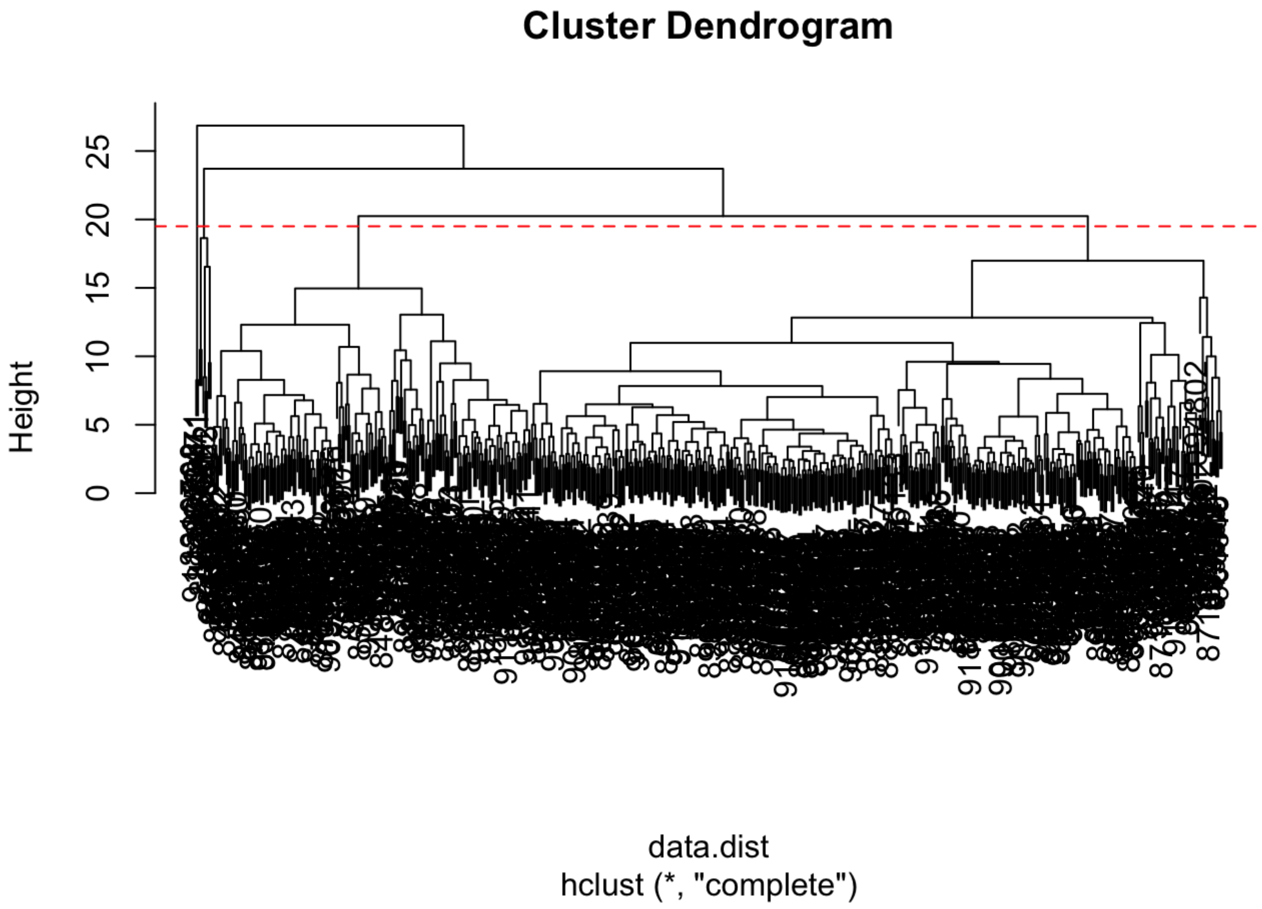
set up clustering

```
data.scaled <- scale(wisc.data)
data.dist <- dist(data.scaled)
wisc.hclust <- hclust(data.dist)
```

Results of hierarchical clustering

Q11. Using the plot() and abline() functions, what is the height at which the clustering model has 4 clusters? 19.5 lead to 4 clusters

```
plot(wisc.hclust)
abline(h=19.5, col="red", lty=2)
```



Selecting number of 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
```

Q12. Can you find a better cluster vs diagnoses match by cutting into a different number of clusters between 2 and 10?

The below is the result for 3 and 5 clusters. I also tried 6,7,8,9. I didn’t see the result being improved a lot.

```
wisc.hclust.clusters <- cutree(wisc.hclust, k=3)
table(wisc.hclust.clusters, diagnosis)
```

```
                    diagnosis
wisc.hclust.clusters   B   M
                   1 355 205
                   2   2   5
                   3   0   2
```

```
wisc.hclust.clusters <- cutree(wisc.hclust, k=5)
table(wisc.hclust.clusters, diagnosis)
```

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

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

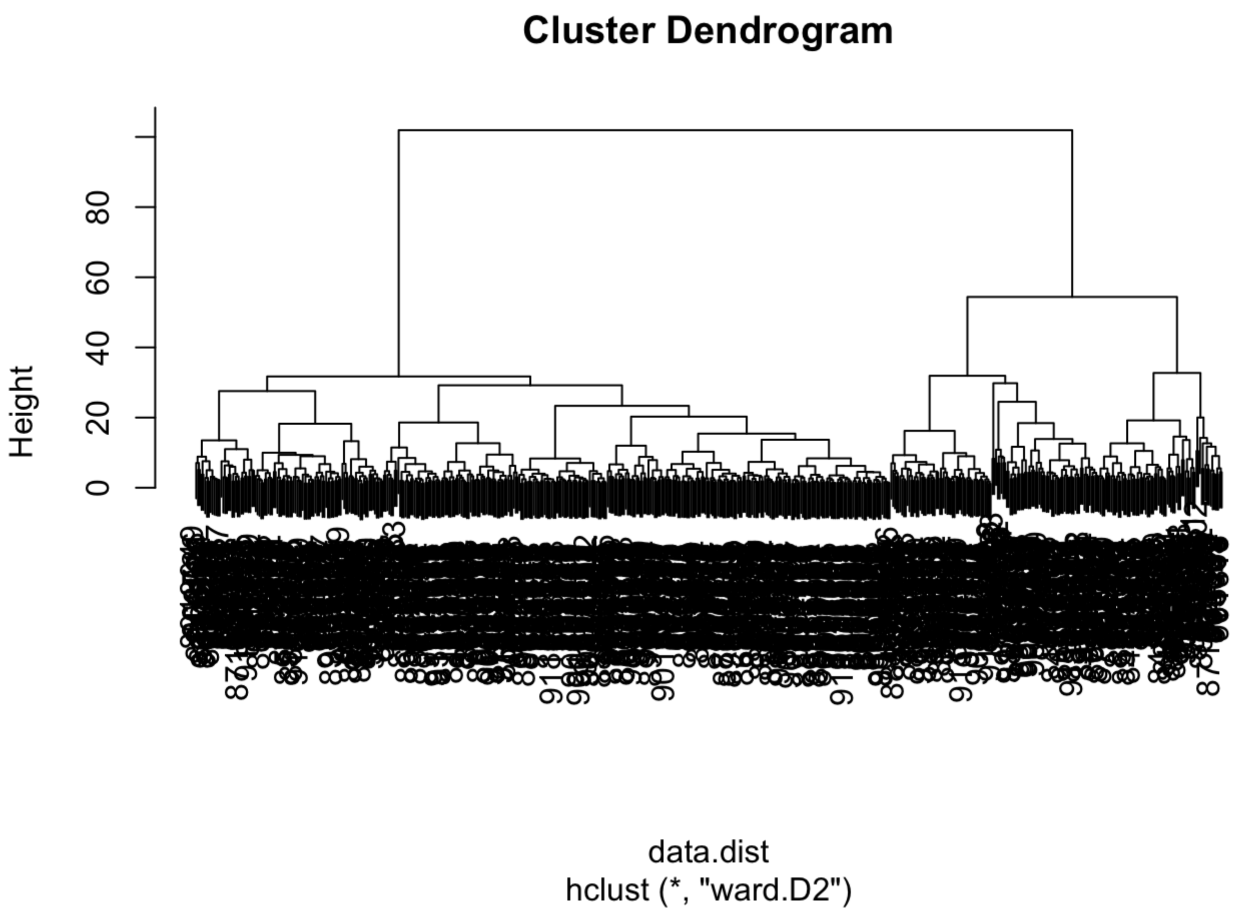
```
                    diagnosis
wisc.hclust.clusters   B   M
                   1  12 165
                   2   2   5
                   3 343  40
                   4   0   2
```

Using different methods

Q13. Which method gives your favorite results for the same data.dist dataset? Explain your reasoning.

I like the ward.D2 the most since it gave a much cleaner split for 2 clusters.

```
plot(hclust(data.dist,method="ward.D2"))
```



5. Combining methods

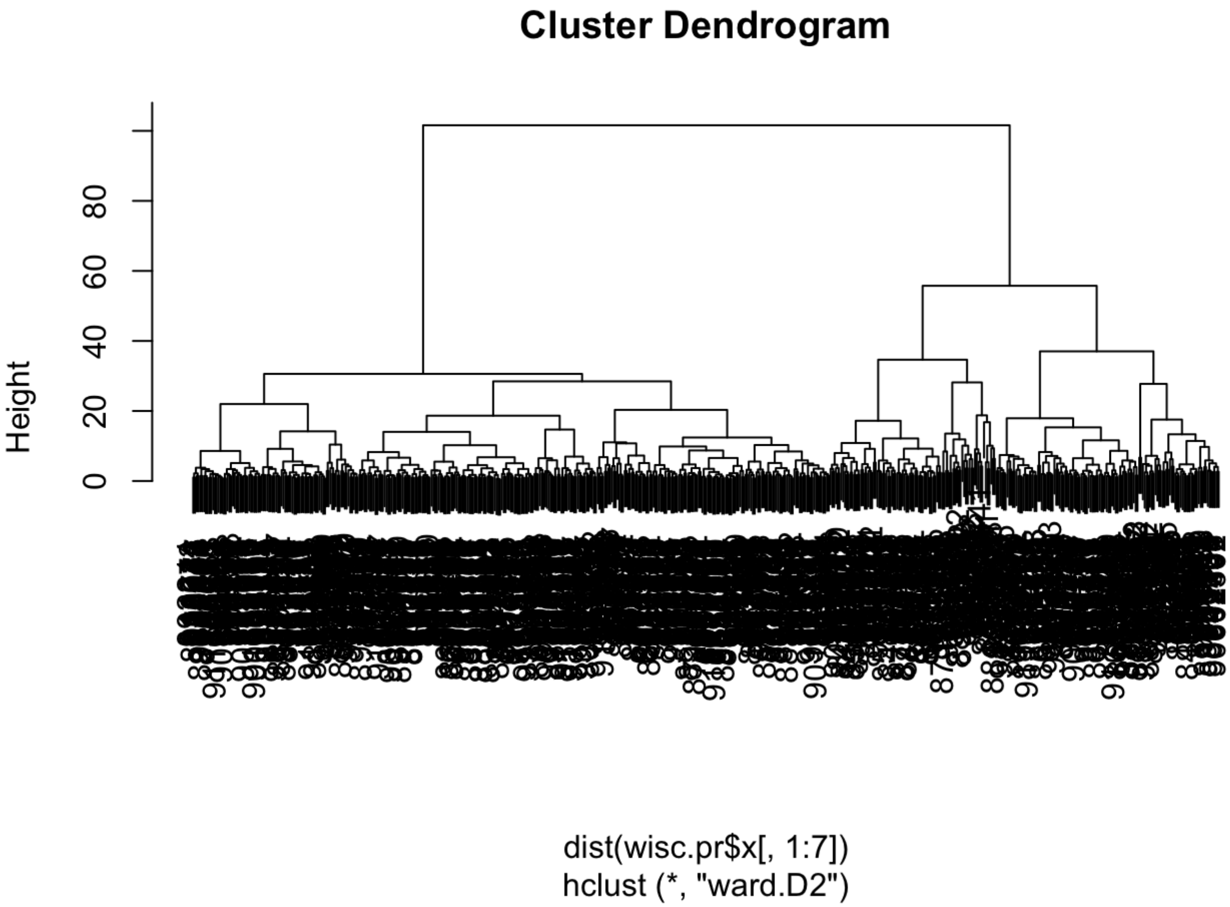
Clustering on PCA results

set up hclust with the first 7 PCs (explains > 90% of variability):

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

hclust visualization:

```
plot(wisc.pr.hclust)
```



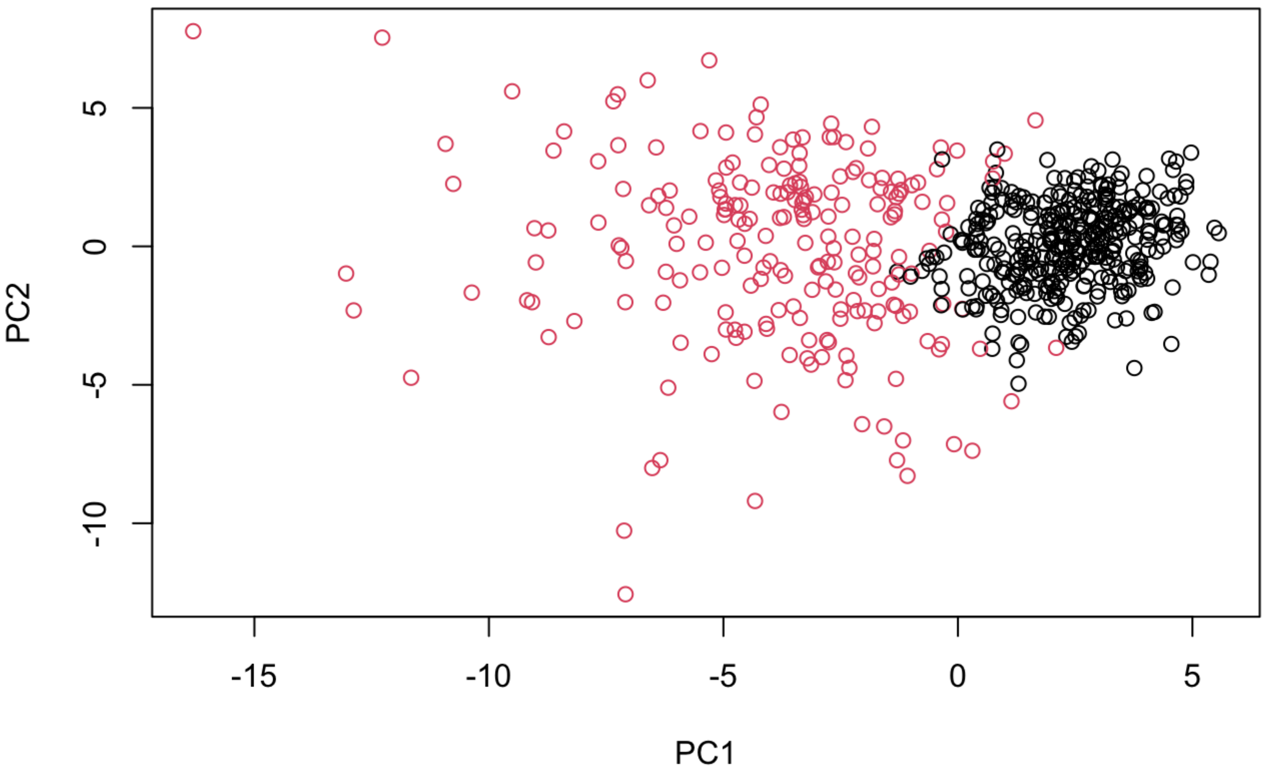
hclust clusters:

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

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

Using clusters from hclust to plot the graph:

```
g <- as.factor(grps)
#levels(g)
g <- relevel(g,2)
#levels(g)
plot(wisc.pr$x[,1:2], col=g)
```



Q15. How well does the newly created model with four clusters separate out the two diagnoses?

cut to 4 clusters:

```
table(cutree(wisc.pr.hclust, k=4), diagnosis)
```

```
   diagnosis
      B   M
  1   0  45
  2   2  77
  3  26  66
  4 329  24
```

Cut with 2 clusters

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

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

I think the new model cut with 4 clusters is worse in separating the diagnosis as shown in above 2 cells.

Q16. How well do the k-means and hierarchical clustering models you created in previous sections (i.e. before PCA) do in terms of separating the diagnoses? Again, use the table() function to compare the output of each model (wisc.km$cluster and wisc.hclust.clusters) with the vector containing the actual diagnoses.

actual diagnosis:

```
table(diagnosis)
```

```
diagnosis
  B   M 
357 212 
```

Hierarchical:

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

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

clustering with PCA:

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

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

Clustering with PCA yielded the best separation in tw0 clusters. using hierachical clustering directly on pre-PCA data had a hard time separating two clusters.

Q17. Which of your analysis procedures resulted in a clustering model with the best specificity? How about sensitivity?

Calculations are made base on above 3 tables. For Hierarchical clustering, 2 cluster failed to separate the diagnosis, so 4 is used. Among the 4 clusters, cluster 1(B:12,M:165) is M, cluster 3(B:343,M:40) is B while the other two clusters are seen as outliers. For clustering with PCA, cluster 1(B:28,M:188) is M, cluster 2(B:329,M:24) is B.

specificity:

Hierarchical: (343+40)/357 = 1.072829

clustering with PCA: (329+24)/357 = 0.9887955

sensitivity:

Hierarchical: (12+165)/212 = 0.8349057

clustering with PCA: (28+188)/212 = 1.018868

compared to Hierarchical cluster, clustering with PCA has lower specificity and higher sensitivity