class8

Rachel Diao

2/14/2022

Start by importing files

```
url <- "https://tinyurl.com/UK-foods"
x <- read.csv(url)

#Wanted only 4 columns, need to reindex
#Want food to be an index, not a column
rownames(x) <- x[,1]
x <- x[,-1]
head(x)
```

```
##                England Wales Scotland N.Ireland
## Cheese             105   103      103        66
## Carcass_meat       245   227      242       267
## Other_meat         685   803      750       586
## Fish               147   160      122        93
## Fats_and_oils      193   235      184       209
## Sugars             156   175      147       139
```

```
#Alternative: import csv with right formatting
#x <- read.csv(url, row.names=1)
```

Question 1

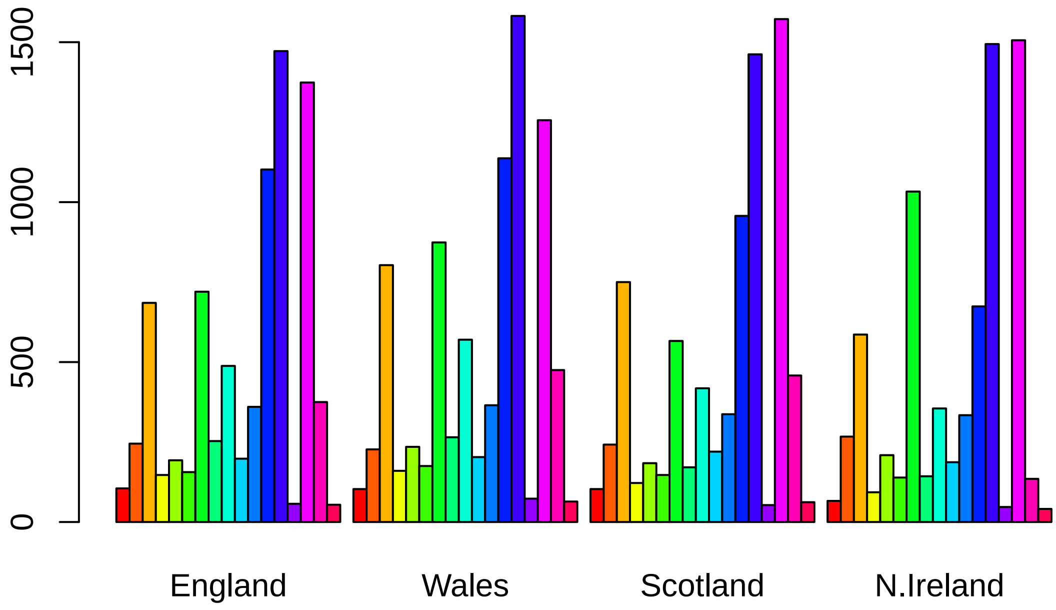
There are 17 rows and 5 columns in this table (before alterations). This can be determined with dim(), which returns the dimensions of a dataframe, or nrow() and ncol() in combination to find numbers of rows and columns respectively.

Question 2

I would prefer to import the CSV with the correct format without having to alter it, because there is less chance of there being some sort of re-indexing errors if you import the file with the correct dimensions and indexing to begin with.

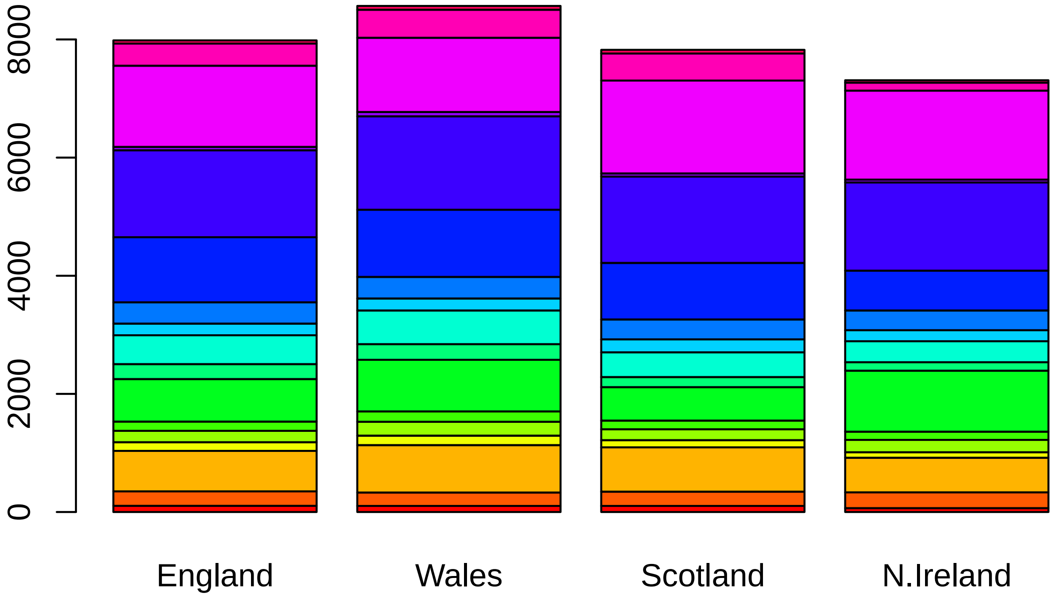
Make a barplot to visualize the data, which is showing how much of each food group is consumed per person in each part of the UK per week (in grams). Each color is a different food group.

```
barplot(as.matrix(x), beside=T, col=rainbow(nrow(x)))
```



## Question 3 IF you set optional argument beside=FALSE (the default), you get the following visualization, which provides a more direct side-by-side comparison of percentage of each food group is consumed per person per week in each part of the UK (gives you better comparison of TOTALS).

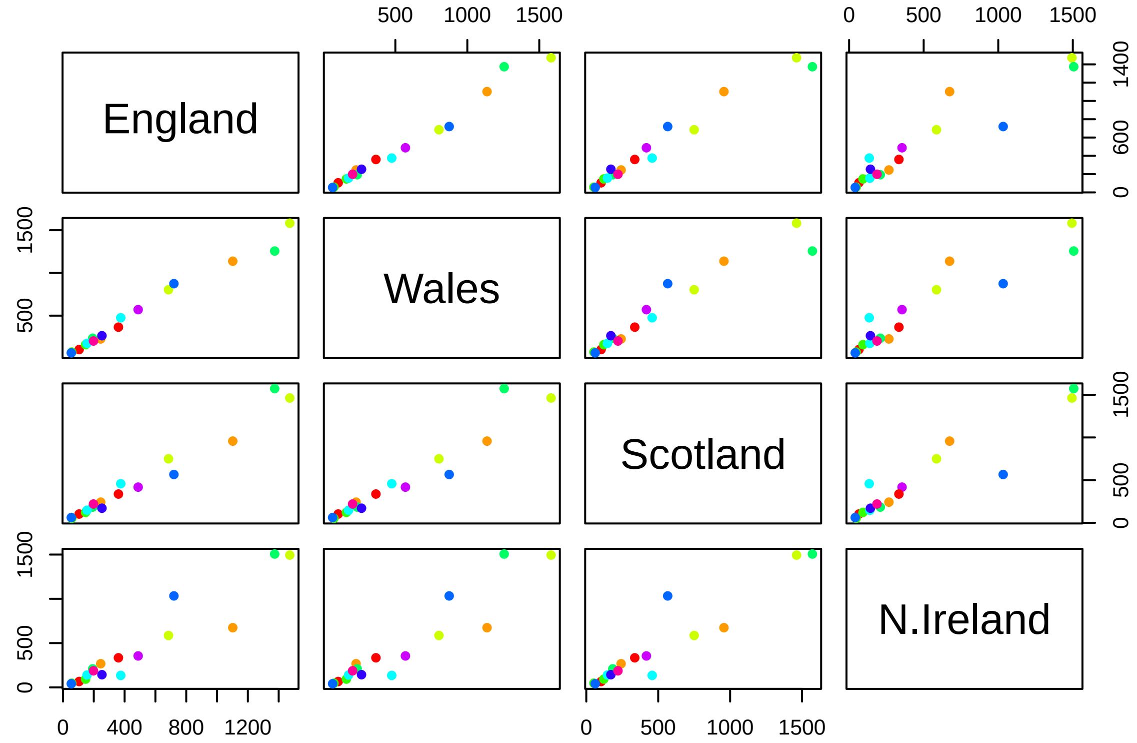
```
barplot(as.matrix(x), beside=FALSE, col=rainbow(nrow(x)))
```



Question 5

Generate pairwise plots.These are pairwise comparisons of the amt. of each food group consumed between every combination of pairs of parts of the UK. Anything that falls on the diagonal is a similarity between the pairs (i.e. similar amt of food consumed between people in these two diff. parts of the UK).

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



Question 6

The average person in Northern Ireland seems to consume a lot more fresh potatoes per week than the other parts of the UK, but also less of other meat, fresh/other vegetables, and fresh fruit.

PCA analysis with prcomp, t(x) to transpose the data table such that it's usable. The summary essentially means that 97% of the variance can be explained by the first two PCs.

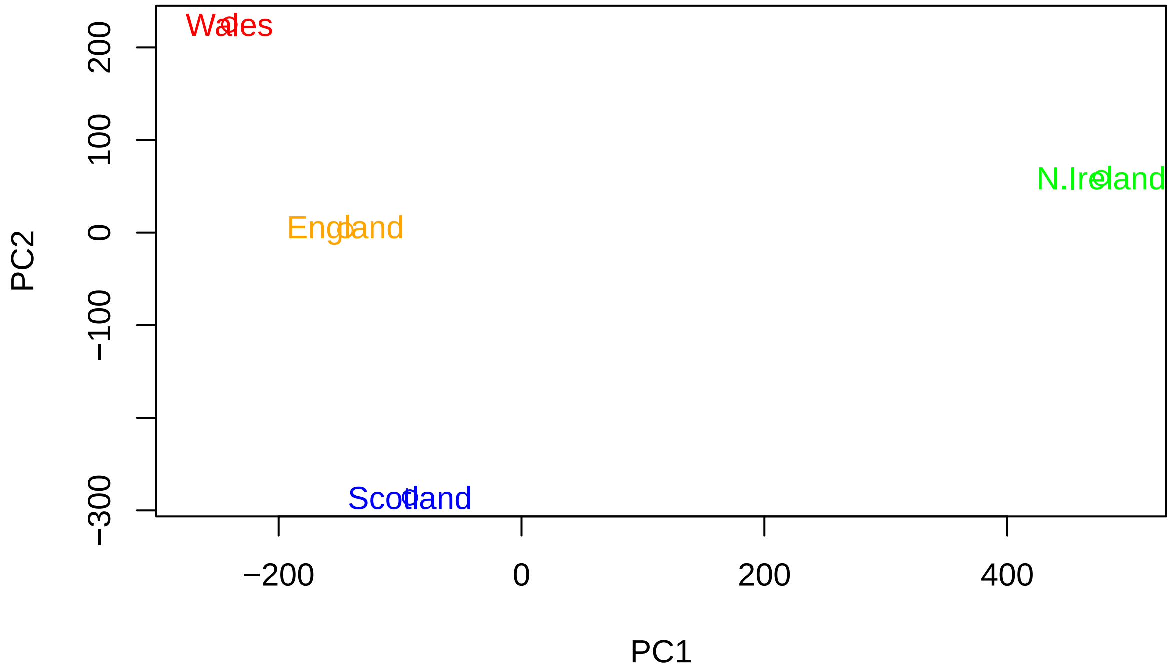
```
pca <- prcomp( t(x) )
summary(pca)
```

```
## 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
```

Question 7 and 8

Make a PCA plot!

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



How much variation does each principle component account for?

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

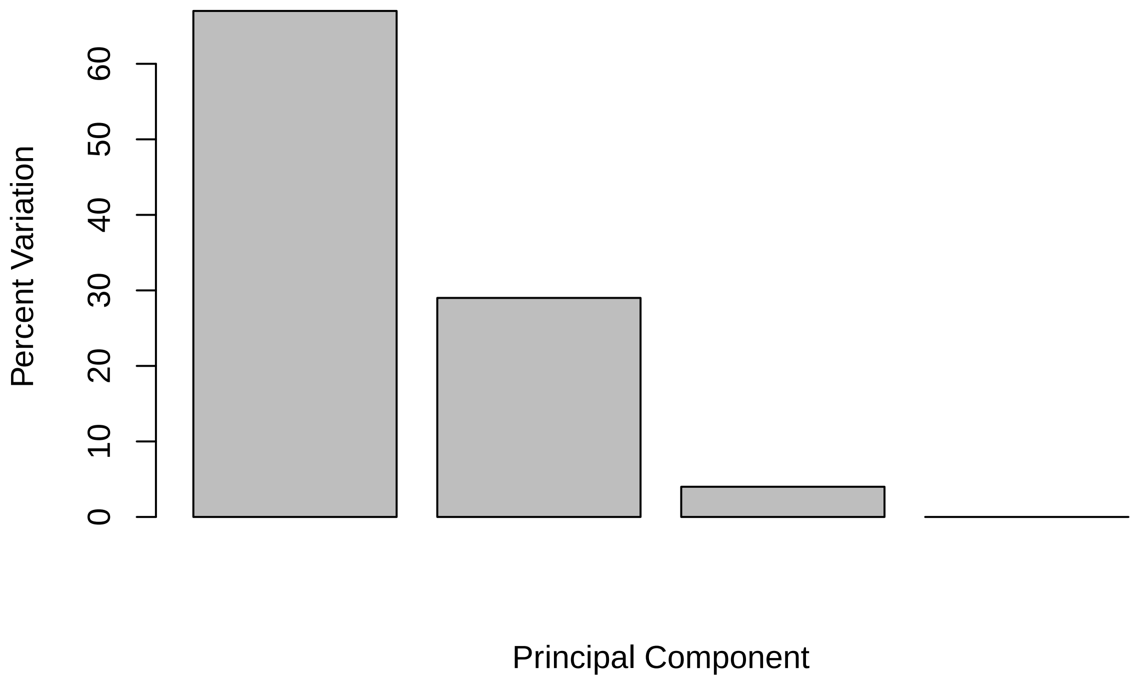
```
## [1] 67 29  4  0
```

```
#OR:
z <- summary(pca)
z$importance
```

```
##                              PC1       PC2      PC3          PC4
## Standard deviation     324.15019 212.74780 73.87622 4.188568e-14
## Proportion of Variance   0.67444   0.29052  0.03503 0.000000e+00
## Cumulative Proportion    0.67444   0.96497  1.00000 1.000000e+00
```

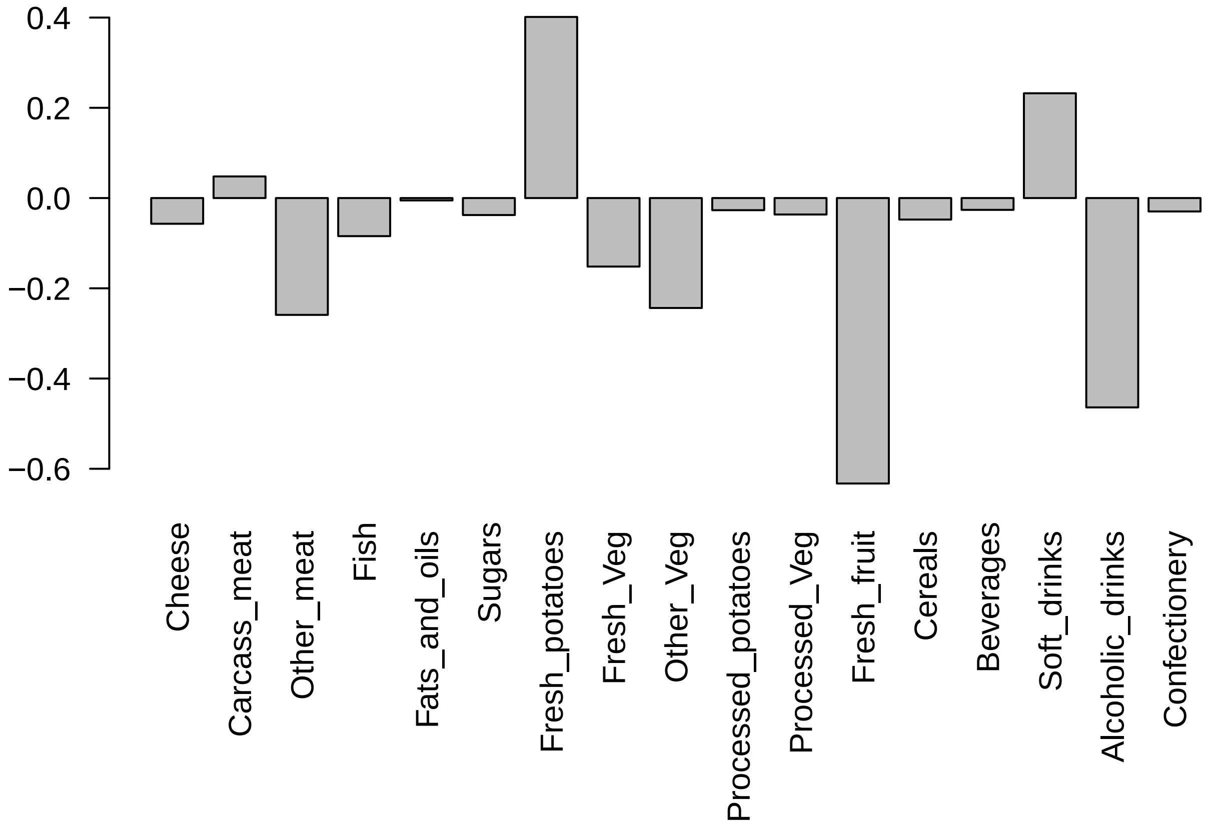
Graph the percent of variation each PC accounts for:

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



Loading scores - consider the impact of original variables on the principle components. This will show us more specifically what's driving the variation.

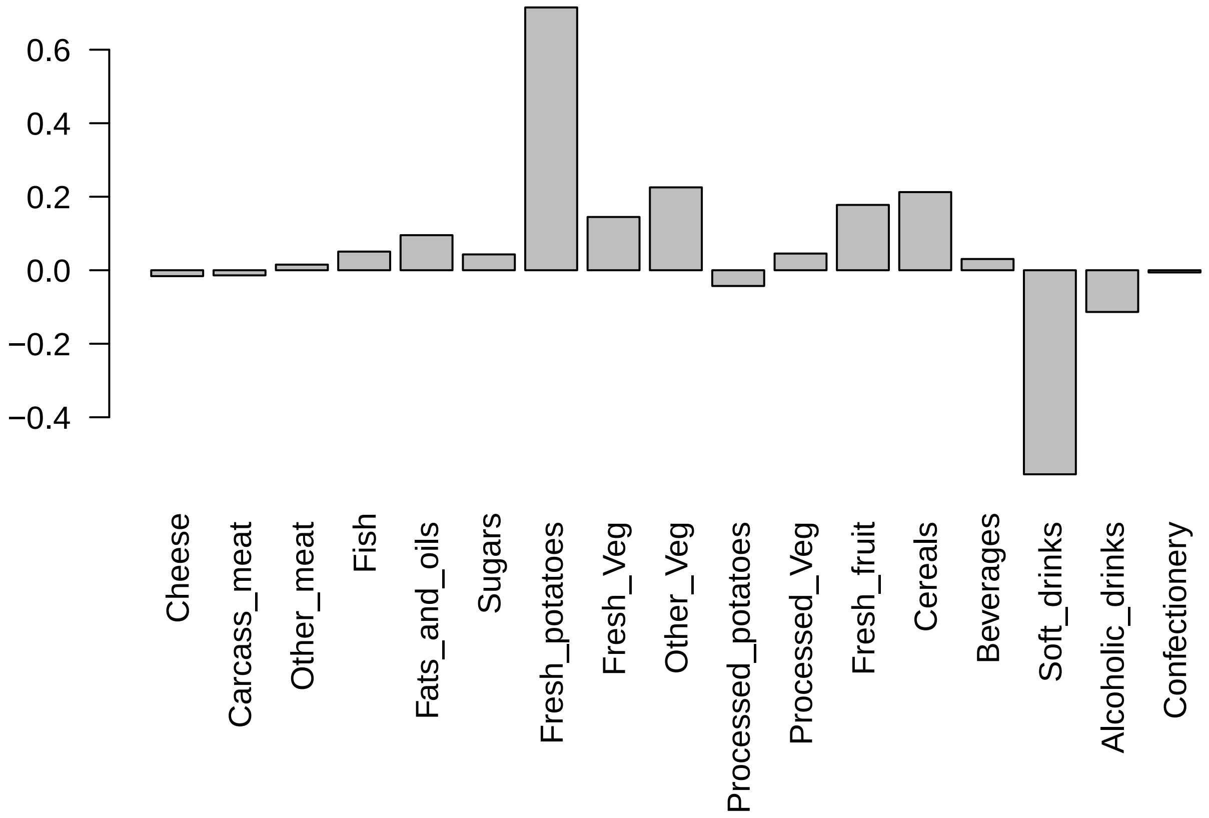
```
par(mar=c(10, 3, 0.35, 0))
barplot( pca$rotation[,1], las=2 )
```



Question 9

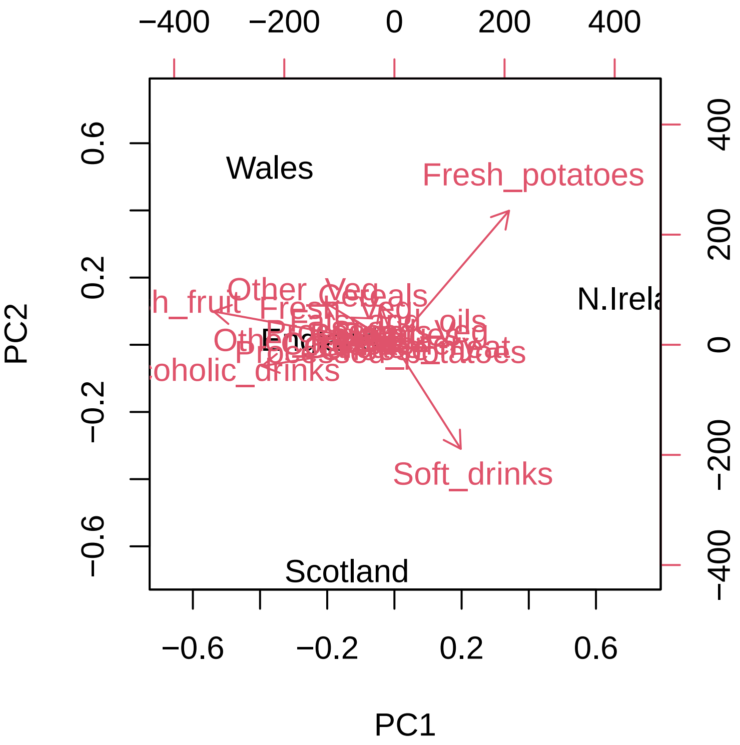
Fresh potatoes and soft drinks feature prominently in PC2, and this PC essentially indicates that individuals in Scotland likely drink more soft drinks per week, driving their score down to the negatives. England and Northern Ireland likely also have very high soft drink consumption but this is balanced out by some of the positive values shown (i.e. fresh potatoes), as indicated by their scores hovering around 0, and individuals in Wales likely consume the least soft drinks per week.

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



Next test out the biplot. THis display shows which categories of foods contribute to the PC values for each country.

```
biplot(pca)
```



RNAseq data

Import file and look at the top of the dataframe:

```
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
```

Question 10

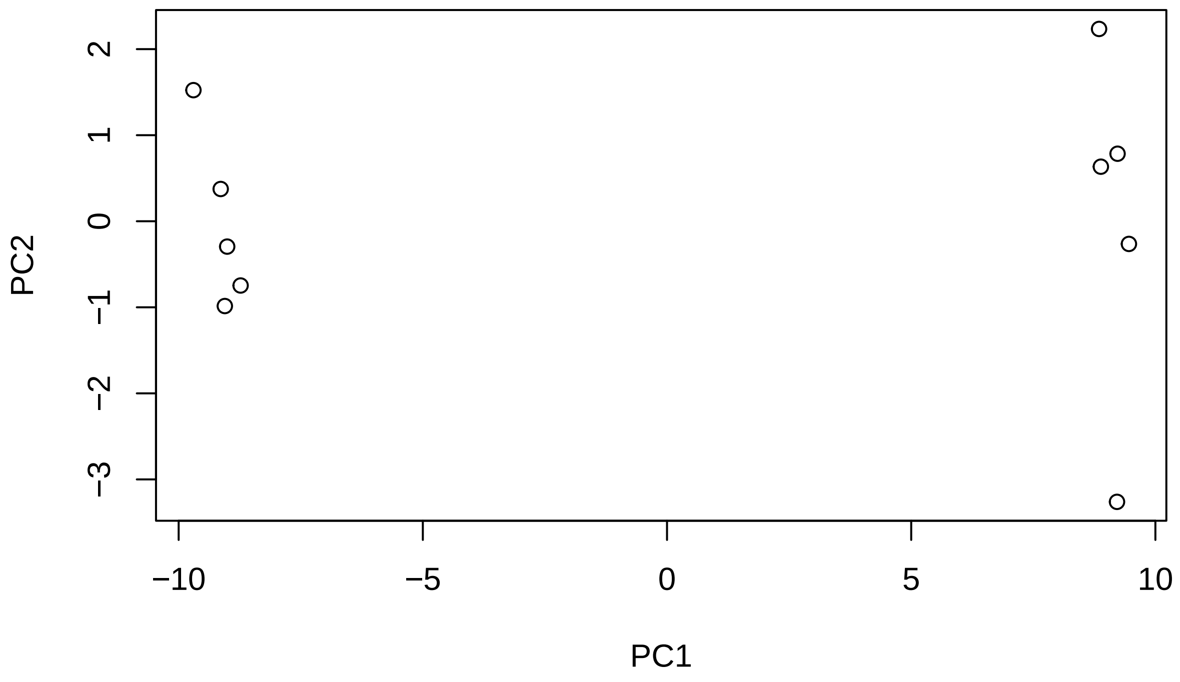
There are 10 samples (5 wt, 5 ko) and 100 genes in this dataset.

Next make a PCA plot

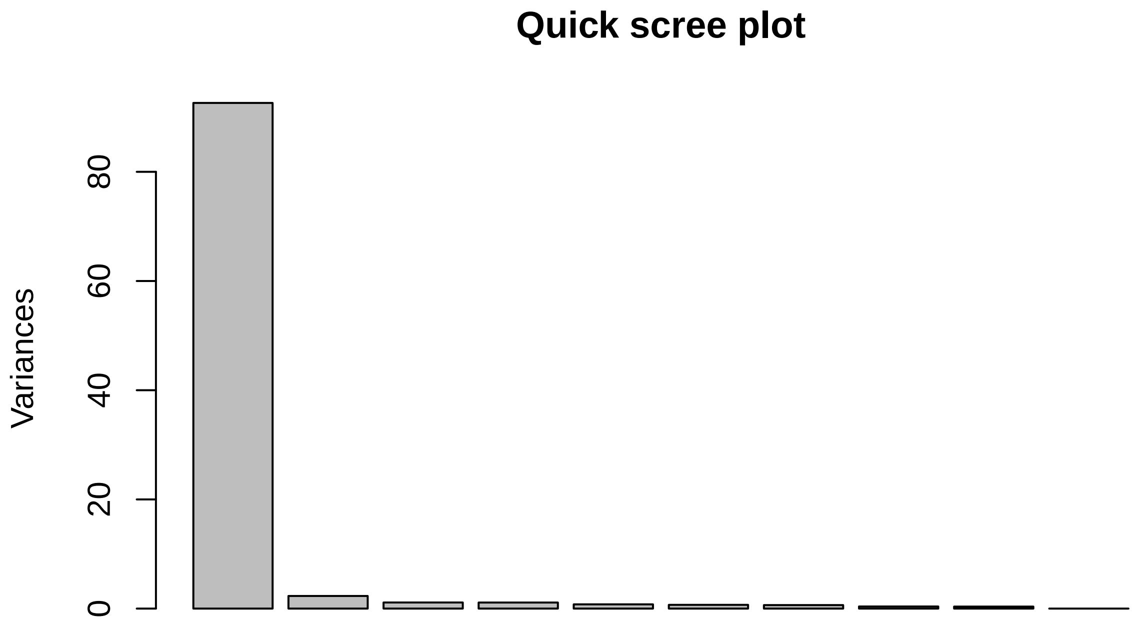
```
## Again we have to take the transpose of our data
pca <- prcomp(t(rna.data), scale=TRUE)
summary(pca)
```

```
## 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 3.348e-15
## Proportion of Variance 0.00385 0.00364 0.000e+00
## Cumulative Proportion  0.99636 1.00000 1.000e+00
```

```
## Simple un polished plot of pc1 and pc2
plot(pca$x[,1], pca$x[,2], xlab="PC1", ylab="PC2")
```



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

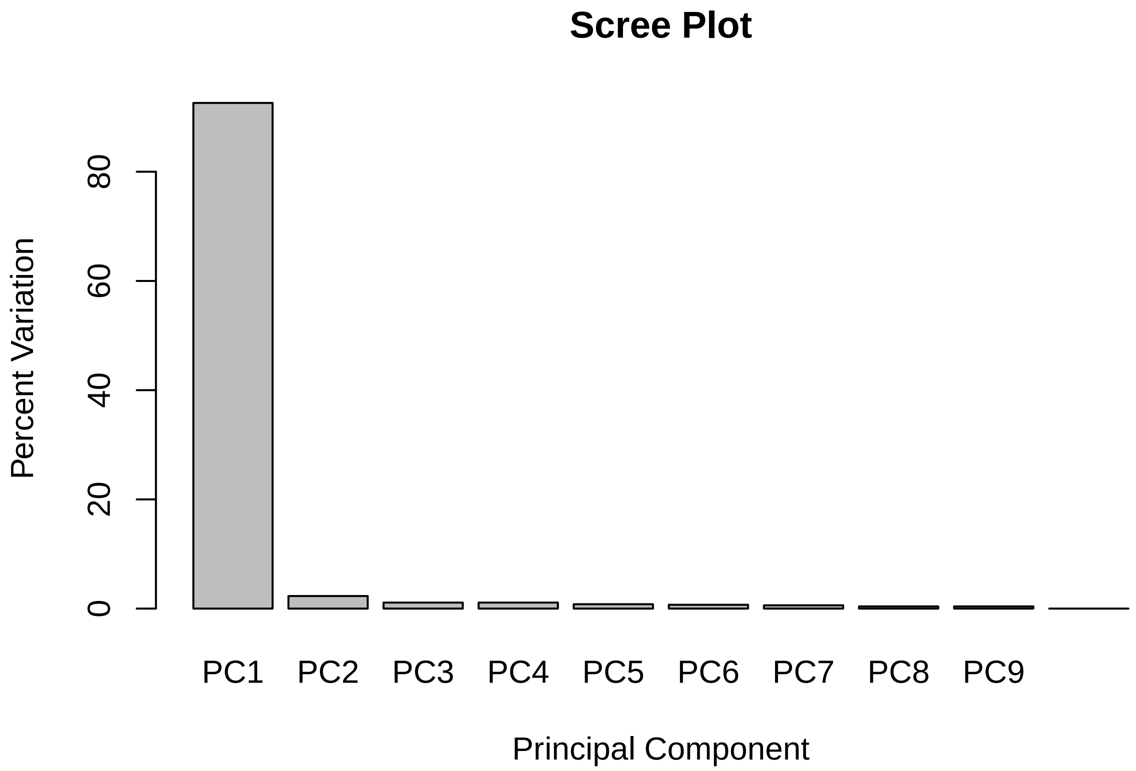


```
## Variance captured per PC
pca.var <- pca$sdev^2

## 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
```

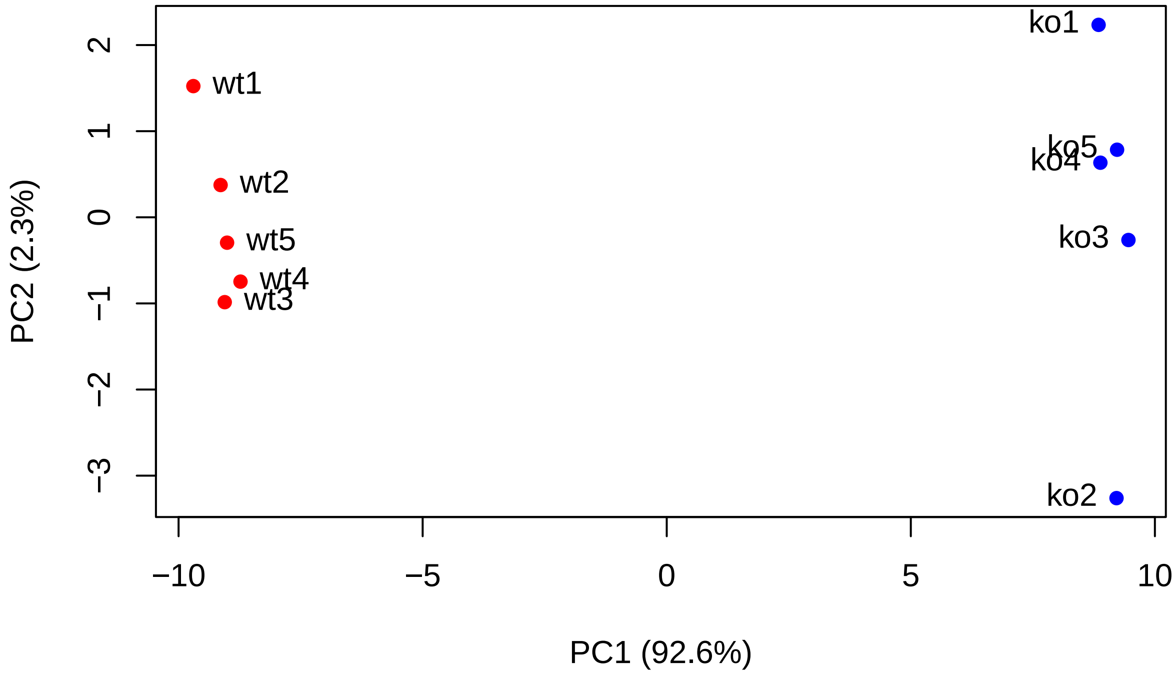
Same scree plot but a lot more useful bc it's actually labeled

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



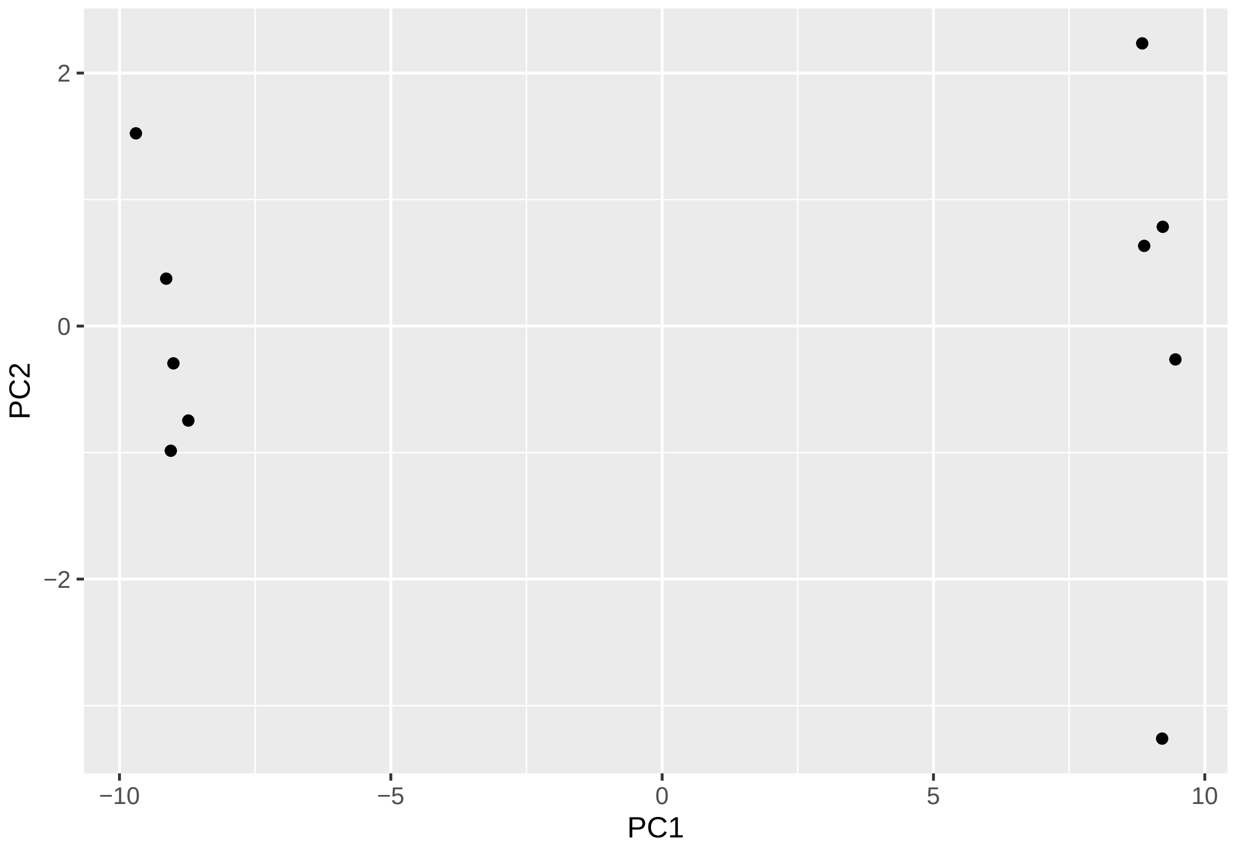
Make our PCA plot prettier and more useful.

```
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)))
```



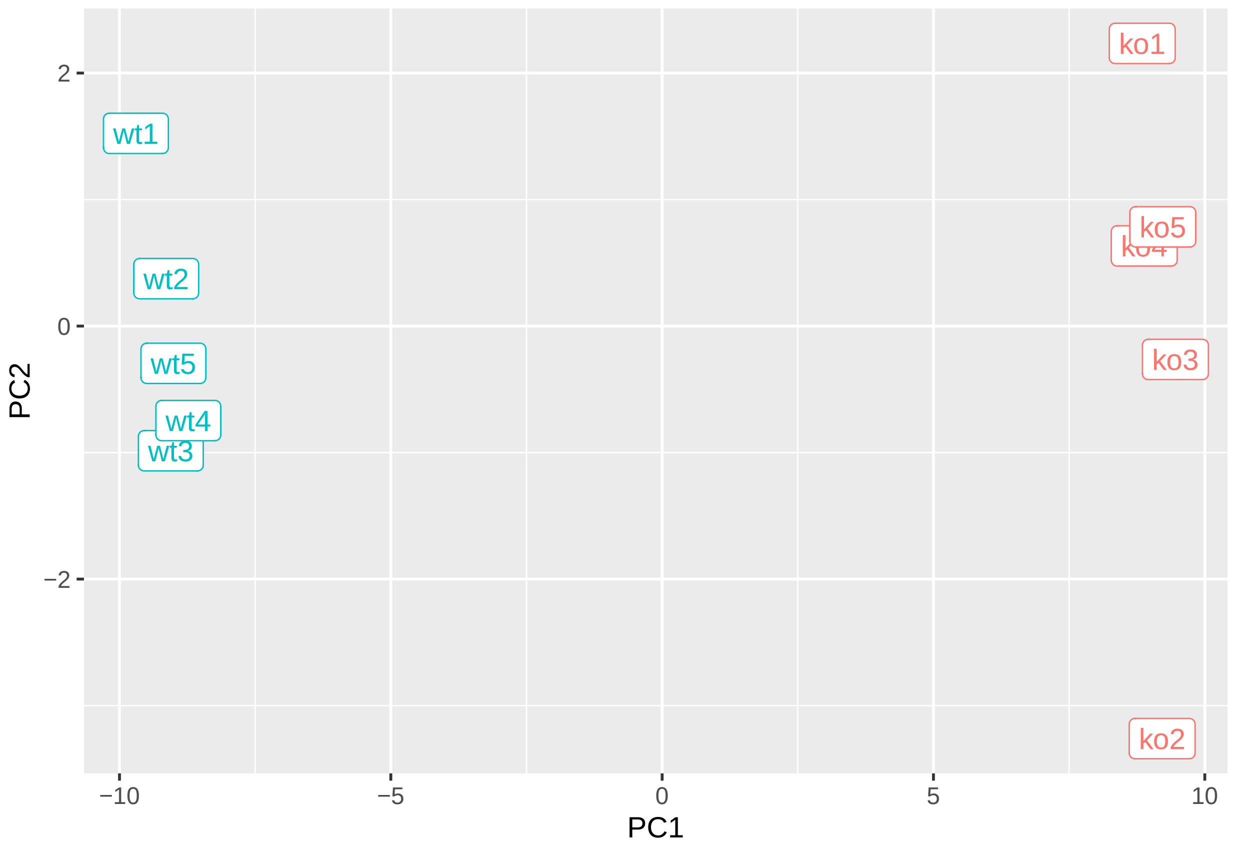
Now do the same in ggplot2!

```
library(ggplot2)

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

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

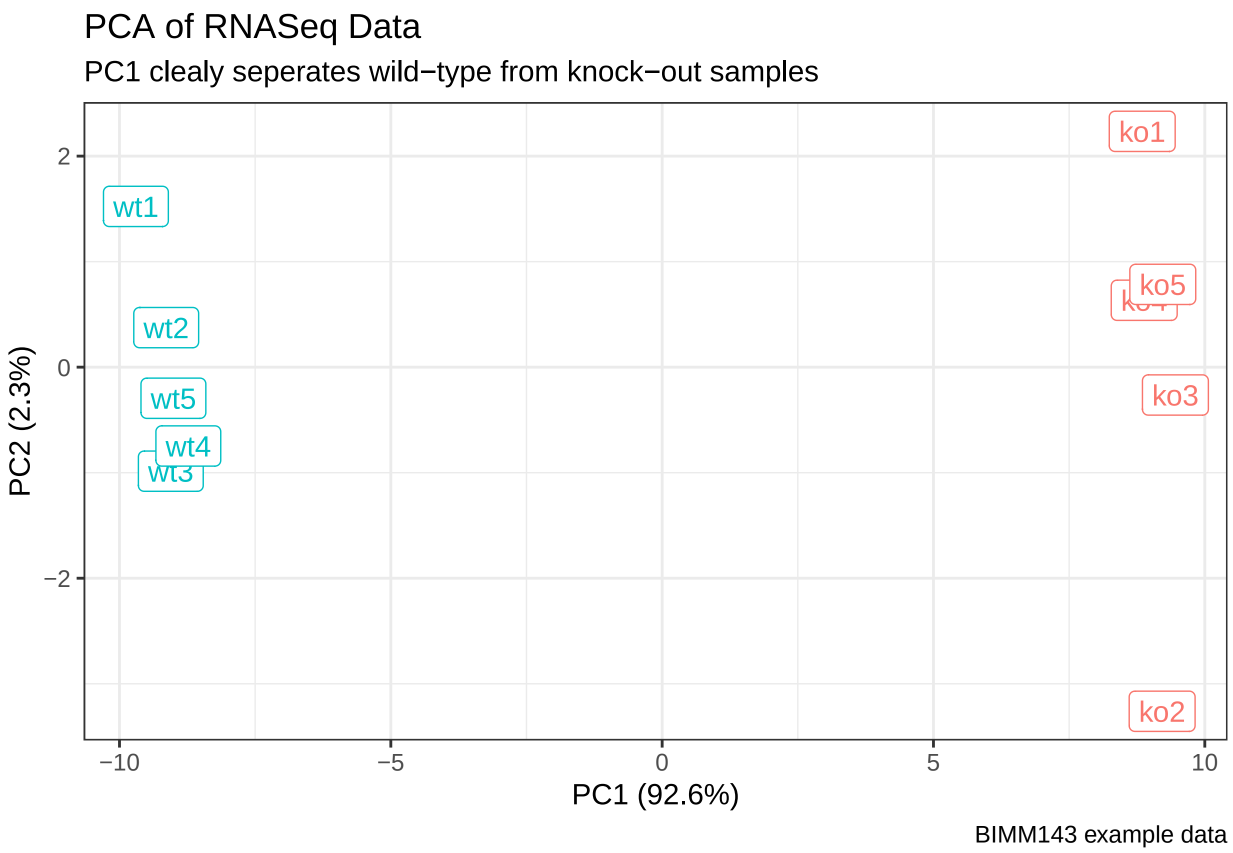


Now make it look better haha

```
# Add a 'wt' and 'ko' "condition" column
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
```



```
p + labs(title="PCA of RNASeq Data",
       subtitle = "PC1 clealy seperates wild-type from knock-out samples",
       x=paste0("PC1 (", pca.var.per[1], "%)"),
       y=paste0("PC2 (", pca.var.per[2], "%)"),
       caption="BIMM143 example data") +
     theme_bw()
```



PCA of RNASeq Data

PC1 clealy seperates wild−type from knock−out samples

BIMM143 example data