

# Mini Project

Julia Ainsworth

## Table of contents

<b>Downloading data</b>	<b>2</b>
Q1: How many observations are in this data set? . . . . .	3
Q2: How many of the observations have a malignant diagnosis? . . . . .	4
Q3 How many variables/features in the data are suffixed with <code>_mean</code> ? . . . . .	4
<b>Part 2: PCA Component Analysis</b>	<b>4</b>
Q4: From your results, what proportion of the original variance is captured by the first principal components (PC1)? . . . . .	6
Q5. How many principal components (PCs) are required to describe at least 70% of the original variance in the data? . . . . .	6
Q6. How many principal components (PCs) are required to describe at least 90% of the original variance in the data? . . . . .	6
Q7: What stands out to you about this plot? Is it easy or difficult to understand? Why? . . . . .	7
Q8. Generate a similar plot for principal components 1 and 3. What do you notice about these plots? . . . . .	8
Q9. For the first principal component, what is the component of the loading vector (i.e. <code>wisc.pr\$rotation[,1]</code> ) for the feature <code>concave.points_mean</code> ? This tells us how much this original feature contributes to the first PC. . . . .	12
<b>Part 3: Hierarchical clustering</b>	<b>13</b>
Q10. Using the <code>plot()</code> and <code>abline()</code> functions, what is the height at which the clustering model has 4 clusters? . . . . .	13
Q12. Which method gives your favorite results for the same <code>data.dist</code> dataset? Explain your reasoning. . . . .	14
skipping following step for pdf: 3D . . . . .	19
Q13. How well does the newly created model with four clusters separate out the two diagnoses? . . . . .	19

Q14. How well do the 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. . . . .	19
<b>Prediction</b>	<b>20</b>
skipping question 15, optional . . . . .	21
Q16. Which of these new patients should we prioritize for follow up based on your results? . . . . .	21

## Downloading data

First step is to get the data

```
#Downloaded data into working directory from class website
fna.data <- "WisconsinCancer.csv"
wisc.df <- read.csv(fna.data, row.names = 1)
head(wisc.df, n = 3)
```

```
      diagnosis radius_mean texture_mean perimeter_mean area_mean
842302         M      17.99      10.38         122.8      1001
842517         M      20.57      17.77         132.9      1326
84300903        M      19.69      21.25         130.0      1203
      smoothness_mean compactness_mean concavity_mean concave.points_mean
842302         0.11840         0.27760         0.3001         0.14710
842517         0.08474         0.07864         0.0869         0.07017
84300903        0.10960         0.15990         0.1974         0.12790
      symmetry_mean fractal_dimension_mean radius_se texture_se perimeter_se
842302         0.2419         0.07871         1.0950         0.9053         8.589
842517         0.1812         0.05667         0.5435         0.7339         3.398
84300903        0.2069         0.05999         0.7456         0.7869         4.585
      area_se smoothness_se compactness_se concavity_se concave.points_se
842302      153.40         0.006399         0.04904         0.05373         0.01587
842517       74.08         0.005225         0.01308         0.01860         0.01340
84300903      94.03         0.006150         0.04006         0.03832         0.02058
      symmetry_se fractal_dimension_se radius_worst texture_worst
842302      0.03003         0.006193         25.38         17.33
842517      0.01389         0.003532         24.99         23.41
84300903      0.02250         0.004571         23.57         25.53
      perimeter_worst area_worst smoothness_worst compactness_worst
842302          184.6         2019         0.1622         0.6656
```

842517	158.8	1956	0.1238	0.1866
84300903	152.5	1709	0.1444	0.4245
	concavity_worst	concave.points_worst	symmetry_worst	
842302	0.7119	0.2654	0.4601	
842517	0.2416	0.1860	0.2750	
84300903	0.4504	0.2430	0.3613	
	fractal_dimension_worst			
842302	0.11890			
842517	0.08902			
84300903	0.08758			

Then, getting rid of the first column:

```
wisc.data <- wisc.df[,-1]
diagnosis <- wisc.df$diagnosis
as.factor(diagnosis)
```

```
[1] M M M M M M M M M M M M M M M M M M B B B M M M M M M M M M M M M M
[38] B M M M M M M M M B M B B B B B M M B M M B B B B M B M M B B B B M B M M
[75] B M B M M B B B M M B M M M B B B M B B M M B B B B M M B B B B M B B M B B
[112] B B B B B B M M M B M M B B B M M B M B M M B M M B B M B B M B B B B M B
[149] B B B B B B B B M B B B B M M B M B B M M B B M M B B B B M B B M M M B M
[186] B M B B B M B B M M B M M M M B M M M B M B M B B M B M M M M B B M M B B
[223] B M B B B B B M M B B M B B M M B M B B B B M B B B B M B M M M M M M M
[260] M M M M M M M B B B B B B M B M B B M B B M B M M B B B B B B B B B B B
[297] B M B B M B M B B B B B B B B B B B B B M B B B M B M B B B B M M M B B
[334] B B M B M B M B B B M B B B B B B B M M M B B B B B B B B B B M M B M M
[371] M B M M B B B B B M B B B B B M B B B M B B M M B B B B B M B B B B B B
[408] B M B B B B B M B B M B B B B B B B B B B B M B M M B M B B B B B M B B
[445] M B M B B M B M B B B B B B B B M M B B B B B B M B B B B B B B B B M B
[482] B B B B B B M B M B B M B B B B B M M B M B M B B B B M B B M B M B M M
[519] B B B M B B B B B B B B B B M B M M B B B B B B B B B B B B B B B B B
[556] B B B B B B B M M M M M M B
```

Levels: B M

**Q1: How many observations are in this data set?**

```
nrow(wisc.data)
```

```
[1] 569
```

There are 569 observations in the data set.

## Q2: How many of the observations have a malignant diagnosis?

```
table(diagnosis)
```

```
diagnosis
  B    M
357 212
```

There are 212 malignant diagnoses

## Q3 How many variables/features in the data are suffixed with \_\_mean?

```
length(grep(pattern = "mean", x = as.character(variable.names(wisc.df))))
```

```
[1] 10
```

10 of the variables in the data are suffixed with \_\_mean

## Part 2: PCA Component Analysis

Checking the column means and standard deviation

```
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

```
#Performing PCA on wisc.data; scaling here IS appropriate
wisc.pr <- prcomp(wisc.data, scale. = TRUE)
```

```
#Next, summarizing results:
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)?**

A: 44.27%

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

PC1-PC3

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

7 PCs!

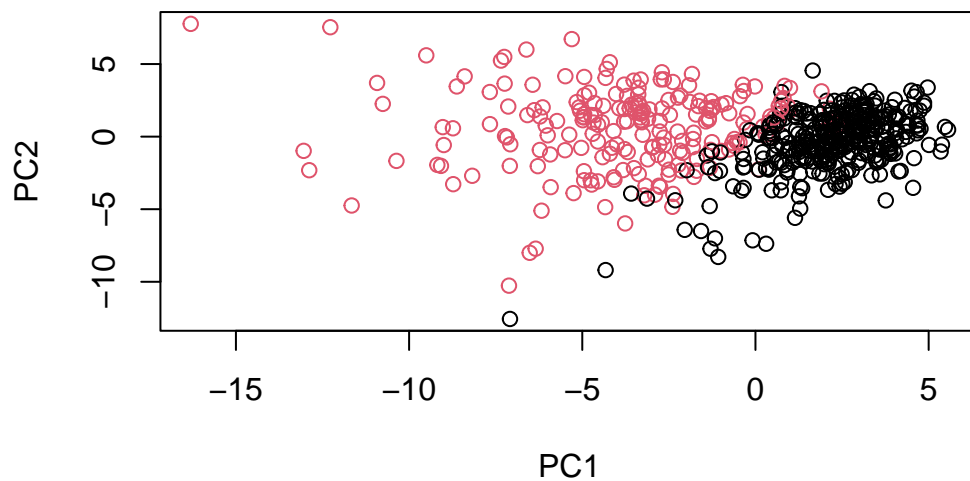
Interpreting the results of the PCA:

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



```
[445] M B M B B M B M B B B B B B B M M B B B B B B M B B B B B B B B B M B
[482] B B B B B B M B M B B M B B B B M M B M B M B B B B M B B M B M B M M
[519] B B B M B B B B B B B B B B M B M M B B B B B B B B B B B B B B B B
[556] B B B B B B B M M M M M M B
Levels: B M
```

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

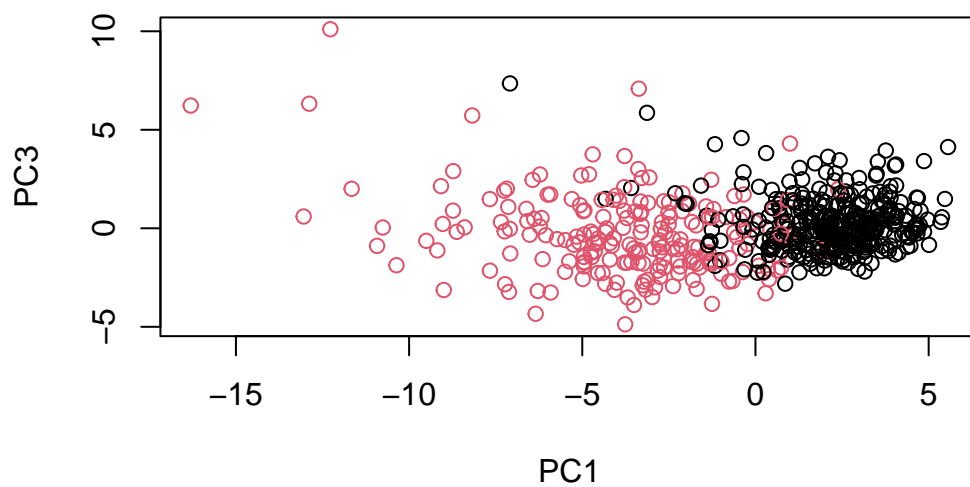


**Q8. Generate a similar plot for principal components 1 and 3. What do you notice about these plots?**

There is less separation than between PC1 and PC2.

```
# PC 1 and 3
plot(x = wisc.pr$x[,1], y = wisc.pr$x[,3], col = as.factor(diagnosis),
     xlab = "PC1", ylab = "PC3")
```

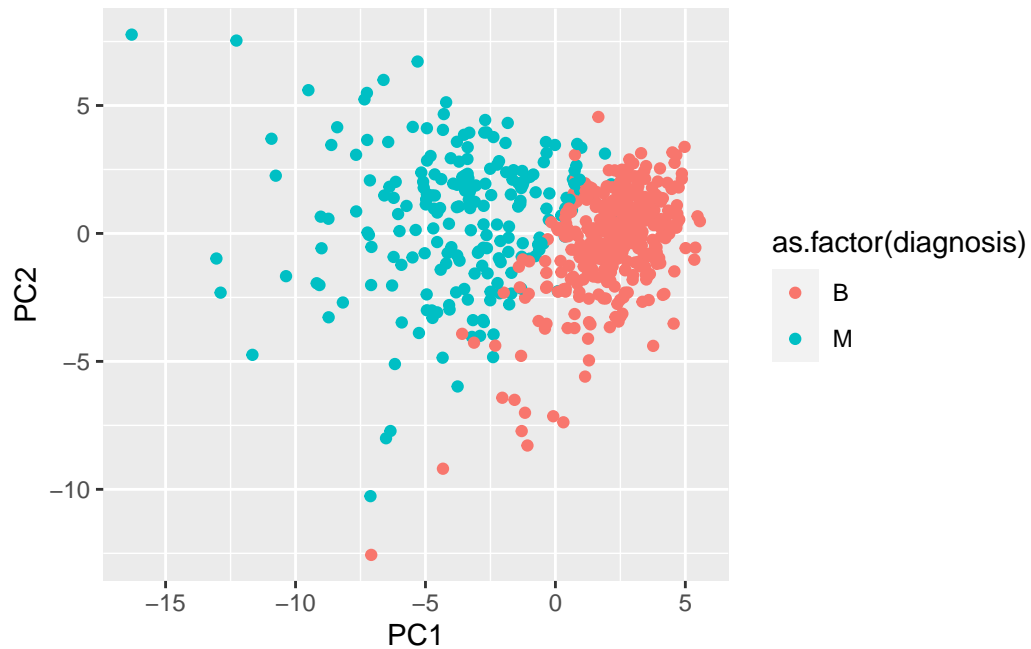




Next step is to go to ggplot to make it look better

```
#Creating a data.frame
df <- as.data.frame(wisc.pr$x)
df$diagnosis <- diagnosis
library(ggplot2)

#Making a plot
ggplot(df) +
  aes(PC1, PC2, col = as.factor(diagnosis)) +
  geom_point()
```

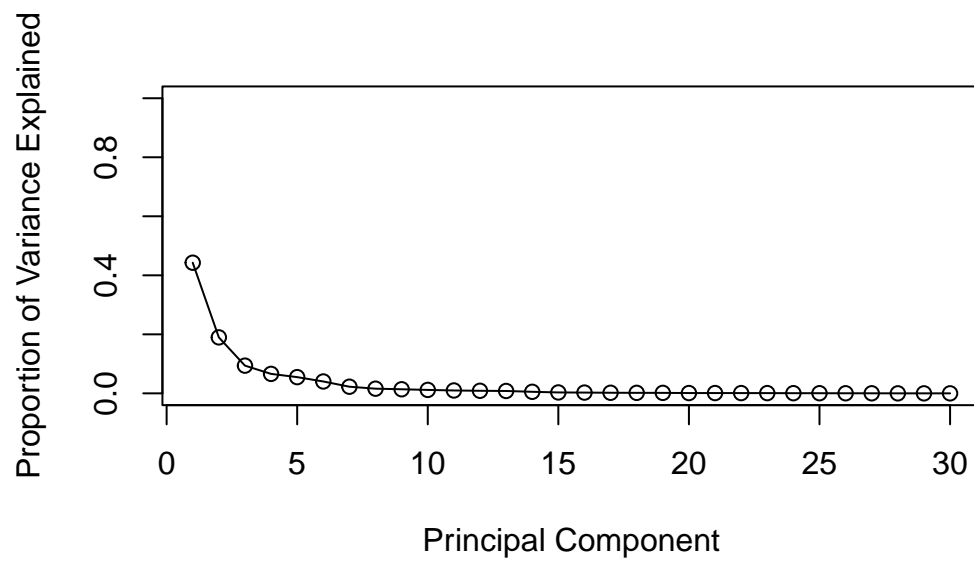


Now, we will look at the scree plots to see the proportion of variance

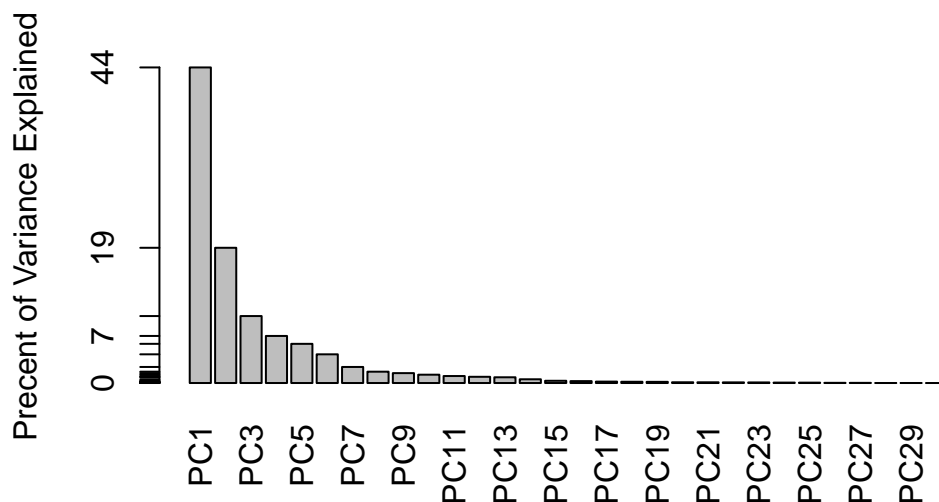
```
#Calculating the variance
pr.var <- wisc.pr$sdev^2
head(pr.var)
```

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

```
#Variance explained by each principal component (30 total): pve
pve <- pr.var/30
plot(pve, xlab = "Principal Component",
     ylab = "Proportion of Variance Explained",
     ylim = c(0,1), type = "o")
```



```
# 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 )
```



**Q9. For the first principal component, what is the component of the loading vector (i.e. `wisc.pr$rotation[,1]`) for the feature `concave.points_mean`? This tells us how much this original feature contributes to the first PC.**

```
wisc.pr$rotation[,1]
```

radius_mean	texture_mean	perimeter_mean
-0.21890244	-0.10372458	-0.22753729
area_mean	smoothness_mean	compactness_mean
-0.22099499	-0.14258969	-0.23928535
concavity_mean	concave.points_mean	symmetry_mean
-0.25840048	-0.26085376	-0.13816696
fractal_dimension_mean	radius_se	texture_se
-0.06436335	-0.20597878	-0.01742803
perimeter_se	area_se	smoothness_se
-0.21132592	-0.20286964	-0.01453145
compactness_se	concavity_se	concave.points_se
-0.17039345	-0.15358979	-0.18341740
symmetry_se	fractal_dimension_se	radius_worst
-0.04249842	-0.10256832	-0.22799663
texture_worst	perimeter_worst	area_worst

-0.10446933	-0.23663968	-0.22487053
smoothness_worst	compactness_worst	concavity_worst
-0.12795256	-0.21009588	-0.22876753
concave.points_worst	symmetry_worst	fractal_dimension_worst
-0.25088597	-0.12290456	-0.13178394

Q9 answer: -0.26085376

## Part 3: Hierarchical clustering

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

The Euclidean distance between all pairs of observations:

```
data.dist <- dist(data.scaled)
```

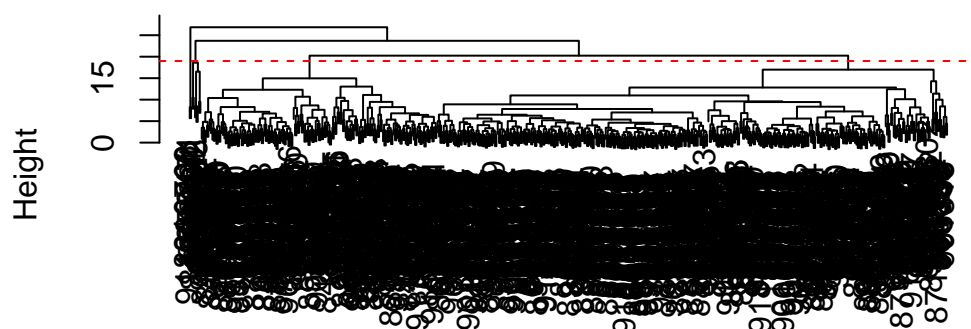
Hierarchical clustering model

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

**Q10. Using the `plot()` and `abline()` functions, what is the height at which the clustering model has 4 clusters?**

```
plot(wisc.hclust)
abline(h = 19, col = "red", lty = 2)
```

## Cluster Dendrogram



data.dist  
hclust (\*, "complete")

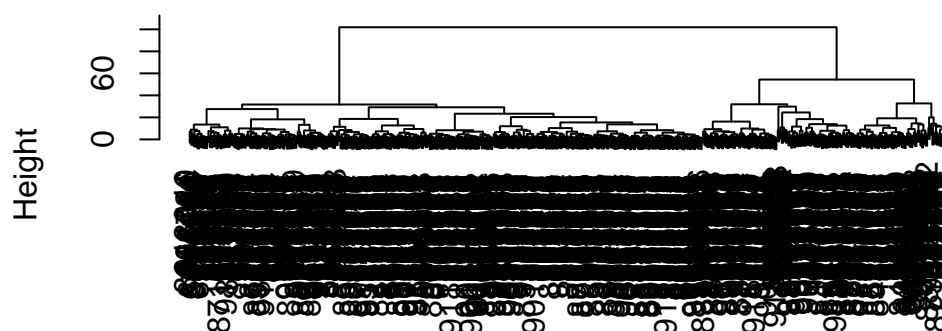
```
wisc.hclust.clusters <- cutree(wisc.hclust, k = 4)
table(wisc.hclust.clusters, diagnosis)
```

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

**Q12. Which method gives your favorite results for the same data.dist dataset?  
Explain your reasoning.**

```
wisc.hclust2 <- hclust(data.dist, method = "ward.D2")
plot(wisc.hclust2)
```

## Cluster Dendrogram



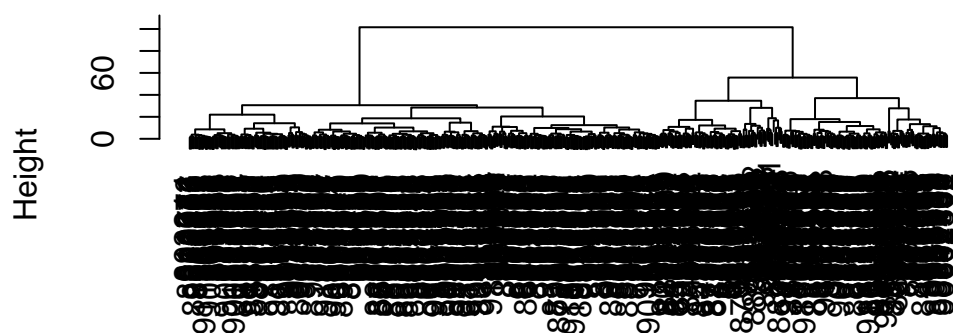
```
data.dist  
hclust (*, "ward.D2")
```

Answer Q12: My favorite result was also with ward.D2, as it seemed like the neatest/easiest to interpret

Clustering on PCA results

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

## Cluster Dendrogram



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

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

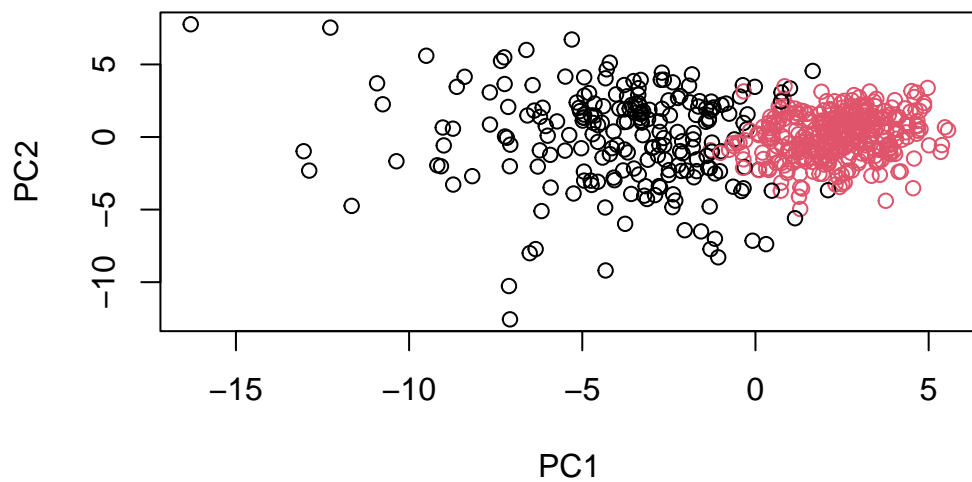
```
grps
  1   2
216 353
```

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

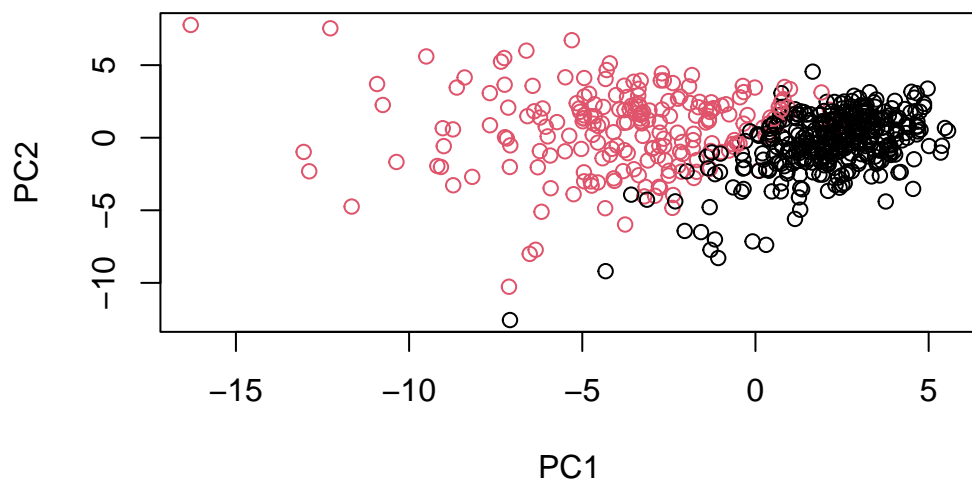
```
      diagnosis
grps   B     M
  1   28  188
  2  329   24
```

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





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



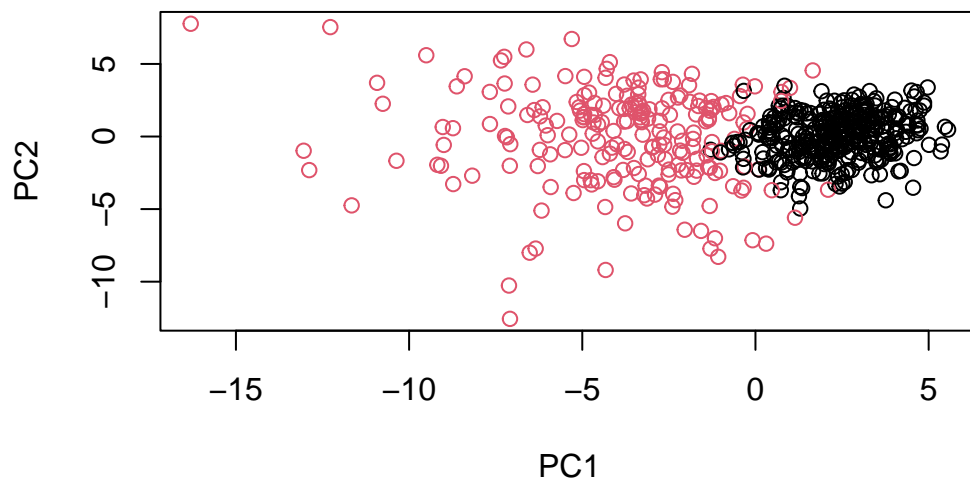
```
g <- as.factor(grps)
levels(g)
```

```
[1] "1" "2"
```

```
g <- relevel(g,2)
levels(g)
```

```
[1] "2" "1"
```

```
#Now plotting with reordered factor
plot(wisc.pr$x[,1:2], col =g)
```



skipping following step for pdf: 3D

```
#Clustering along the first 7 PCs
wisc.pr.hclust <- hclust(dist(wisc.pr$x[, 1:7]), method="ward.D2")
wisc.pr.hclust.clusters <- cutree(wisc.pr.hclust, k =2)
table(wisc.pr.hclust.clusters)
```

```
wisc.pr.hclust.clusters
 1  2
216 353
```

**Q13. How well does the newly created model with four clusters separate out the two diagnoses?**

Comparing to actual diagnoses:

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

```
              diagnosis
wisc.pr.hclust.clusters  B  M
1      28 188
2     329  24
```

**Q14. How well do the 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.**

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

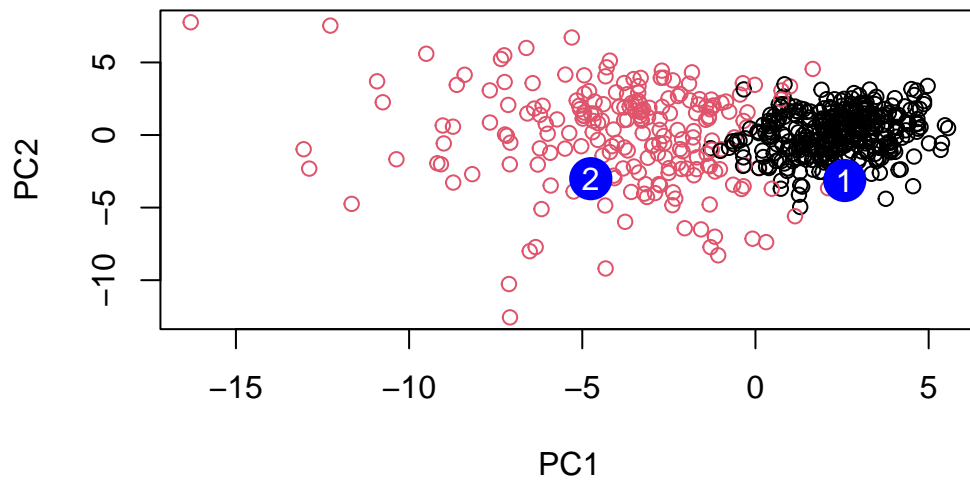
```
              diagnosis
wisc.hclust.clusters  B  M
1      12 165
2       2   5
3     343  40
4       0   2
```

## Prediction

```
#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			

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



skipping question 15, optional

**Q16. Which of these new patients should we prioritize for follow up based on your results?**