class07

Jennifer

Principal Component Analysis

PCA of UK food data

Read data from website and try a few visualizations.

> Q1. How many rows and columns are in your new data frame named x? What R functions could you use to answer this questions?

```
url <- "https://tinyurl.com/UK-foods"
x <- read.csv(url, row.names = 1)
```

```
dim(x)
```

```
[1] 17  4
```

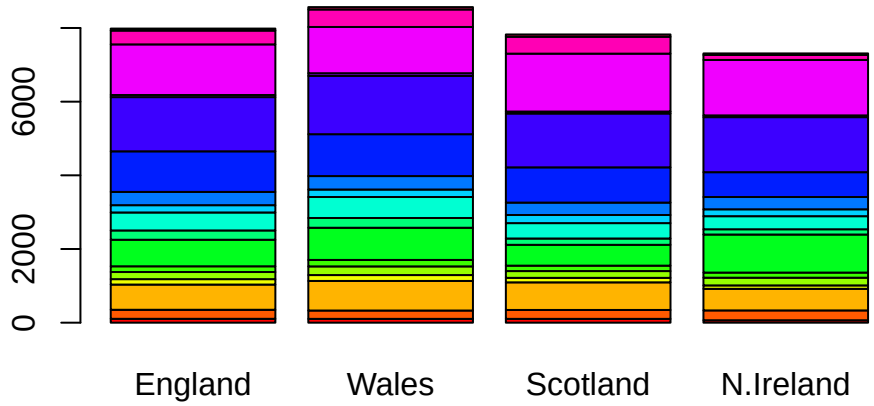
There are 17 rows and 4 columns.

> Q2. Which approach to solving the 'row-names problem' mentioned above do you prefer and why? Is one approach more robust than another under certain circumstances?

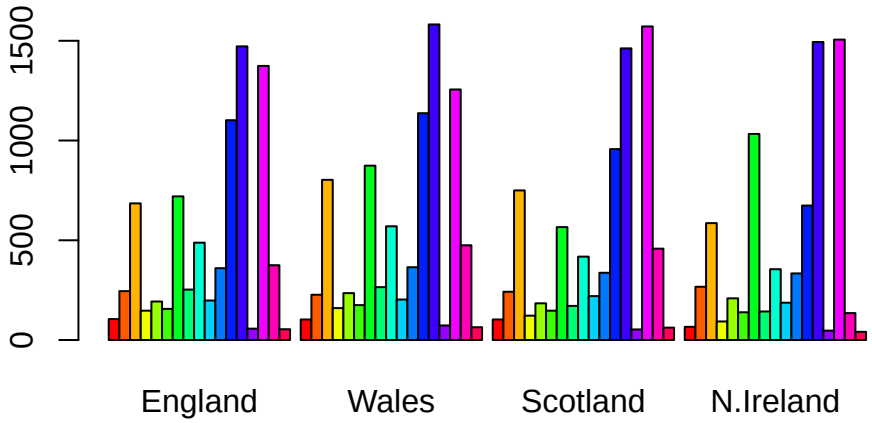
Using the argument 'row.names=1' provides a simpler and quicker way to adjust the dimensions of a data set.

> Q3. Changing what optional argument in the above barplot() function results in the following plot?

```
cols<-rainbow(nrow(x))
barplot(as.matrix(x), col = cols)
```



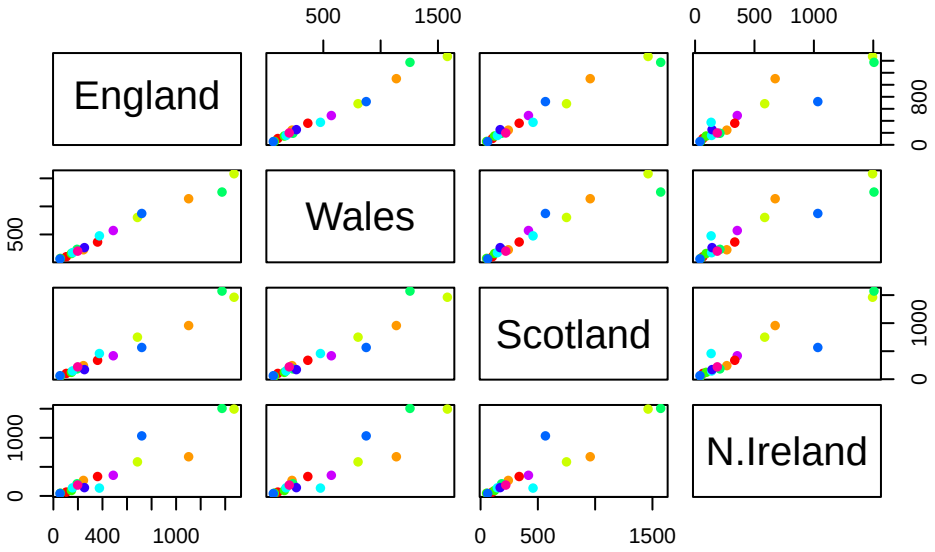

```
barplot(as.matrix(x), col = cols, beside = TRUE)
```



In the 'barplot()' function, adding the argument 'beside = TRUE' will result in a grouped bar plot and taking the argument out will result in a stacked bar plot.

> Q5. Generating all pairwise plots may help somewhat. Can you make sense of the following code and resulting figure? What does it mean if a given point lies on the diagonal for a given plot?

```
pairs(x, col=rainbow(10), pch=16)
```

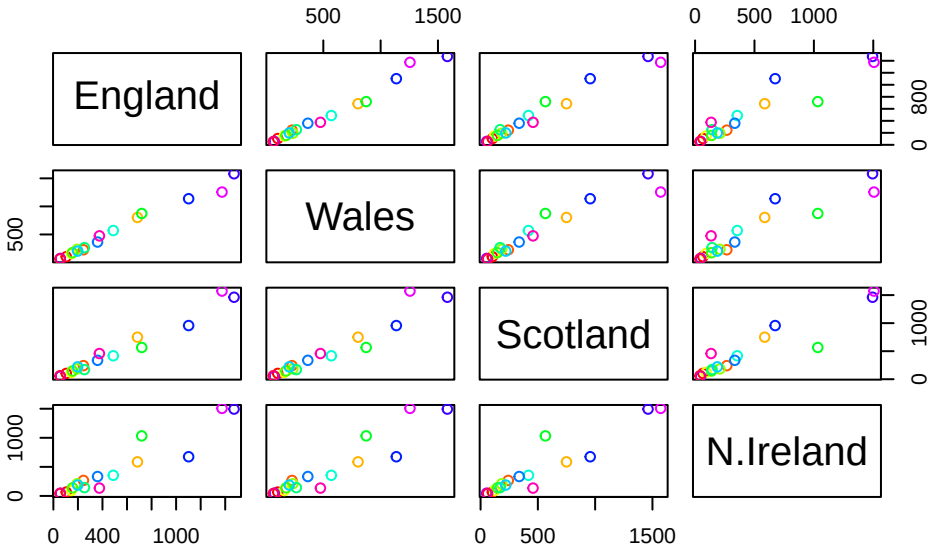


If a point lies on the diagonal for a given plot it means that people in the different countries have the same amount of consumption of the food being measured.

> Q6. What is the main differences between N. Ireland and the other countries of the UK in terms of this data-set?

The main differences between N. Ireland and the other countries of the UK are that there are some points on the plot that are higher up relative to the diagonal. This shows that people in N. Ireland consume more of certain foods.

```
pairs(x, col = cols)
```



PCA to the rescue! The main base R PCA function is called 'prcomp()' and we will need to give it the transpose of our input data!

```
pca<-prcomp(t(x))
```

```
# Use the prcomp() PCA function
pca <- prcomp( t(x) )
summary(pca)
```

```
Importance of components:
                            PC1      PC2      PC3        PC4
Standard deviation     324.1502 212.7478 73.87622 4.189e-14
Proportion of Variance   0.6744   0.2905  0.03503 0.000e+00
Cumulative Proportion    0.6744   0.9650  1.00000 1.000e+00
```

```
attributes(pca)
```

```
$names
[1] "sdev"     "rotation" "center"   "scale"    "x"
```

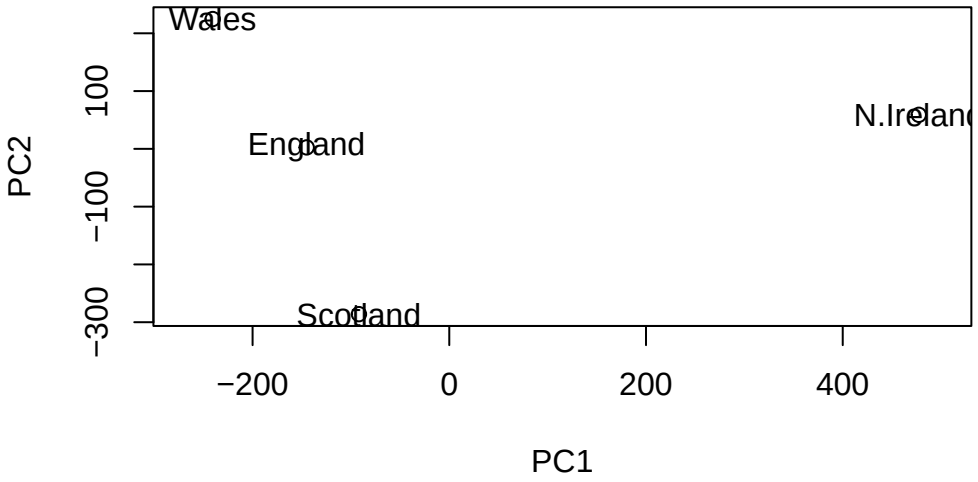
```
$class
[1] "prcomp"
```

To make our new PCA plot we access 'pca$x'

Q7. Complete the code below to generate a plot of PC1 vs PC2. The second line adds text labels over the data points.

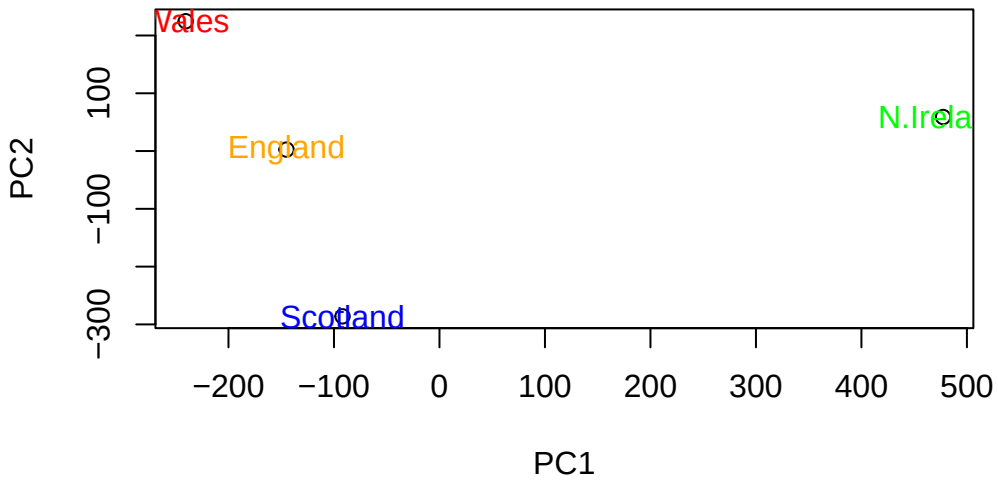
Plot PC1 vs PC2

```
plot(pca$x[,1], pca$x[,2], xlab="PC1", ylab="PC2", xlim=c(-270,500))
text(pca$x[,1], pca$x[,2], colnames(x))
```



Q8. Customize your plot so that the colors of the country names match the colors in our UK and Ireland map and table at start of this document.

```
country_cols <- c("orange", "red", "blue", "green")
plot(pca$x[,1], pca$x[,2], xlab="PC1", ylab="PC2")
text(pca$x[,1], pca$x[,2], colnames(x), col = country_cols)
```



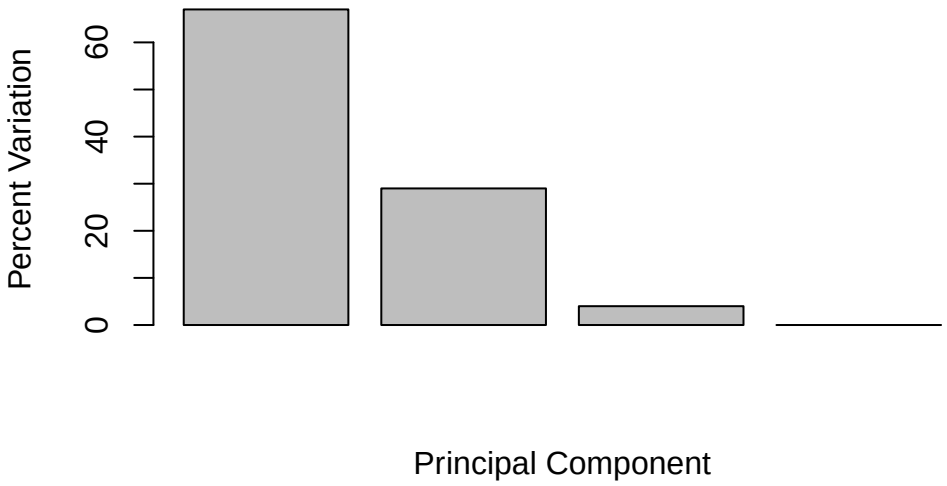
Calculate how much variation in the original data each PC accounts for.

```
v <- round( pca$sdev^2/sum(pca$sdev^2) * 100 )
v
```

```
[1] 67 29  4  0
```

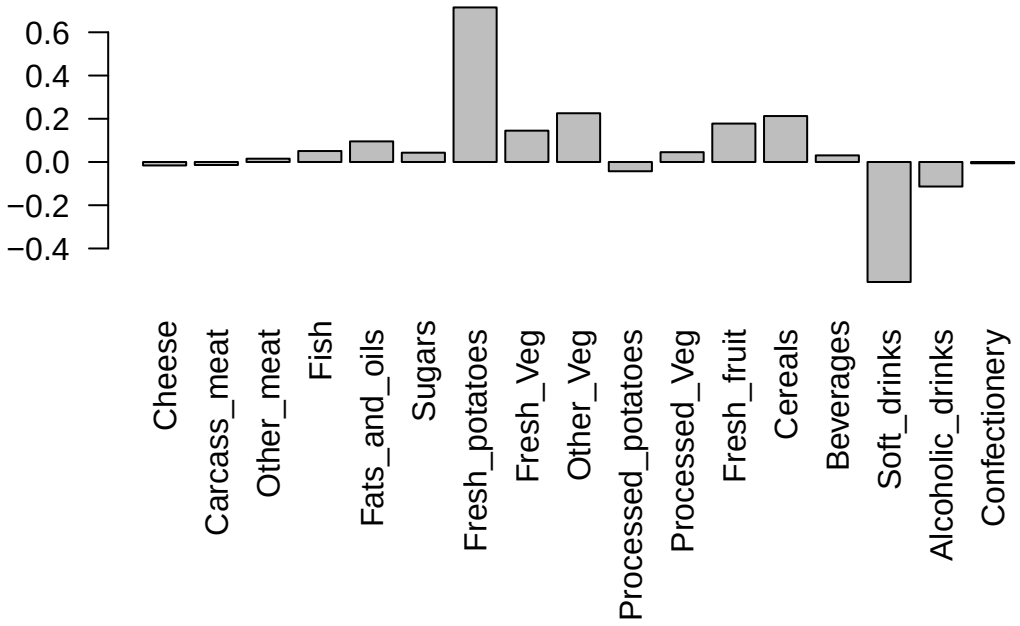
We can plot the variances.

```
barplot(v, xlab="Principal Component", ylab="Percent Variation")
```



Q9. Generate a similar 'loadings plot' for PC2. What two food groups feature prominantely and what does PC2 maninly tell us about?

```
par(mar=c(10, 3, 0.35, 0))
barplot( pca$rotation[,2], las=2 )
```



Fresh potaotes and soft drinks are featured prominately. PC2 mainly tells us that foods such as fresh potatoes push Ireland to the right positive side while soft drinks push contries to the left.

PCA of RNA-Seq data

Read data from website

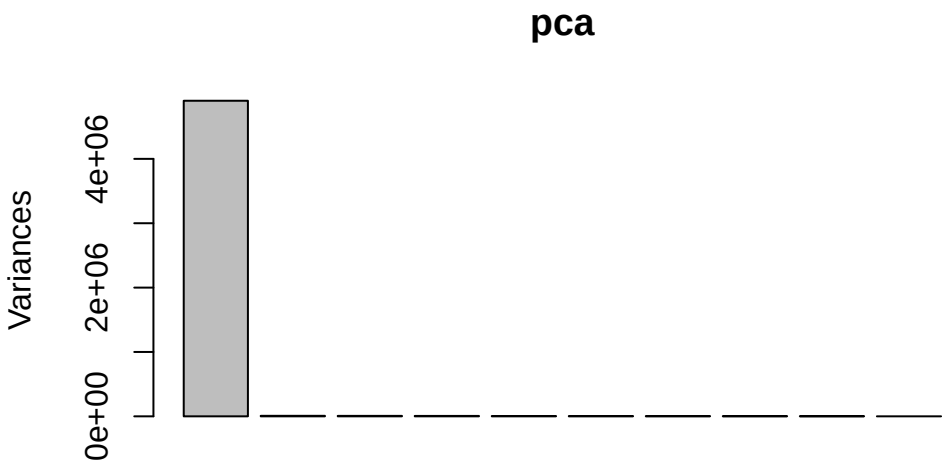
```
url2 <- "https://tinyurl.com/expression-CSV"
rna.data <- read.csv(url2, row.names=1)
head(rna.data)
```

```
       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
```

Q10. How many genes and samples are in this data set?

There are 100 genes and 10 samples.

```
pca <- prcomp( t(rna.data) )
plot(pca)
```



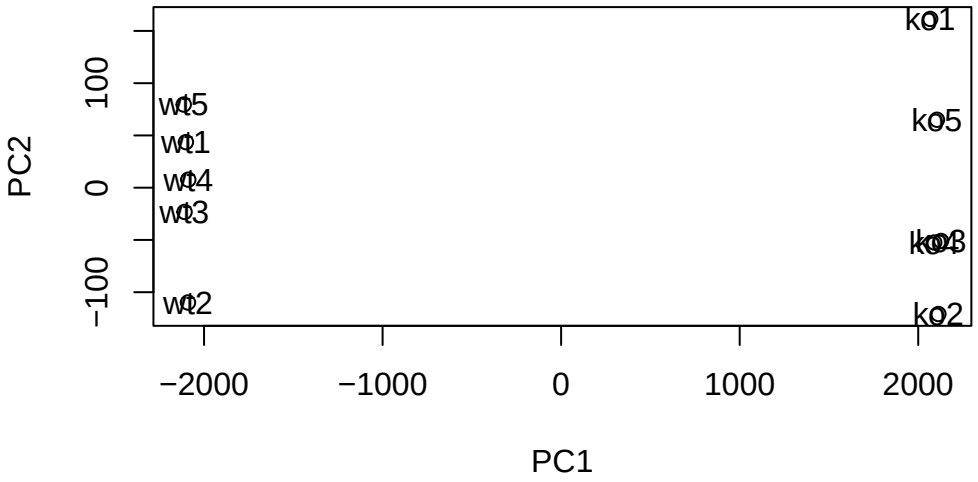
Let's generate a summary to see how much variation in the original data each PC accounts for.

```
pca <- prcomp(t(rna.data))
summary(pca)
```

```
Importance of components:
                            PC1      PC2      PC3      PC4      PC5      PC6
Standard deviation     2214.2633 88.9209 84.33908 77.74094 69.66341 67.78516
Proportion of Variance    0.9917  0.0016  0.00144  0.00122  0.00098  0.00093
Cumulative Proportion     0.9917  0.9933  0.99471  0.99593  0.99691  0.99784
                           PC7      PC8      PC9      PC10
Standard deviation     65.29428 59.90981 53.20803 3.142e-13
Proportion of Variance  0.00086  0.00073  0.00057 0.000e+00
Cumulative Proportion   0.99870  0.99943  1.00000 1.000e+00
```

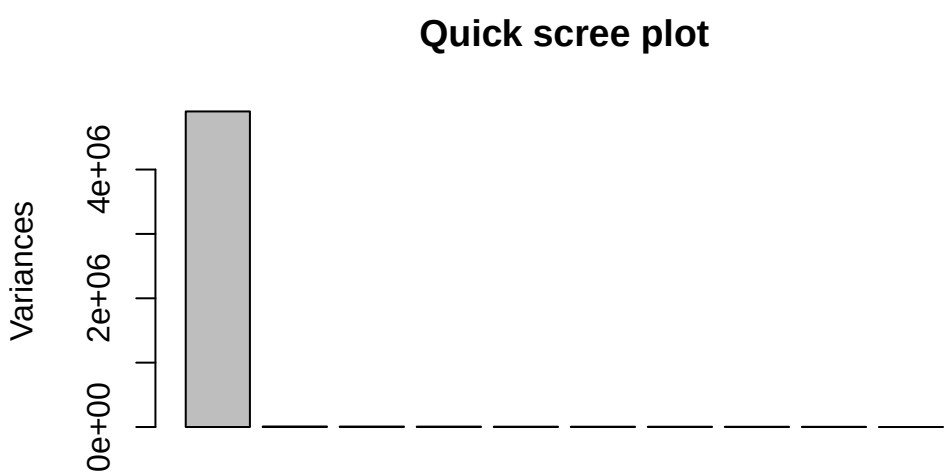
Let's do our PCA plot of this RNA-Seq data.

```
plot(pca$x[,1], pca$x[,2], xlab="PC1", ylab="PC2")
text(pca$x[,1], pca$x[,2], colnames(rna.data))
```



We can generate a quick barplot summary of this Proportion of Variance.

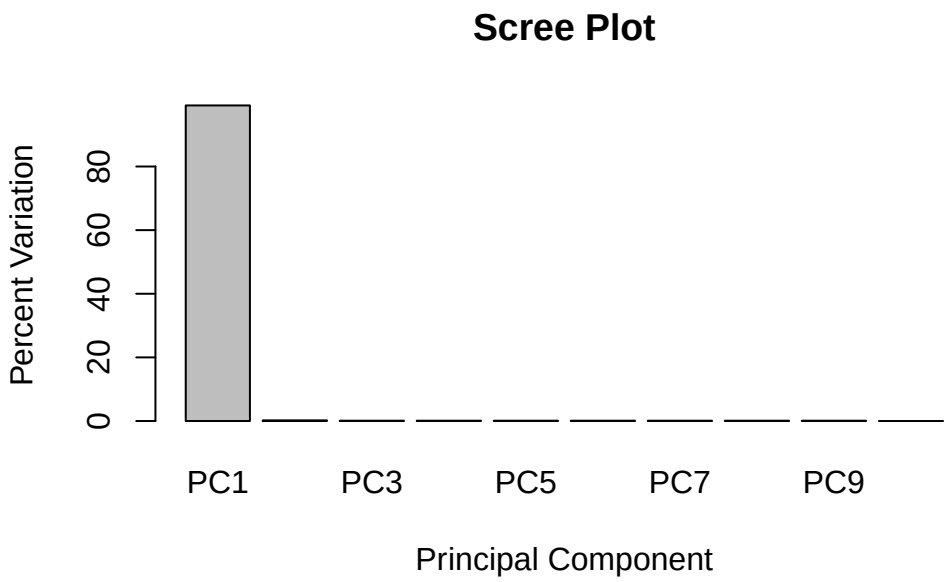
```
plot(pca, main="Quick scree plot")
```



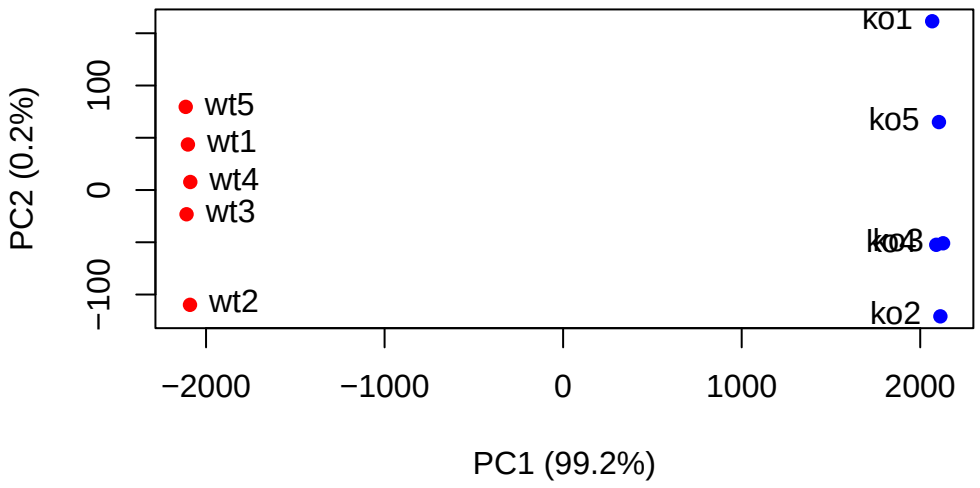

```r
pca.var <- pca$sdev^2
pca.var.per <- round(pca.var/sum(pca.var)*100, 1)
pca.var.per
```

```
[1] 99.2  0.2  0.1  0.1  0.1  0.1  0.1  0.1  0.1  0.0
```

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

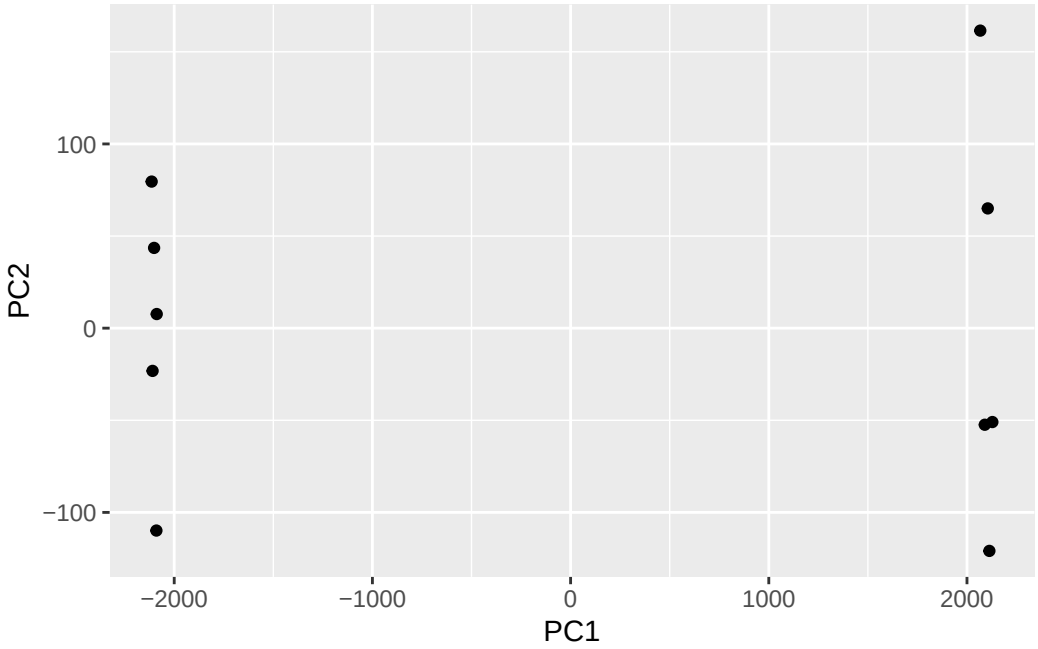



```
colvec <- colnames(rna.data)
colvec[grep("wt", colvec)] <- "red"
colvec[grep("ko", colvec)] <- "blue"

plot(pca$x[,1], pca$x[,2], col=colvec, pch=16,
     xlab=paste0("PC1 (", pca.var.per[1], "%)"),
     ylab=paste0("PC2 (", pca.var.per[2], "%)"))

text(pca$x[,1], pca$x[,2], labels = colnames(rna.data), pos=c(rep(4,5), rep(2,5)))
```

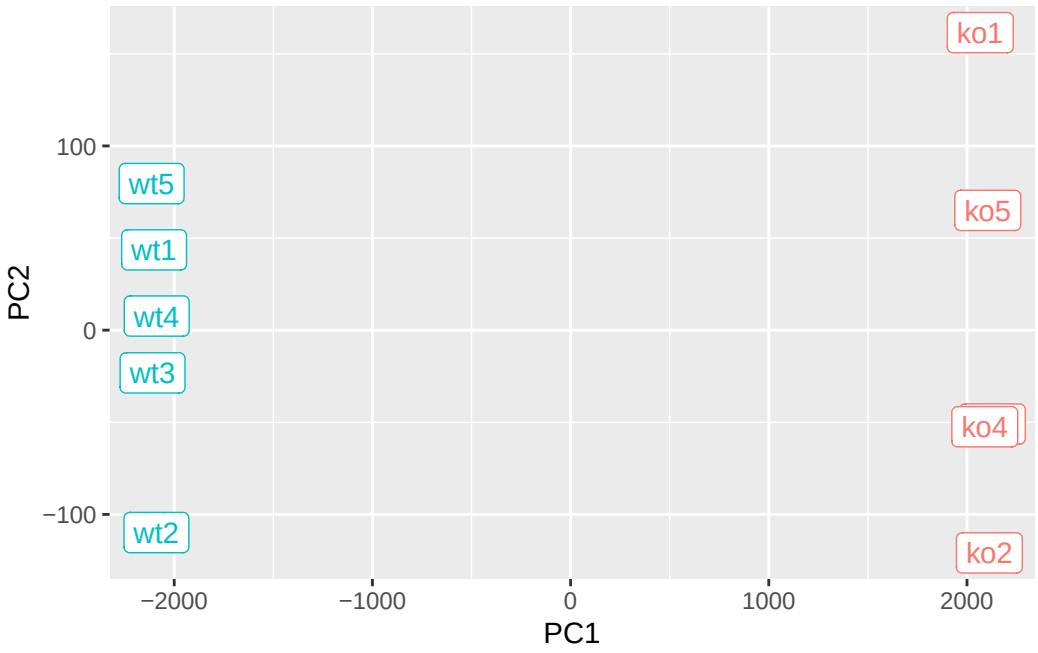


ggplot!

```
library(ggplot2)

df <- as.data.frame(pca$x)

# Our first basic plot
ggplot(df) +
  aes(PC1, PC2) +
  geom_point()
```




```
df$samples <- colnames(rna.data)
df$condition <- substr(colnames(rna.data),1,2)

p <- ggplot(df) +
        aes(PC1, PC2, label=samples, col=condition) +
        geom_label(show.legend = FALSE)
p
```


ko1
100
wt5
ko5
wt1
PC2
0
wt4
wt3
ko4
−100
wt2
ko2
−2000
−1000
0
1000
2000
PC1