

class08

Amit Subramanian

1. Exploratory data analysis

Let's save and load in our csv file.

```
fna.data <- "WisconsinCancer.csv"
wisc.df <- read.csv("WisconsinCancer.csv")

# Creates new variable without the first column
wisc.data <- wisc.df[,-1]

# Extracts out the diagnosis column
diagnosis <- wisc.df$diagnosis
```

Q1. How many observations are in this dataset?

```
nrow(wisc.df)
```

```
[1] 569
```

There are 569 observations in this dataset.

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

```
# Gives us information from the diagnosis column
table(diagnosis)
```

```
diagnosis
  B    M
357 212
```

There are 212 observations with a malignant diagnosis.

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

```
grep("_mean", colnames(wisc.df))
```

```
[1] 3 4 5 6 7 8 9 10 11 12
```

There are 10 variables/features in the dataset which are suffixed with `_mean`.

2. Principal Component Analysis

Let's perform principal component analysis (PCA) on `wisc.data`.

```
# Checking column means and standard deviation
x <- wisc.data[,-1]
colMeans(wisc.data[,-1])
```

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(x,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

```
# Applies PCA
wisc.pr <- prcomp(x, scale=T)
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

```
y <- summary(wisc.pr)
y
```

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?

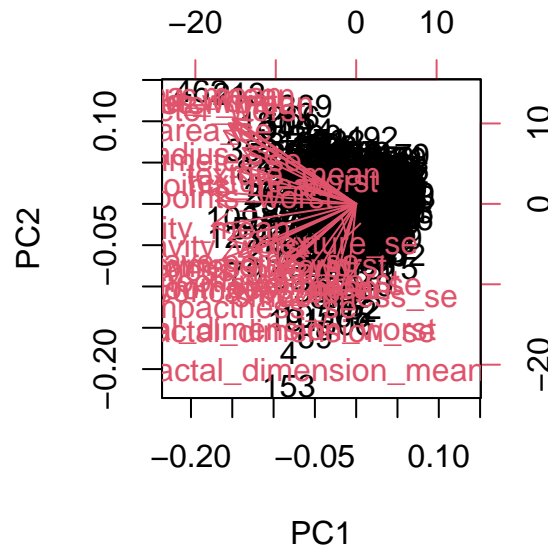
Up to PC3 is required to describe at least 70% of the original variance in the data.

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

Up to PC7 is required to describe at least 90% of the original variance in the data.

Interpreting PCA results

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

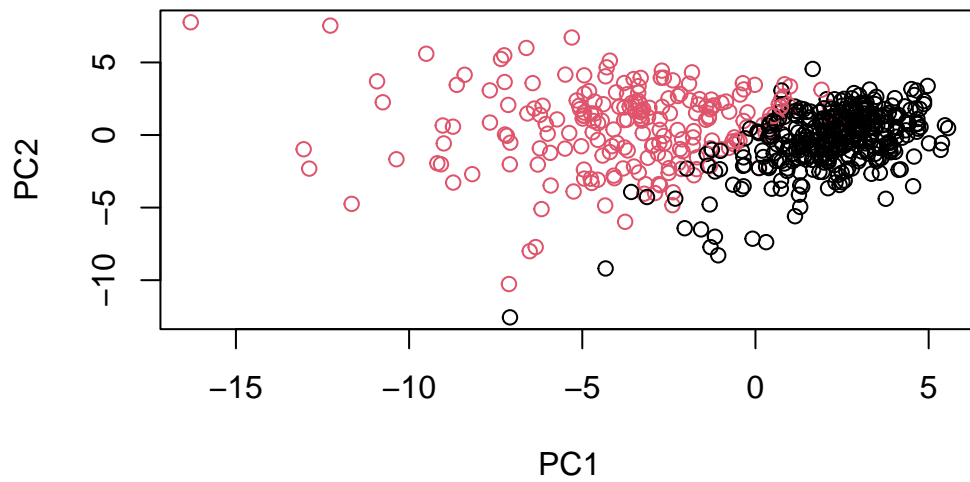


Q7. What stands out to you about this plot? Is it easy or difficult to understand? Why?

There center of the graph is very clustered with different numbers and words. It is very difficult to comprehend any of the data on the plot because there is no digestable information showing clearly on it.

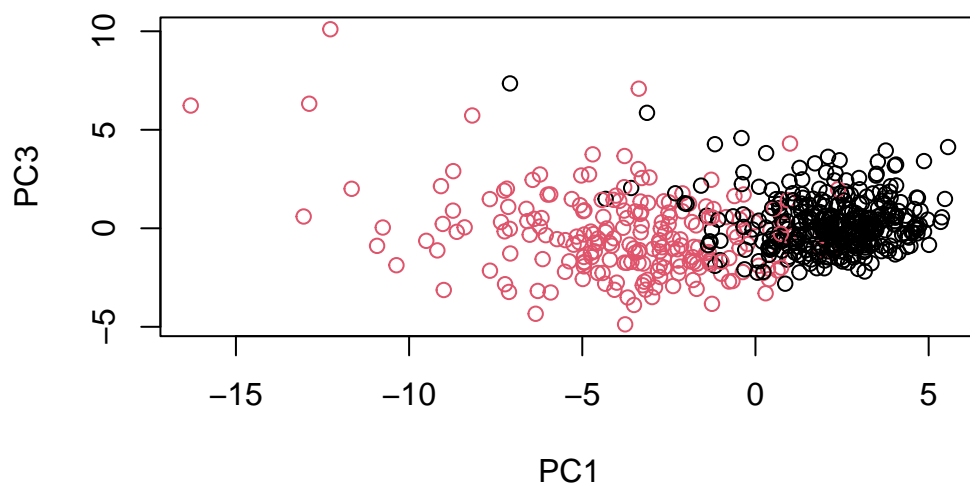
Let's try to improve this scatter plot

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



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

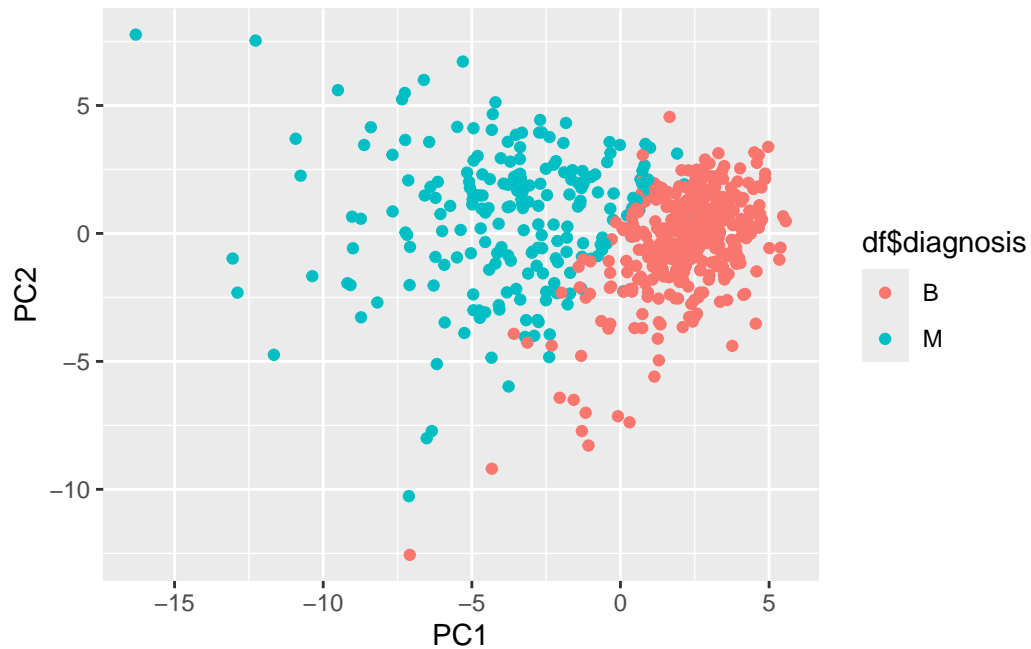
```
# Scatter plot observations by components 1 and 2
plot(wisc.pr$x[,1], wisc.pr$x[,3], col=as.factor(diagnosis),
     xlab="PC1", ylab="PC3")
```



Let's use GGplot2

```
# 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=df$diagnosis) + geom_point()
```

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

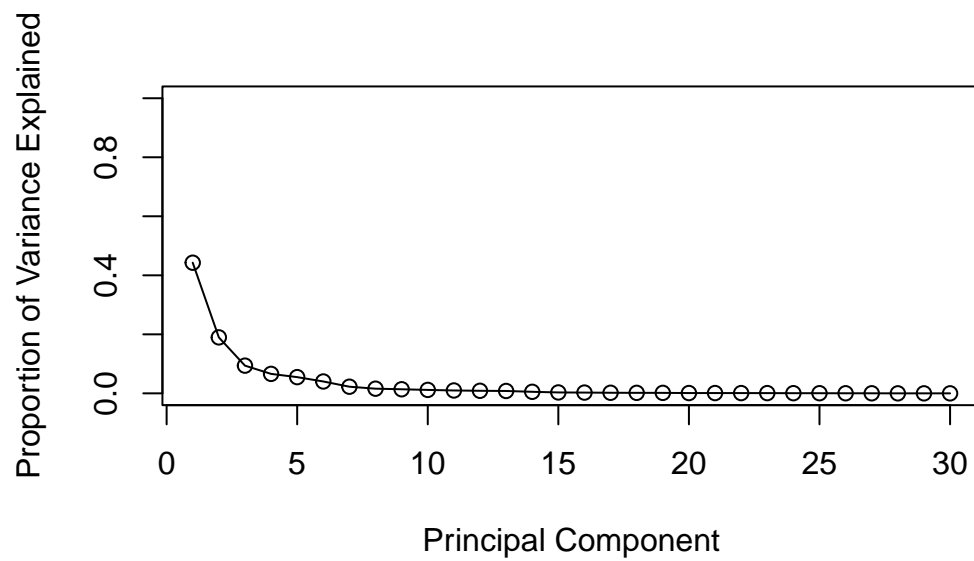


We have to also have it show variance

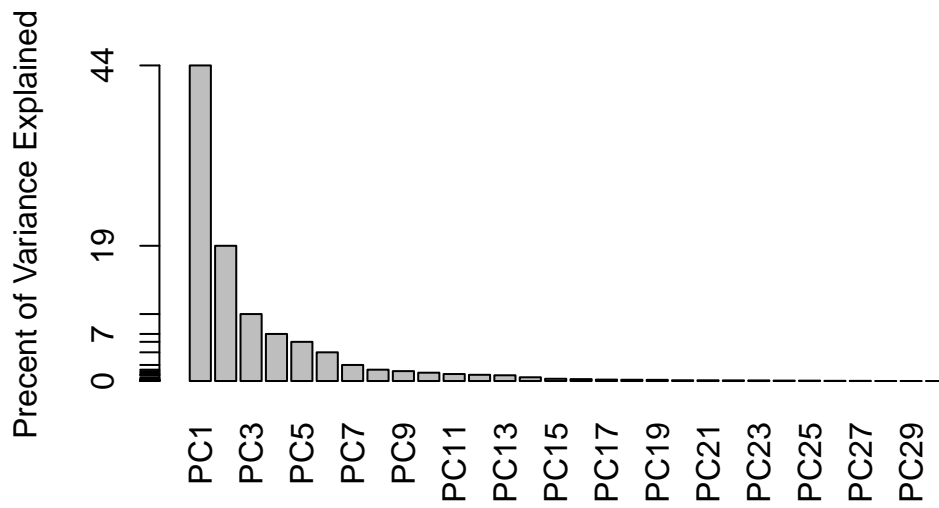
```
# Calculate variance of each component
pr.var <- wisc.pr$sdev^2
head(pr.var)
```

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

```
# 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")
```

```
# Alternative scree plot of the same data, note data driven y-axis
barplot(pve, ylab = "Percent 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`?

```
# Loads in vector
wisc.pr$rotation["concave.points_mean",1]
```

```
[1] -0.2608538
```

The component of loading vector PC1 for the feature `concave.points_mean` is: `'-0.26085376'`. When we compare that to the other data from various principal components, we can see that the data from PC1 is not the only negative value. PC28 and PC29 have even higher negative values, these being: `'-8.88'` and `'-4.21'` which would affect the overall mapping of the data even greater.

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

No amount of PCA in this data would explain 80% of the variance of the data, as the maximum percentage of variance is in PCA1 at 44%.

3. Hierarchical clustering

```
# First scaling the data
data.scaled <- scale(x)

# Calculating the distance between all pairs of observations
data.dist <- dist(data.scaled)

# Hierarchical clustering model using complete linkage
wisc.hclust <- hclust(data.dist, method="complete")
wisc.hclust
```

Call:

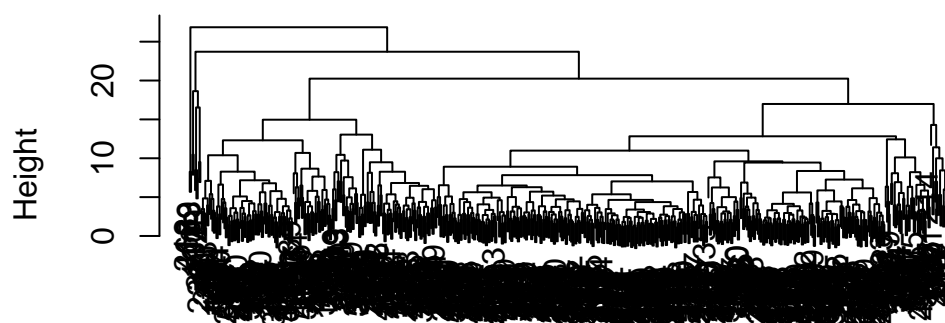
```
hclust(d = data.dist, method = "complete")
```

```
Cluster method   : complete
Distance         : euclidean
Number of objects: 569
```

Let's plot the model.

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

Cluster Dendrogram

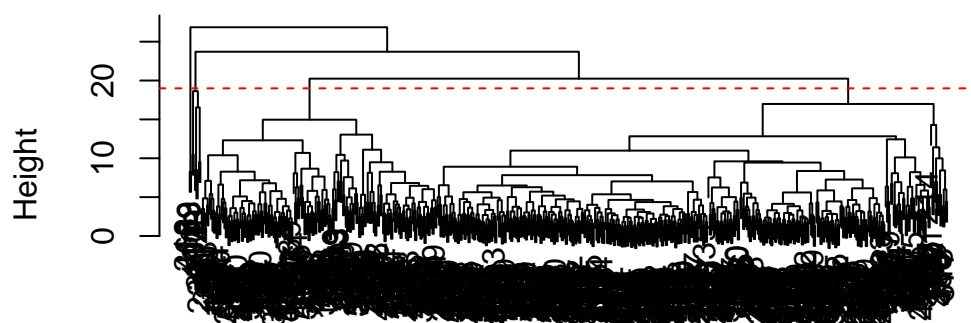


```
data.dist  
hclust (*, "complete")
```

Q11. 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")
```

The height at which the clustering model has 4 clusters would be 19.

Selecting number of clusters

```
# Cutting the tree to only have 4 clusters
wisc.hclust.clusters <- cutree(wisc.hclust, h=19)

# Comparing the cluster membership to 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

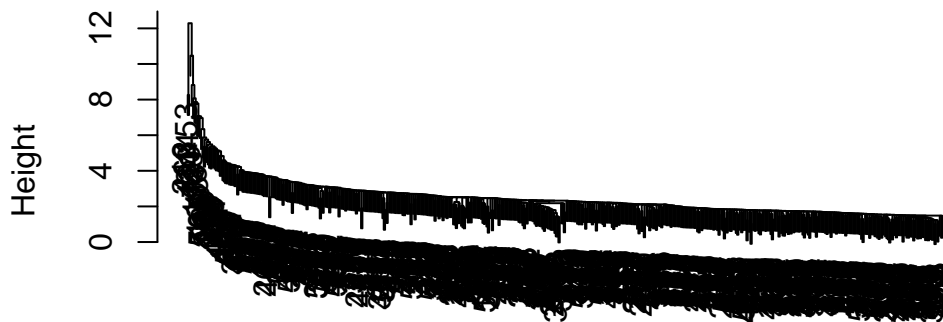
Q12. Can you find a better cluster vs diagnoses match by cutting into a different number of clusters between 2 and 10?

Cutting into 7 clusters would have the best cluster v diagnoses match.

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

```
# Hierarchical clustering model using single linkage
wisc.hclust.single <- hclust(data.dist, method="single")
plot(wisc.hclust.single)
```

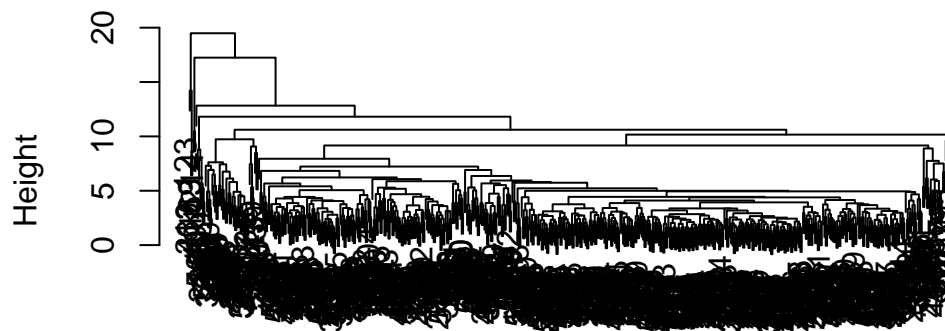
Cluster Dendrogram



data.dist
hclust (*, "single")

```
# Hierarchical clustering model using average linkage
wisc.hclust.average <- hclust(data.dist, method="average")
plot(wisc.hclust.average)
```

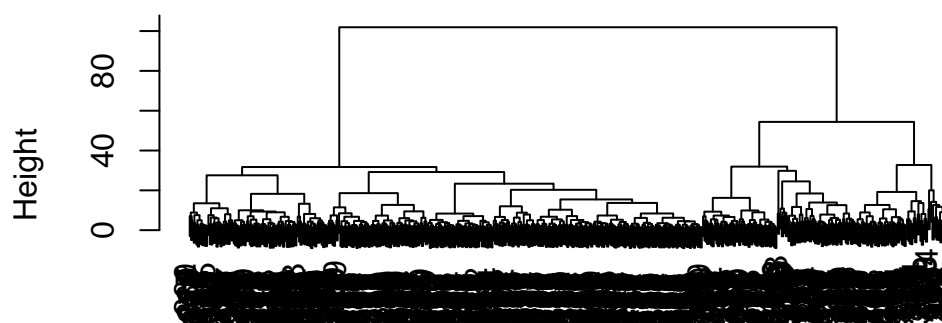
Cluster Dendrogram



data.dist
hclust (*, "average")

```
# Hierarchical clustering model using ward.D2 linkage  
wisc.hclust.ward <- hclust(data.dist, method="ward.D2")  
plot(wisc.hclust.ward)
```

Cluster Dendrogram



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

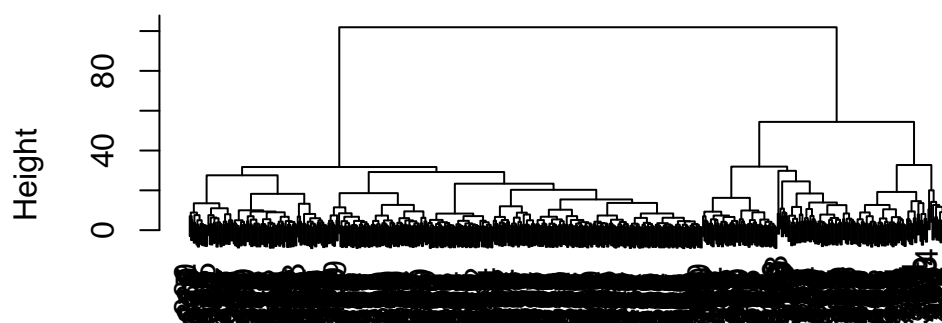
My favorite results for the data.dist dataset was given by the original ‘complete’ method. The graph overall just seems neater, and easier to read and analyze, specifically when it comes to the branching.

5. Combining methods

Now let’s work on clustering our PCA results.

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


Cluster Dendrogram



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

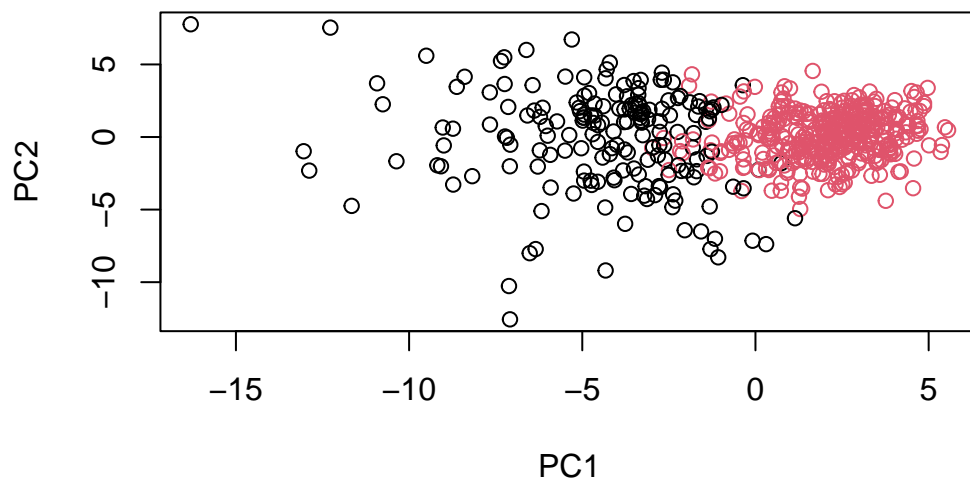
```
# Analyzes our two main clusters  
grps <- cutree(wisc.pr.hclust, k=2)  
table(grps)
```

```
grps  
  1   2  
184 385
```

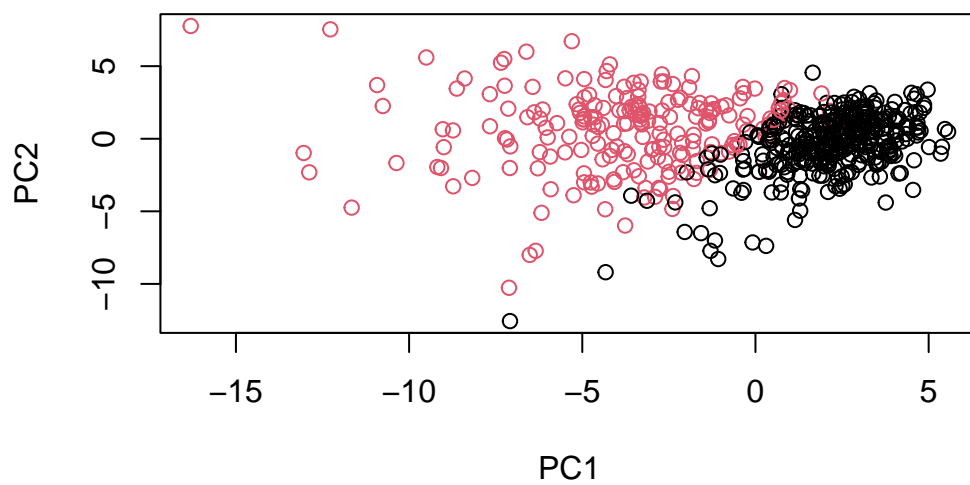
```
table(grps, diagnosis)
```

```
      diagnosis  
grps   B    M  
  1   20 164  
  2  337  48
```

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



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



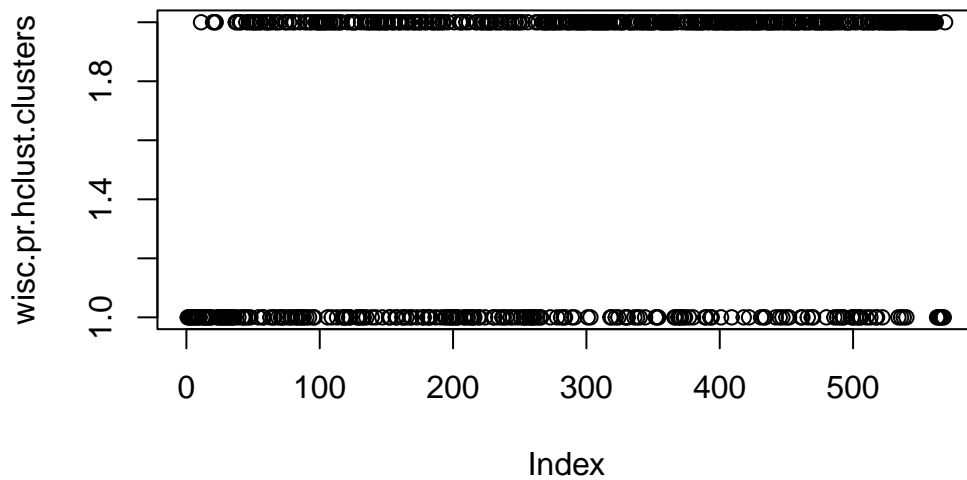
```
# Use the distance along the first 7 PCs for clustering i.e. wisc.pr$x[, 1:7]
wisc.pr.hclust <- hclust(dist(wisc.pr$x[,1:7]), method="ward.D2")
plot(wisc.pr.hclust)
```

Cluster Dendrogram



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

```
# Divides into 2 clusters
wisc.pr.hclust.clusters <- cutree(wisc.pr.hclust, k=2)
plot(wisc.pr.hclust.clusters)
```



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

```
# Compare to actual diagnoses
table(wisc.pr.hclust.clusters, diagnosis)
```

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

Cluster 1 has more instances of malignant cells compared to cluster 2, where there are more benign cells. We can add up the clusters most number of cells and divide them by the total so we can compare this to the actual diagnoses: $(188+329)/569 = 0.909$

Q16. How well does the hierarchical clustering model 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.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	

Cluster 1 has a majority of malignant cells, and a high number of cells in total. Cluster 2 also has mostly malignant cells. Cluster 3 has the highest total number of cells with a majority of them being benign. In cluster 4, there are only 2 total cells, both of them being malignant.

6. Sensitivity/Specificity

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

Hierarchical clustering appears to have the best sensitivity, while it appears that K-means clustering seems to have the best specificity.

7. Prediction

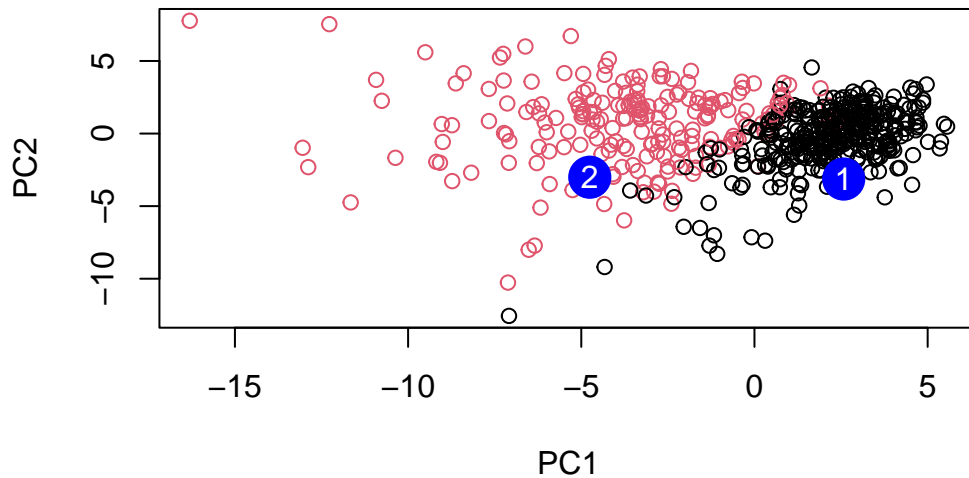
We will use the `predict()` function that will take our PCA model from before and new cancer cell data and project that data onto our PCA space.

```
#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=as.factor(diagnosis))
points(npc[,1], npc[,2], col="blue", pch=16, cex=3)
text(npc[,1], npc[,2], c(1,2), col="white")
```



Q18. Which of these new patients should we prioritize for follow up based on your results?

We should prioritize Patient 2 as the plot shows their principle component contained more malignant cells, indicating a higher risk of threat to their health.