

Class 8: Breast Cancer Mini Project

Rahul Nedunuri (PID A16297840)

Before we dive into the project lets try applying PCA to RNASeq data.

```
url2 <- "https://tinyurl.com/expression-CSV"  
rna.data <- read.csv(url2, row.names=1)  
head(rna.data)
```

| | wt1 | wt2 | wt3 | wt4 | wt5 | ko1 | ko2 | ko3 | ko4 | ko5 |
|-------|------|-----|------|------|-----|-----|-----|-----|-----|-----|
| gene1 | 439 | 458 | 408 | 429 | 420 | 90 | 88 | 86 | 90 | 93 |
| gene2 | 219 | 200 | 204 | 210 | 187 | 427 | 423 | 434 | 433 | 426 |
| gene3 | 1006 | 989 | 1030 | 1017 | 973 | 252 | 237 | 238 | 226 | 210 |
| gene4 | 783 | 792 | 829 | 856 | 760 | 849 | 856 | 835 | 885 | 894 |
| gene5 | 181 | 249 | 204 | 244 | 225 | 277 | 305 | 272 | 270 | 279 |
| gene6 | 460 | 502 | 491 | 491 | 493 | 612 | 594 | 577 | 618 | 638 |

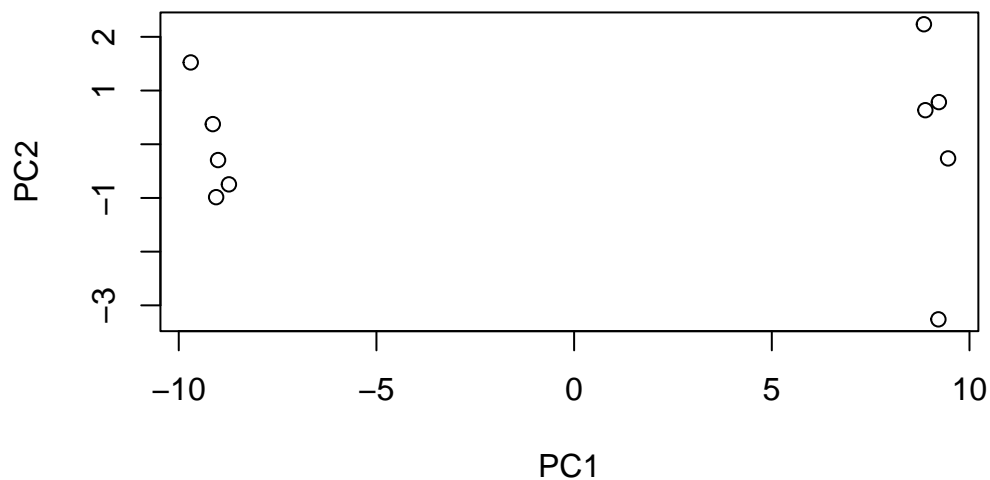
Number of genes:

```
nrow(rna.data)
```

```
[1] 100
```

Run PCA

```
pca <- prcomp(t(rna.data), scale=TRUE)  
  
## Simple un polished plot of pc1 and pc2  
plot(pca$x[,1], pca$x[,2], xlab="PC1", ylab="PC2")
```



```
summary(pca)
```

Importance of components:

| | PC1 | PC2 | PC3 | PC4 | PC5 | PC6 | PC7 |
|------------------------|--------|--------|---------|---------|---------|---------|---------|
| Standard deviation | 9.6237 | 1.5198 | 1.05787 | 1.05203 | 0.88062 | 0.82545 | 0.80111 |
| Proportion of Variance | 0.9262 | 0.0231 | 0.01119 | 0.01107 | 0.00775 | 0.00681 | 0.00642 |
| Cumulative Proportion | 0.9262 | 0.9493 | 0.96045 | 0.97152 | 0.97928 | 0.98609 | 0.99251 |

| | PC8 | PC9 | PC10 |
|------------------------|---------|---------|-----------|
| Standard deviation | 0.62065 | 0.60342 | 3.457e-15 |
| Proportion of Variance | 0.00385 | 0.00364 | 0.000e+00 |
| Cumulative Proportion | 0.99636 | 1.00000 | 1.000e+00 |

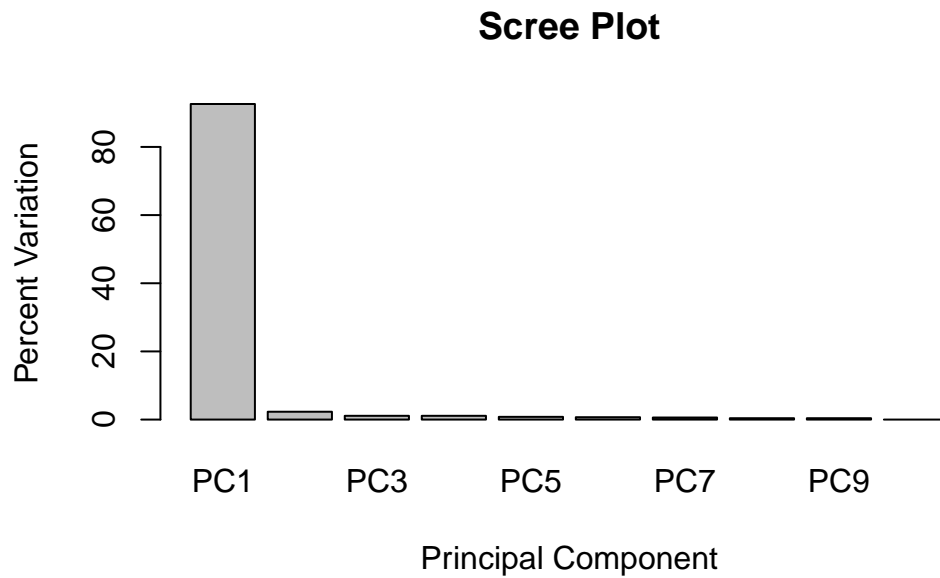
Together, PC1 and PC2 capture almost 95% of all variance in the data. Great!

```
## Variance captured per PC
pca.var <- pca$sdev^2

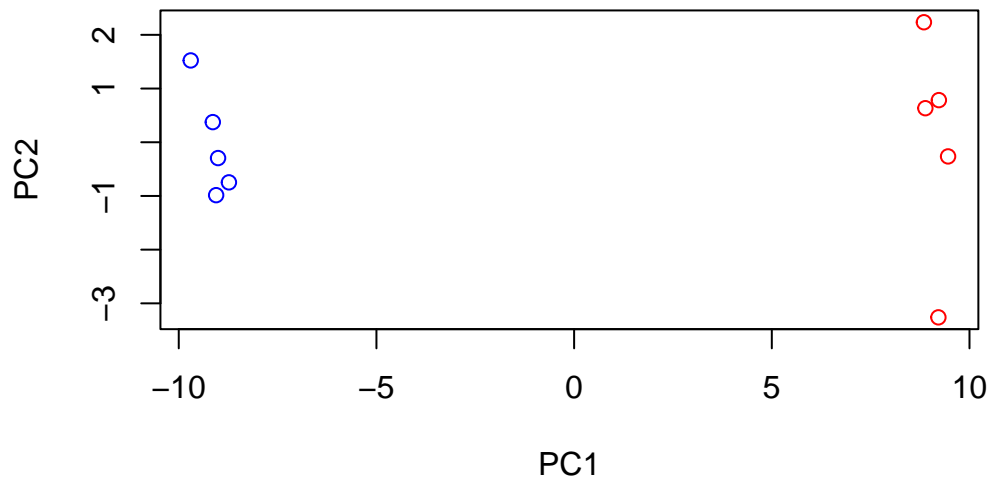
## Percent variance is often more informative to look at
pca.var.per <- round(pca.var/sum(pca.var)*100, 1)
pca.var.per
```

```
[1] 92.6  2.3  1.1  1.1  0.8  0.7  0.6  0.4  0.4  0.0
```

```
barplot(pca.var.per, main="Scree Plot",  
        names.arg = paste0("PC", 1:10),  
        xlab="Principal Component", ylab="Percent Variation")
```



```
# We have 5 wt + 5 ko samples  
  
mycols <- c(rep('blue', 5), rep('red', 5)) # use to color the PCA plot points  
  
plot(pca$x[,1], pca$x[,2], xlab="PC1", ylab="PC2", col=mycols)
```



I could examine which genes contribute most to PC1...

```
head(sort(abs(pca$rotation[,1]), decreasing=T))
```

| gene100 | gene66 | gene45 | gene68 | gene98 | gene60 |
|-----------|-----------|-----------|-----------|-----------|-----------|
| 0.1038708 | 0.1038455 | 0.1038402 | 0.1038395 | 0.1038372 | 0.1038055 |

Analysis of Breast Cancer FNA Data

FNA (fine needle aspiration): biopsy method

Exploratory Data Analysis

Gathering the data

```
# 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)
```

```
head(wisc.df)
```

| | diagnosis | radius_mean | texture_mean | perimeter_mean | area_mean |
|----------|-----------|-------------|--------------|----------------|-----------|
| 842302 | M | 17.99 | 10.38 | 122.80 | 1001.0 |
| 842517 | M | 20.57 | 17.77 | 132.90 | 1326.0 |
| 84300903 | M | 19.69 | 21.25 | 130.00 | 1203.0 |
| 84348301 | M | 11.42 | 20.38 | 77.58 | 386.1 |
| 84358402 | M | 20.29 | 14.34 | 135.10 | 1297.0 |
| 843786 | M | 12.45 | 15.70 | 82.57 | 477.1 |

| | 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 |
| 84348301 | 0.14250 | 0.28390 | 0.2414 | 0.10520 |
| 84358402 | 0.10030 | 0.13280 | 0.1980 | 0.10430 |
| 843786 | 0.12780 | 0.17000 | 0.1578 | 0.08089 |

| | symmetry_mean | fractal_dimension_mean | radius_se | texture_se | perimeter_se |
|----------|---------------|------------------------|-----------|------------|--------------|
| 842302 | 0.2419 | | 0.07871 | 1.0950 | 0.9053 |
| 842517 | 0.1812 | | 0.05667 | 0.5435 | 0.7339 |
| 84300903 | 0.2069 | | 0.05999 | 0.7456 | 0.7869 |
| 84348301 | 0.2597 | | 0.09744 | 0.4956 | 1.1560 |
| 84358402 | 0.1809 | | 0.05883 | 0.7572 | 0.7813 |
| 843786 | 0.2087 | | 0.07613 | 0.3345 | 0.8902 |

| | 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 |
| 84348301 | 27.23 | 0.009110 | 0.07458 | 0.05661 | 0.01867 |
| 84358402 | 94.44 | 0.011490 | 0.02461 | 0.05688 | 0.01885 |
| 843786 | 27.19 | 0.007510 | 0.03345 | 0.03672 | 0.01137 |

| | 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 |
| 84348301 | 0.05963 | 0.009208 | 14.91 | 26.50 |
| 84358402 | 0.01756 | 0.005115 | 22.54 | 16.67 |
| 843786 | 0.02165 | 0.005082 | 15.47 | 23.75 |

| | perimeter_worst | area_worst | smoothness_worst | compactness_worst |
|--------|-----------------|------------|------------------|-------------------|
| 842302 | 184.60 | 2019.0 | 0.1622 | 0.6656 |

| | | | | |
|----------|-------------------------|----------------------|----------------|--------|
| 842517 | 158.80 | 1956.0 | 0.1238 | 0.1866 |
| 84300903 | 152.50 | 1709.0 | 0.1444 | 0.4245 |
| 84348301 | 98.87 | 567.7 | 0.2098 | 0.8663 |
| 84358402 | 152.20 | 1575.0 | 0.1374 | 0.2050 |
| 843786 | 103.40 | 741.6 | 0.1791 | 0.5249 |
| | 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 | |
| 84348301 | 0.6869 | 0.2575 | 0.6638 | |
| 84358402 | 0.4000 | 0.1625 | 0.2364 | |
| 843786 | 0.5355 | 0.1741 | 0.3985 | |
| | fractal_dimension_worst | | | |
| 842302 | 0.11890 | | | |
| 842517 | 0.08902 | | | |
| 84300903 | 0.08758 | | | |
| 84348301 | 0.17300 | | | |
| 84358402 | 0.07678 | | | |
| 843786 | 0.12440 | | | |

wisc.df\$diagnosis is our expert medical diagnosis

Let's make a dataframe that excludes this diagnosis column

```
wisc.data <- wisc.df[,-1]
head(wisc.data)
```

| | | | | | |
|----------|------------------------|----------------|---------------------|---------------|-----------------|
| | radius_mean | texture_mean | perimeter_mean | area_mean | smoothness_mean |
| 842302 | 17.99 | 10.38 | 122.80 | 1001.0 | 0.11840 |
| 842517 | 20.57 | 17.77 | 132.90 | 1326.0 | 0.08474 |
| 84300903 | 19.69 | 21.25 | 130.00 | 1203.0 | 0.10960 |
| 84348301 | 11.42 | 20.38 | 77.58 | 386.1 | 0.14250 |
| 84358402 | 20.29 | 14.34 | 135.10 | 1297.0 | 0.10030 |
| 843786 | 12.45 | 15.70 | 82.57 | 477.1 | 0.12780 |
| | compactness_mean | concavity_mean | concave.points_mean | symmetry_mean | |
| 842302 | 0.27760 | 0.3001 | 0.14710 | 0.2419 | |
| 842517 | 0.07864 | 0.0869 | 0.07017 | 0.1812 | |
| 84300903 | 0.15990 | 0.1974 | 0.12790 | 0.2069 | |
| 84348301 | 0.28390 | 0.2414 | 0.10520 | 0.2597 | |
| 84358402 | 0.13280 | 0.1980 | 0.10430 | 0.1809 | |
| 843786 | 0.17000 | 0.1578 | 0.08089 | 0.2087 | |
| | fractal_dimension_mean | radius_se | texture_se | perimeter_se | area_se |

| | | | | | |
|---|----------|----------|---------|---------|--------|
| 842302 | 0.07871 | 1.0950 | 0.9053 | 8.589 | 153.40 |
| 842517 | 0.05667 | 0.5435 | 0.7339 | 3.398 | 74.08 |
| 84300903 | 0.05999 | 0.7456 | 0.7869 | 4.585 | 94.03 |
| 84348301 | 0.09744 | 0.4956 | 1.1560 | 3.445 | 27.23 |
| 84358402 | 0.05883 | 0.7572 | 0.7813 | 5.438 | 94.44 |
| 843786 | 0.07613 | 0.3345 | 0.8902 | 2.217 | 27.19 |
| smoothness_se compactness_se concavity_se concave.points_se | | | | | |
| 842302 | 0.006399 | 0.04904 | 0.05373 | 0.01587 | |
| 842517 | 0.005225 | 0.01308 | 0.01860 | 0.01340 | |
| 84300903 | 0.006150 | 0.04006 | 0.03832 | 0.02058 | |
| 84348301 | 0.009110 | 0.07458 | 0.05661 | 0.01867 | |
| 84358402 | 0.011490 | 0.02461 | 0.05688 | 0.01885 | |
| 843786 | 0.007510 | 0.03345 | 0.03672 | 0.01137 | |
| 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 | |
| 84348301 | 0.05963 | 0.009208 | 14.91 | 26.50 | |
| 84358402 | 0.01756 | 0.005115 | 22.54 | 16.67 | |
| 843786 | 0.02165 | 0.005082 | 15.47 | 23.75 | |
| perimeter_worst area_worst smoothness_worst compactness_worst | | | | | |
| 842302 | 184.60 | 2019.0 | 0.1622 | 0.6656 | |
| 842517 | 158.80 | 1956.0 | 0.1238 | 0.1866 | |
| 84300903 | 152.50 | 1709.0 | 0.1444 | 0.4245 | |
| 84348301 | 98.87 | 567.7 | 0.2098 | 0.8663 | |
| 84358402 | 152.20 | 1575.0 | 0.1374 | 0.2050 | |
| 843786 | 103.40 | 741.6 | 0.1791 | 0.5249 | |
| 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 | | |
| 84348301 | 0.6869 | 0.2575 | 0.6638 | | |
| 84358402 | 0.4000 | 0.1625 | 0.2364 | | |
| 843786 | 0.5355 | 0.1741 | 0.3985 | | |
| fractal_dimension_worst | | | | | |
| 842302 | 0.11890 | | | | |
| 842517 | 0.08902 | | | | |
| 84300903 | 0.08758 | | | | |
| 84348301 | 0.17300 | | | | |
| 84358402 | 0.07678 | | | | |
| 843786 | 0.12440 | | | | |

We should also store the diagnoses as a factor

```
diagnosis <- factor(wisc.df[,1])  
head(diagnosis)
```

```
[1] M M M M M M  
Levels: B M
```

Q1. How many observations are in this dataset? 569 observations

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

```
[1] 569
```

569 observations are in this dataset.

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

```
table(diagnosis)
```

```
diagnosis  
  B   M  
357 212
```

212 observations have a malignant diagnosis.

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

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

```
[1] 10
```

```
# use ,value=T, to find the names ending in __mean
```

10 variables in the data are suffixed with `__mean`.

PCA

Looks like we need to scale the data. Here we will use `prcomp()` on the `wisc.data` object.

First we have to decide whether to use the `scale=T` or `scale=F` argument.

We can look at the means and sd of each column. If they are similar then we are all good to go.

Let's look at column means to see whether scaling is needed.

```
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 |

We need to scale the data because the means and sd vary heavily in magnitude.

```
wisc.pr <- prcomp(wisc.data, 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 | | | | | |

Q4. From your results, what proportion of the original variance is captured by the first principal components (PC1)?

44.27% of variance (much lower than earlier). This makes sense because data from patient samples expectedly have a lot of noise.

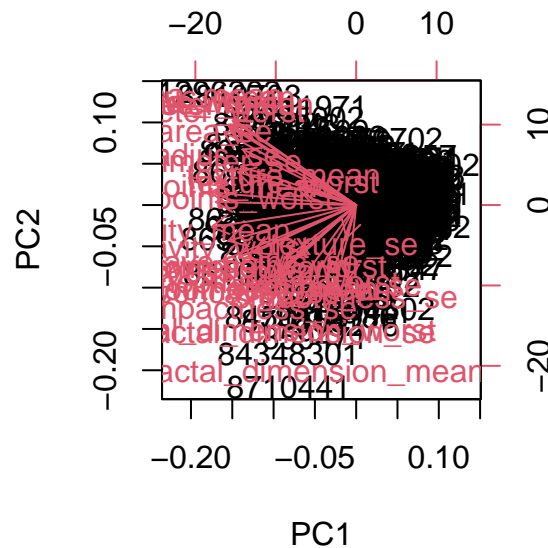
Q5. How many principal components (PCs) are required to describe at least 70% of the original variance in the data?

3 PCs. PC1-3 account for 72.6% of variance in the data.

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

7 PCs. PC1-7 account for 91% of variance in the data.

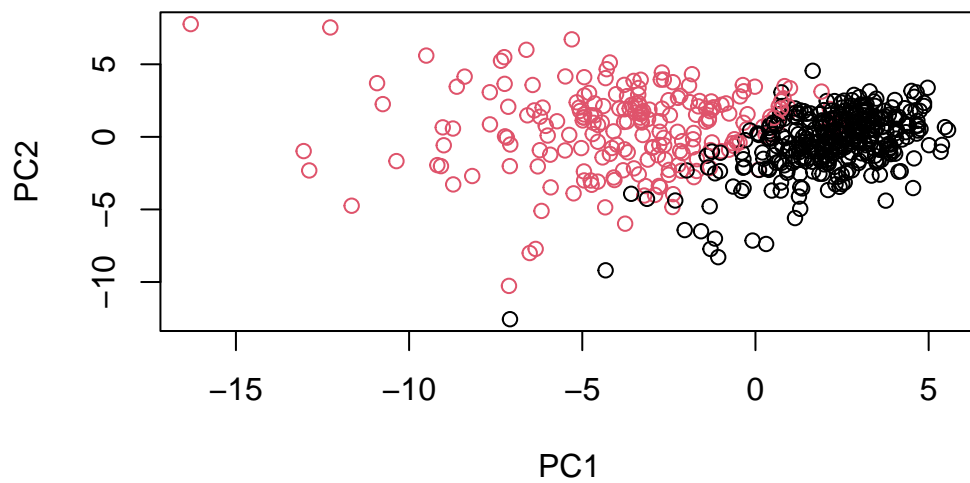
```
biplot(wisc.pr)
```



What a mess!

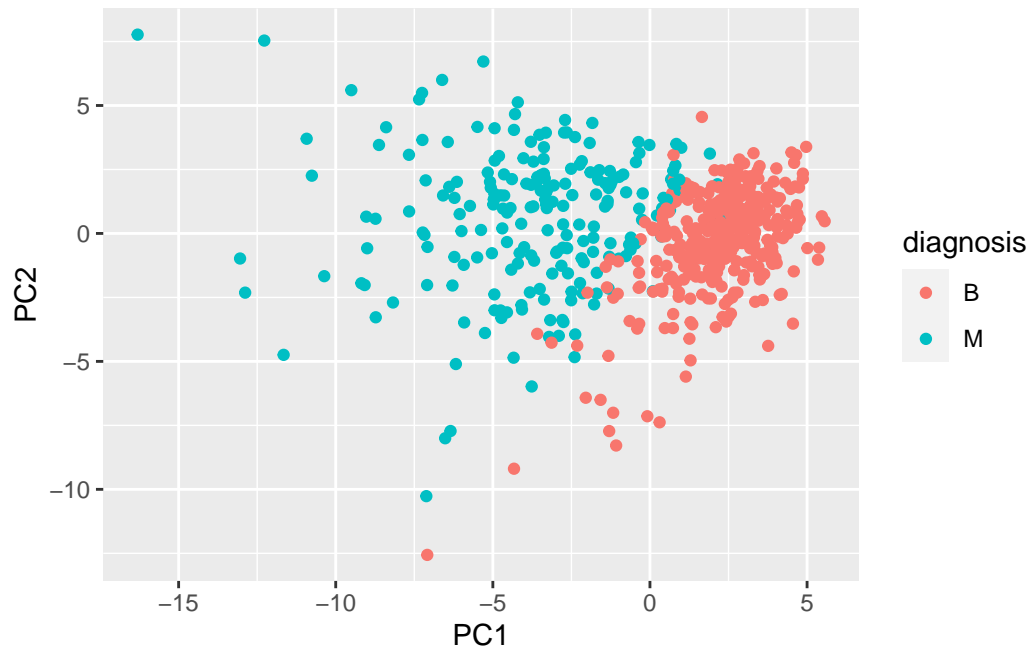
Let's try another plot of PC1 vs PC2.

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



This looks much more readable.

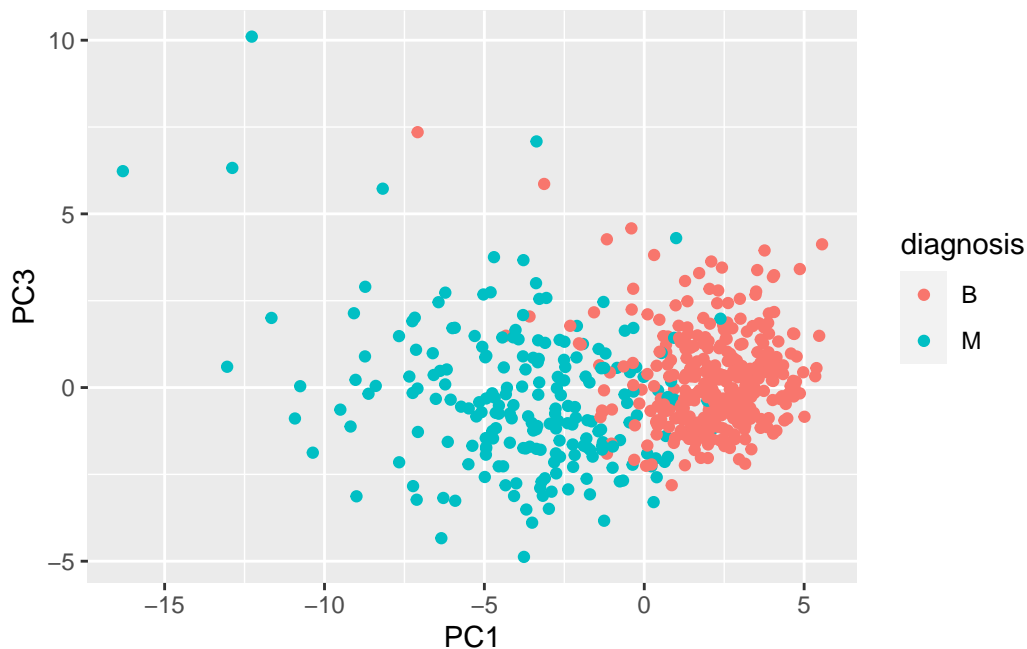
```
library(ggplot2)  
  
pc <- as.data.frame(wisc.pr$x)  
  
ggplot(pc) +  
  aes(PC1, PC2, col=diagnosis) +  
  geom_point()
```



This is even better.

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

```
ggplot(pc) +  
  aes(PC1, PC3, col=diagnosis) +  
  geom_point()
```



These plots have as many datapoints as initial observations. This is because each observation is given a score based on the PCs.

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`?

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

```
[1] -0.2608538
```

-0.2608538 is the component of the loading vector for the feature `concave.points_mean`.

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

5 PCs explain 80% of the variance.

```
tbl <- summary(wisc.pr)
tbl$importance[3,]
```

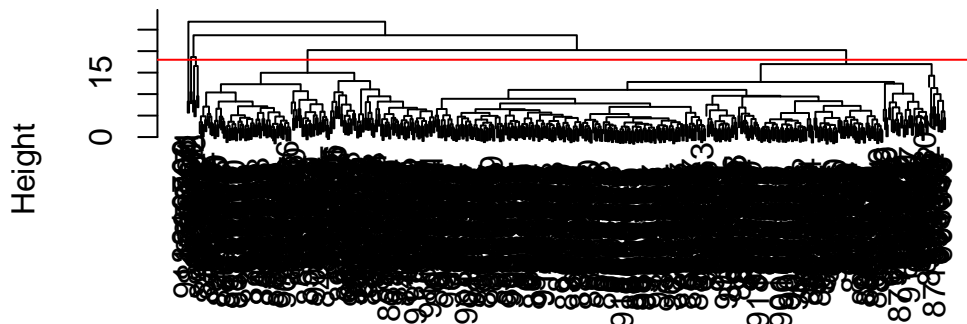
| PC1 | PC2 | PC3 | PC4 | PC5 | PC6 | PC7 | PC8 | PC9 | PC10 |
|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| 0.44272 | 0.63243 | 0.72636 | 0.79239 | 0.84734 | 0.88759 | 0.91010 | 0.92598 | 0.93988 | 0.95157 |
| PC11 | PC12 | PC13 | PC14 | PC15 | PC16 | PC17 | PC18 | PC19 | PC20 |
| 0.96137 | 0.97007 | 0.97812 | 0.98335 | 0.98649 | 0.98915 | 0.99113 | 0.99288 | 0.99453 | 0.99557 |
| PC21 | PC22 | PC23 | PC24 | PC25 | PC26 | PC27 | PC28 | PC29 | PC30 |
| 0.99657 | 0.99749 | 0.99830 | 0.99890 | 0.99942 | 0.99969 | 0.99992 | 0.99997 | 1.00000 | 1.00000 |

Hierarchical Clustering

The main function for hierarchical clustering is called `hclust()` which takes a distance matrix as input.

```
d <- dist(scale(wisc.data))
wisc.hclust <- hclust(d)
plot(wisc.hclust)
abline(h=18, col="red")
```

Cluster Dendrogram



d
hclust (*, "complete")

```
grps <- cutree(wisc.hclust, h=18)
```

Let's see how our cluster grps correspond to malignancy (M or B diagnosis groups).

```
table(grps)
```

```
grps
```

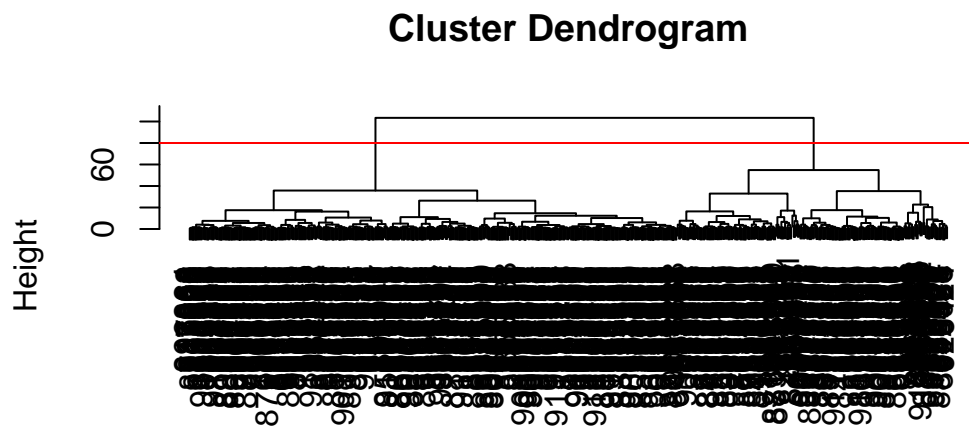
```
  1  2  3  4  5  
177  5 383  2  2
```

5. Combining methods

Here we will perform clustering on PCA results rather than the original data.

In other words, we will cluster using `wisc.pr$x` - our new better variables of PCs. We can choose as many or as few PCs to use as we like.

```
d.pc <- dist(wisc.pr$x[, 1:3])  
  
wisc.pr.hclust <- hclust( d.pc, method="ward.D2" )  
plot(wisc.pr.hclust)  
abline(h=80, col='red')
```



```
d.pc  
hclust (*, "ward.D2")
```



```
grps <- cutree(wisc.pr.hclust, h=80)
table(grps)
```

```
grps
  1  2
203 366
```

We can use the `table()` function to make a cross-table.

```
table(diagnosis)
```

```
diagnosis
  B  M
357 212
```

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

```
      diagnosis
grps  B  M
  1  24 179
  2 333  33
```

From group 1, 179 were malignant and only 24 were benign. From group 2, 333 were benign and 33 were malignant. Cluster 1 mostly captures malignant (M) tumors while cluster 2 captures mainly benign masses/healthy patients (B).

7. Predict

```
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
```

```

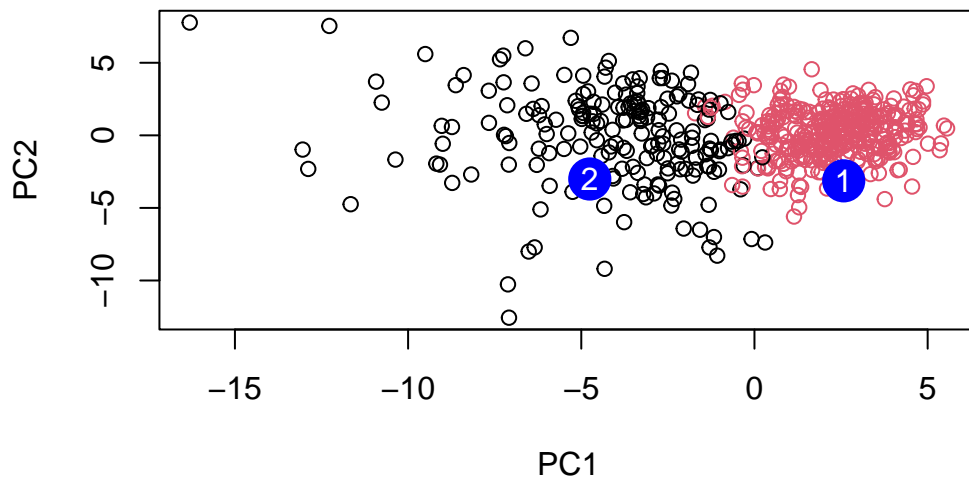
[1,] -0.2307350 0.1029569 -0.9272861 0.3411457 0.375921 0.1610764 1.187882
[2,] -0.3307423 0.5281896 -0.4855301 0.7173233 -1.185917 0.5893856 0.303029
      PC15      PC16      PC17      PC18      PC19      PC20
[1,] 0.3216974 -0.1743616 -0.07875393 -0.11207028 -0.08802955 -0.2495216
[2,] 0.1299153 0.1448061 -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=grps)
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 group 2, which is likely to contain patients with malignancies that were missed in the initial diagnosis.