class07

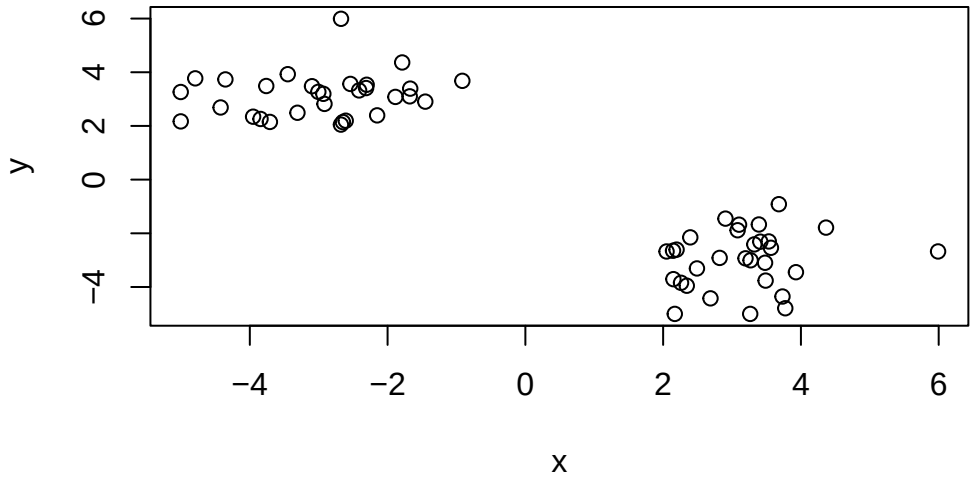
Steven Gan

2022-10-19

Table of contents

K-means Cluster

```
tmp <- c(rnorm(30, -3), rnorm(30, 3))
tmp <- cbind(x = tmp, y = rev(tmp))

plot(tmp)
```




```
km2 <- kmeans(tmp, centers = 2, nstart = 20)

km2$size
```

```
[1] 30 30
```

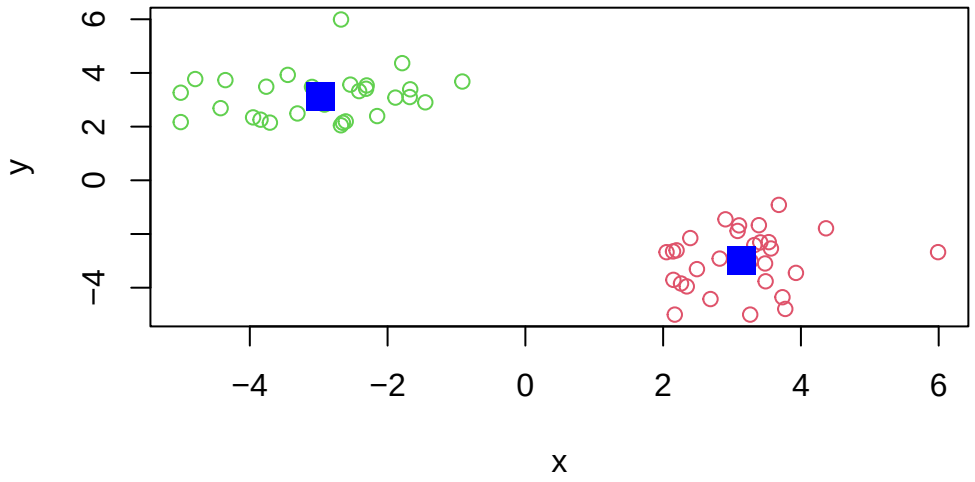
```
sum(km2$cluster == 1)
```

```
[1] 30
```

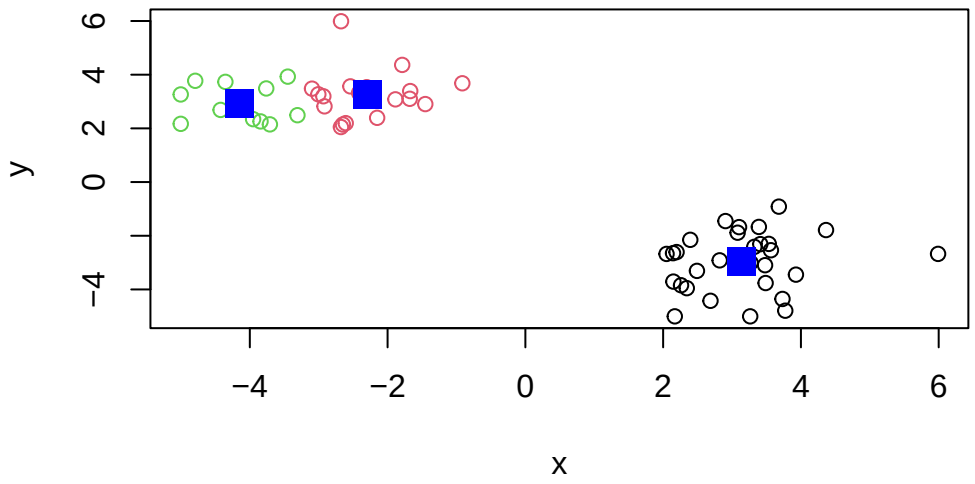
```
sum(km2$cluster == 2)
```

```
[1] 30
```

```r
plot(tmp, col = km2$cluster + 1)
points(km2$centers, col = "blue", pch = 15, cex = 2)
```




```r
km4 <- kmeans(tmp, centers = 3, nstart = 20)
plot(tmp, col = km4$cluster)
points(km4$centers, col = "blue", pch = 15, cex = 2)
```

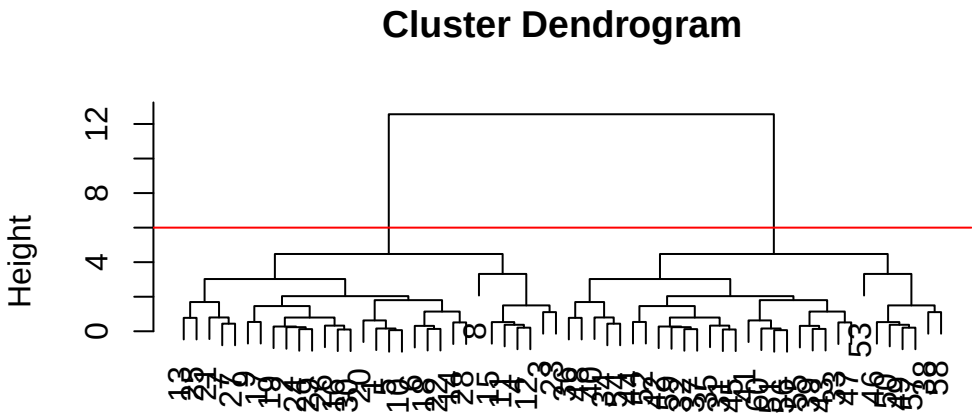


Hierachical Cluster

```
hc <- hclust(dist(tmp))
hc
```

```
Call:
hclust(d = dist(tmp))

Cluster method   : complete
Distance         : euclidean
Number of objects: 60
```

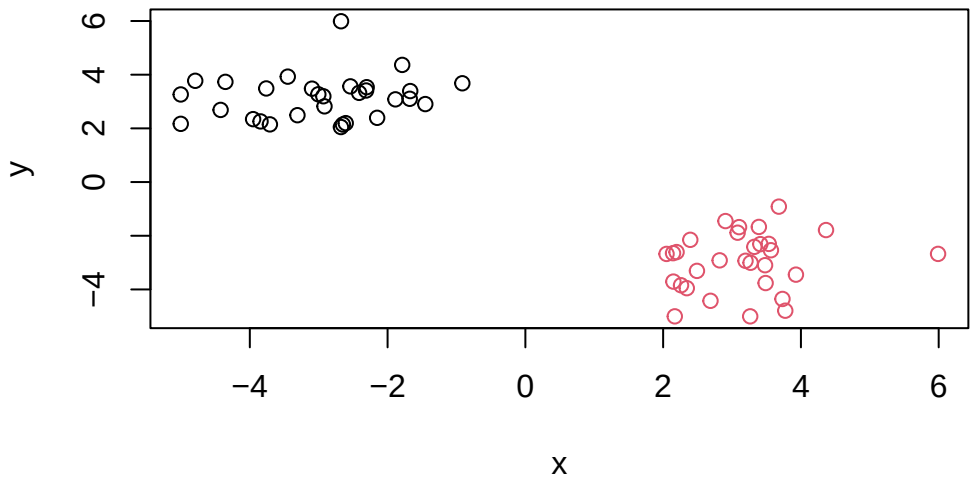
```
plot(hc)
abline(h = 6, col = "red")
```



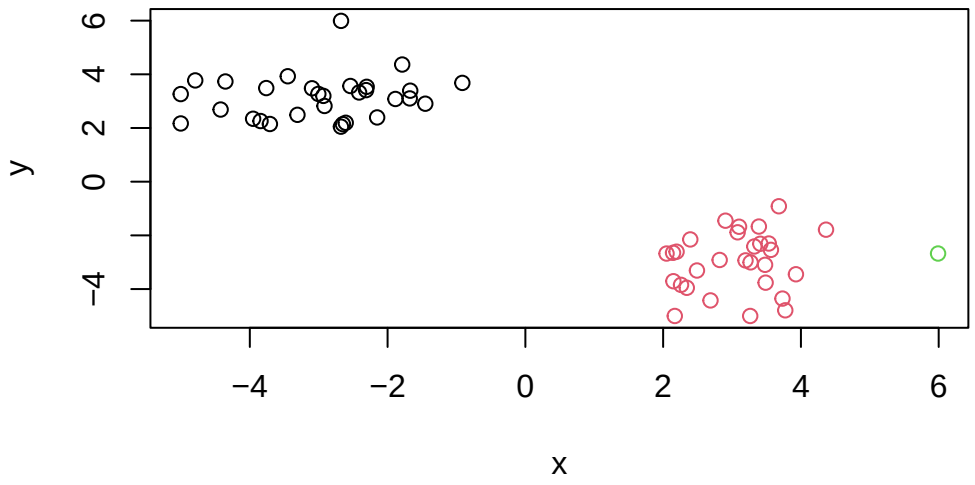

```
cutree(hc, k = 2)
```

```
 [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2
[39] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
```

```
plot(tmp, col = cutree(hc, k = 2))
```




```
hc <- hclust(dist(tmp), "average")
plot(tmp, col = cutree(hc, k = 3))
```



Principal Component Analysis (PCA)

PCA of UK food data

Data import

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

Checking your data

Q1: 17 rows and 5 columns dim()

Q2: row.names = 1 is more preferred as it's more elegant in coding. However, the dirst method is more robust and contains more freedom fo adjustment based on personal requirements.

```
dim(x)
```

```
[1] 17  5
```

```
head(x)
```

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

```
rownames(x) <- x[,1]
x <- x[,-1]
# x <- read.csv("https://tinyurl.com/UK-foods", row.names = 1)
dim(x)
```

```
[1] 17  4
```

```
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
```

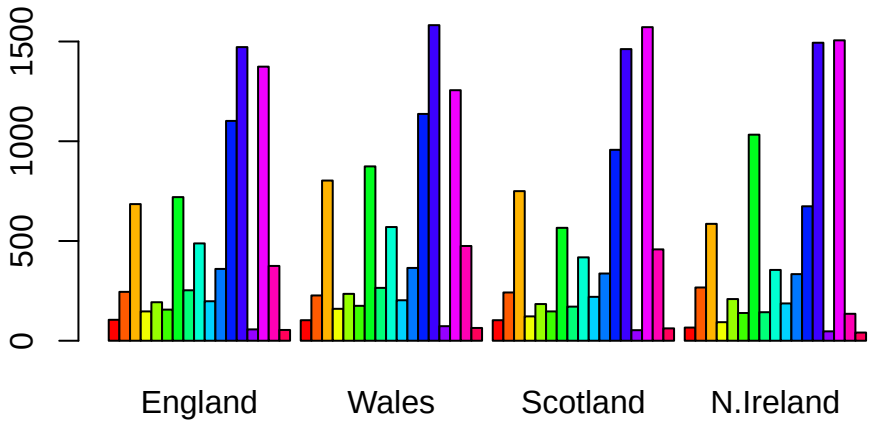
Spotting major differences and trends

Q3: Changing the beside argument from TRUE to FALSE

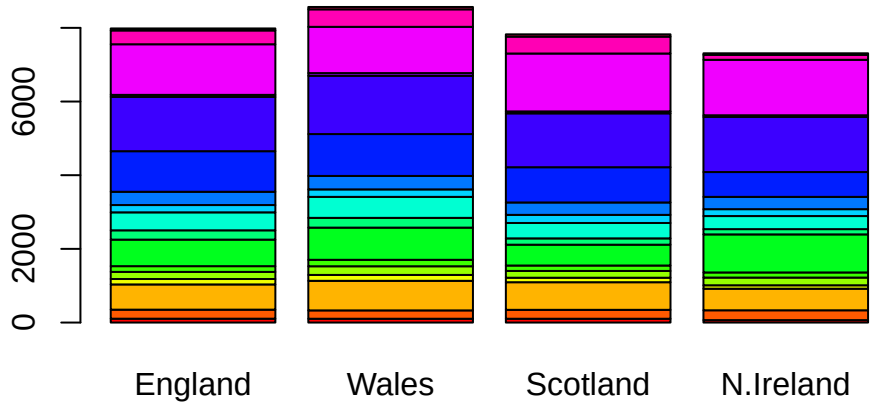
Q5: Each plot means the pairwise relations between the corresponding regions. For a point lies on the diagonal of a plot, it means that type of food consumption are the same in that two corresponding regions.

Q6: N.Ireland exhibits relatively different food consumption patterns in comparing to other regions, which exhibit similar food consumption patterns.

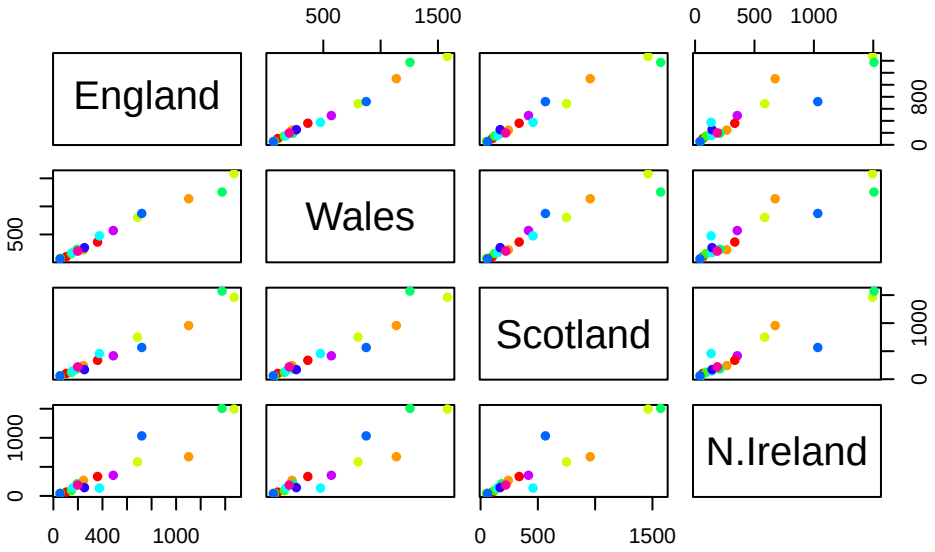
```
barplot(as.matrix(x), beside = T, col = rainbow(nrow(x)))
```



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




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



PCA to the rescue

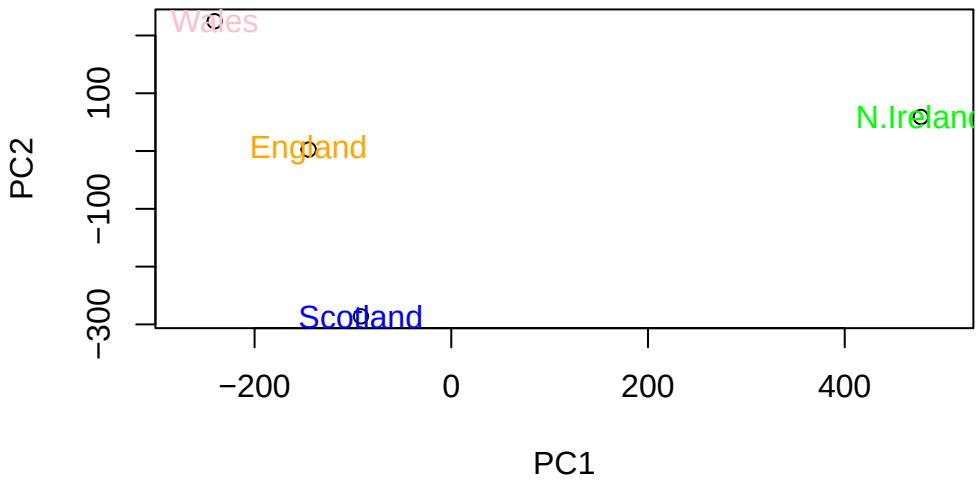
Q7: As below

Q8: As below

```
# Transpose!
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
```

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



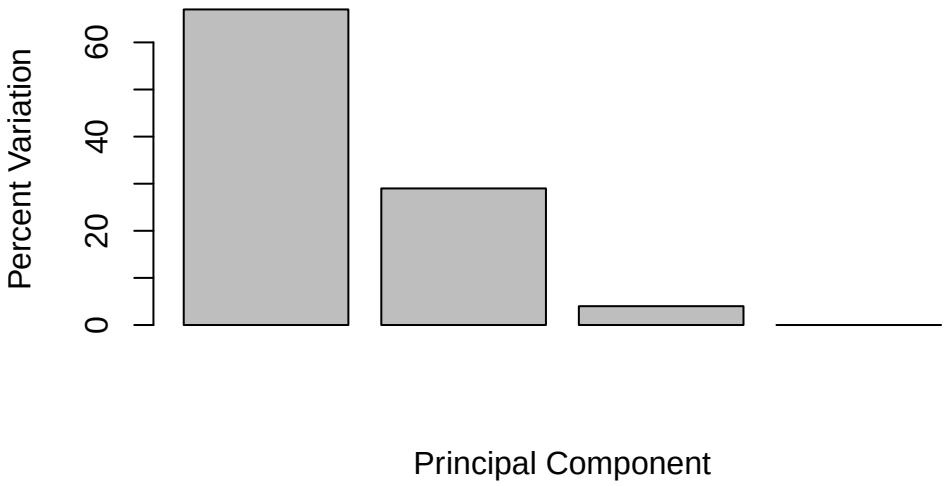

```
v <- round(pca$sdev^2/sum(pca$sdev^2) * 100)
v
```

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

```
z <- summary(pca)
z$importance
```

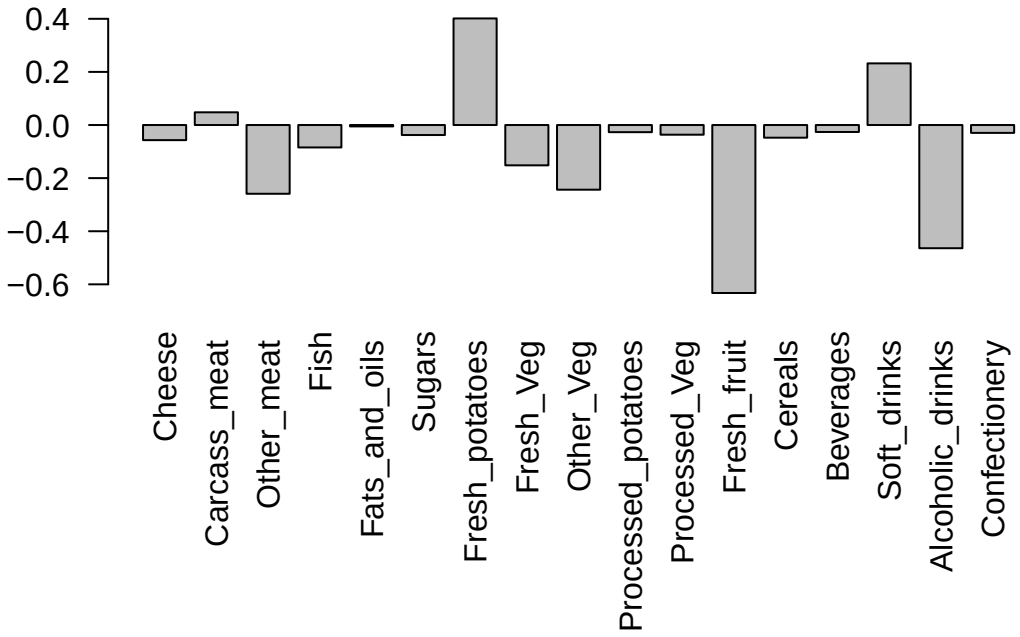
```
                             PC1       PC2      PC3          PC4
Standard deviation     324.15019 212.74780 73.87622 4.188568e-14
Proportion of Variance   0.67444   0.29052  0.03503 0.000000e+00
Cumulative Proportion    0.67444   0.96497  1.00000 1.000000e+00
```

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



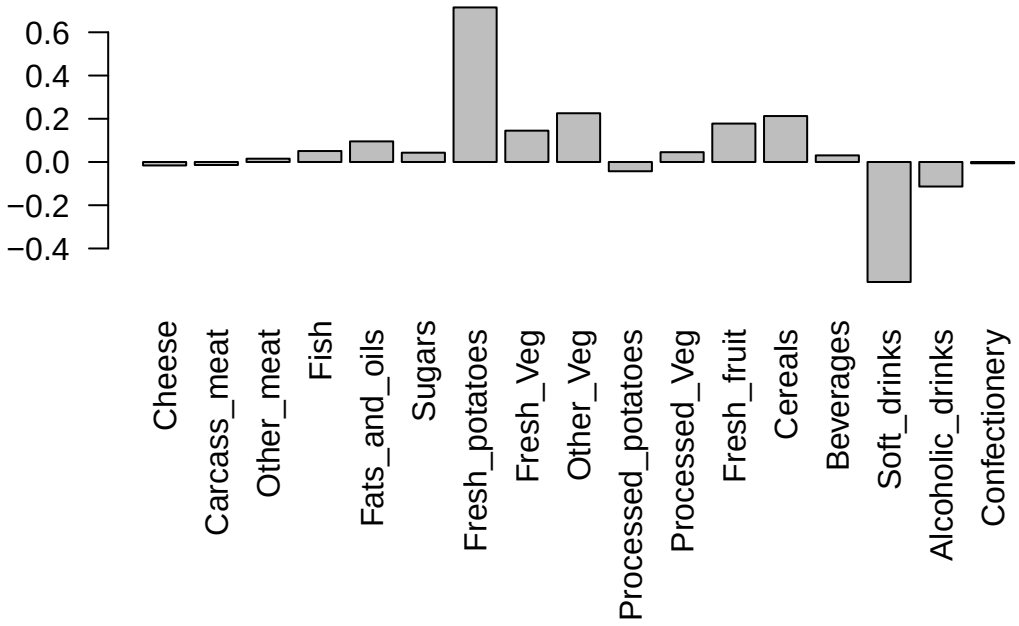
Digging deeper

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

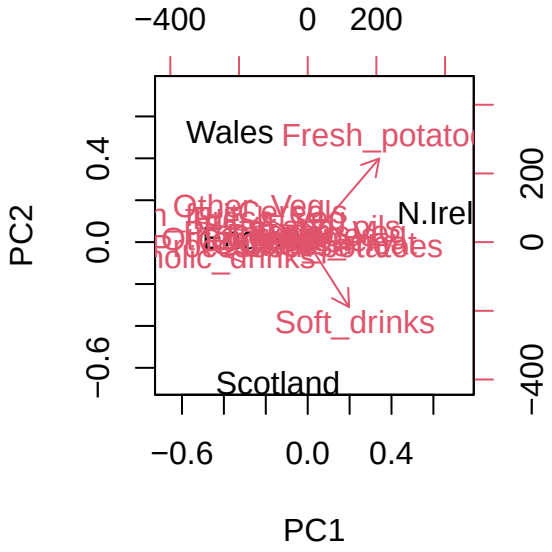


Q9: As below Fresh potatoes and soft drinks dominate. The PC2 mainly tells how the three sets: Wales, England and N.Ireland, and Scotland distinguish each others in the food consumption patterns. Fresh potatoes push the Wales away while soft drinks push Scotland away.

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




```
## The inbuilt biplot() can be useful for small datasets
biplot(pca)
```



PCA of RNA-seq data

Q10: 100 genes and 10 samples are in this data set

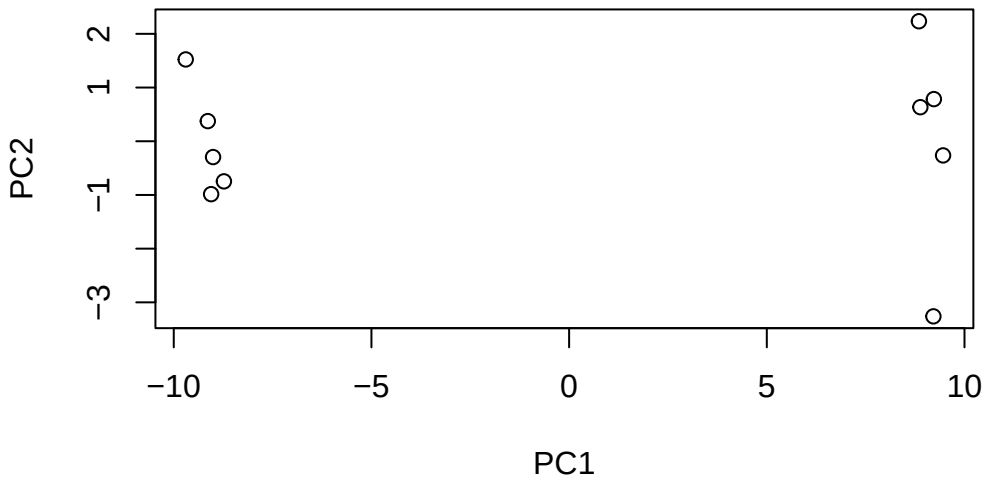
```
url2 <- "https://tinyurl.com/expression-CSV"
rna.data <- read.csv(url2, row.names = 1)
head(rna.data)
```

```
       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
```

```
dim(rna.data)
```

```
[1] 100  10
```

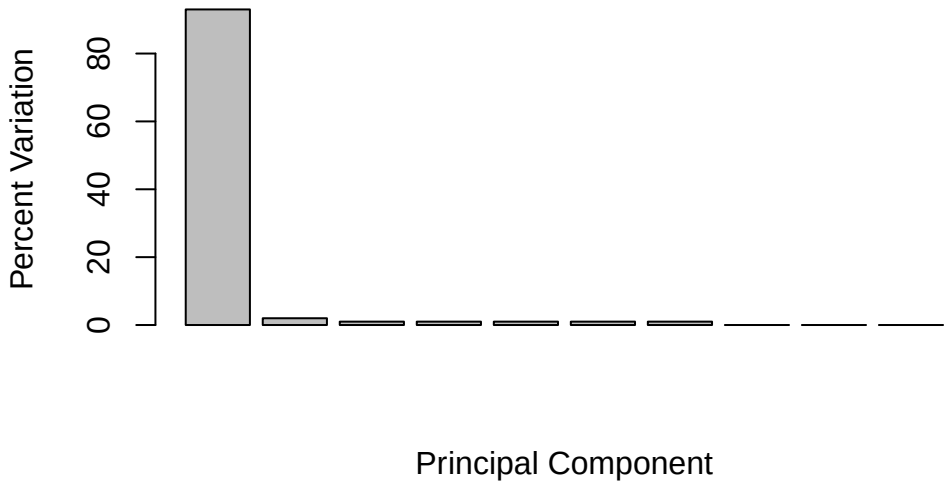
```
# Transpose!
pca.rna <- prcomp(t(rna.data), scale = TRUE)
plot(pca.rna$x[,1], pca.rna$x[,2], xlab = "PC1", ylab = "PC2")
```



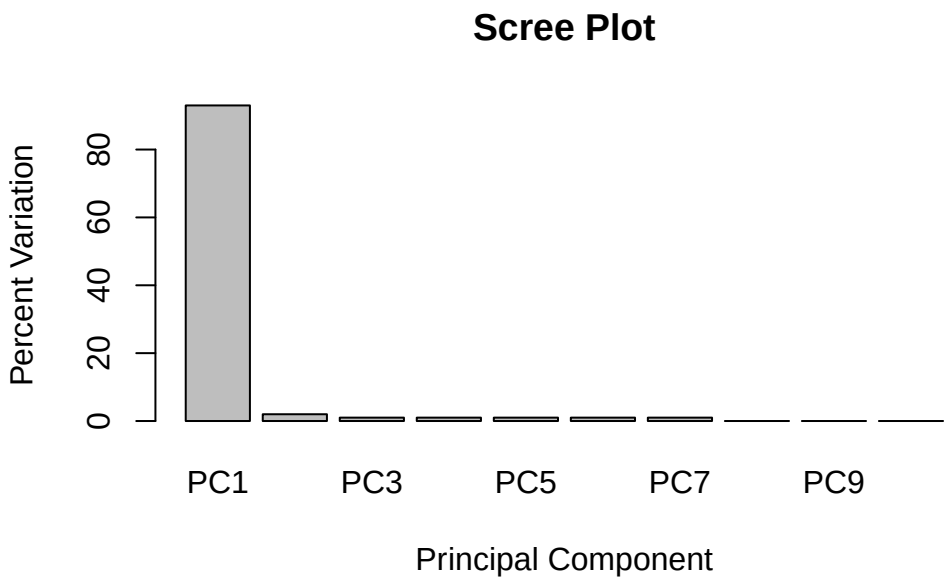

```
summary(pca.rna)
```

```
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
```

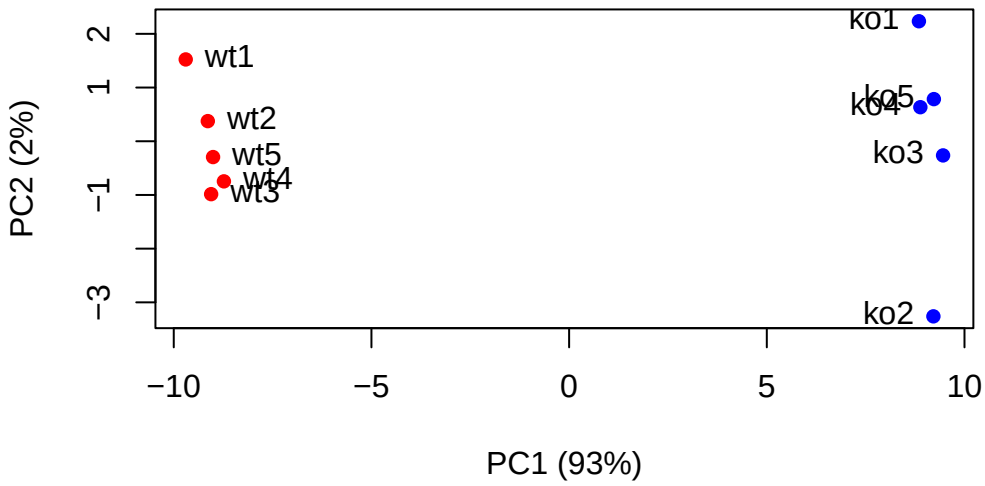
```
# plot(pca.rna, main = "Quick scree plot")
v.rna <- round(pca.rna$sdev ^ 2 / sum(pca.rna$sdev ^ 2) *100)
barplot(v.rna, xlab = "Principal Component", ylab = "Percent Variation")
```




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

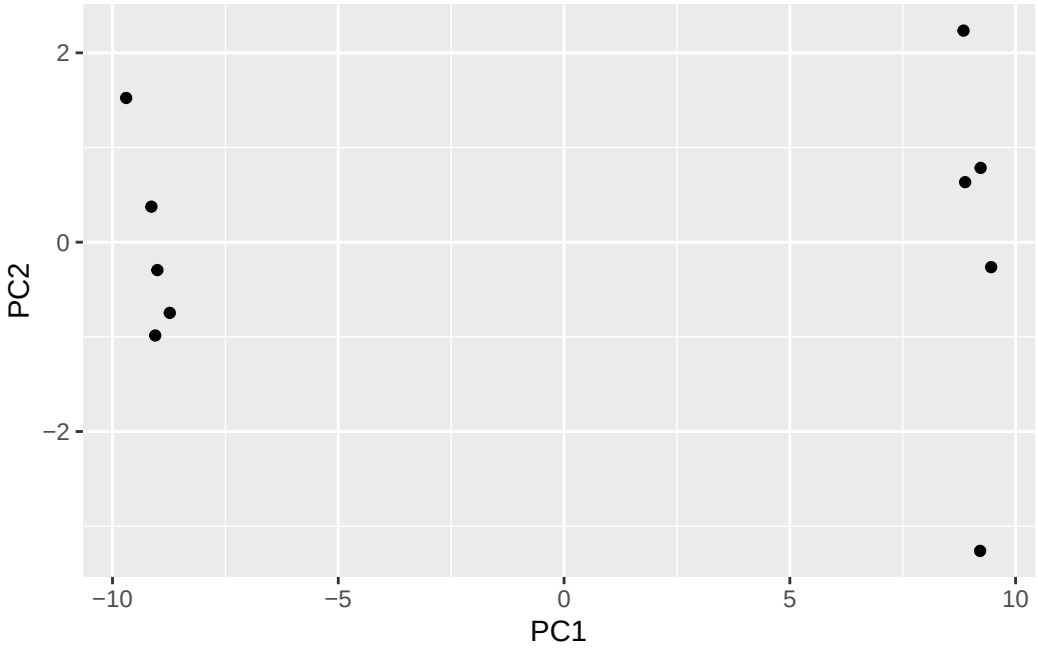



```
colvec <- colnames(rna.data)
colvec[grep("wt", colvec)] <- "red"
colvec[grep("ko", colvec)] <- "blue"

plot(pca.rna$x[,1], pca.rna$x[,2], col = colvec, pch = 16,
     xlab = paste0("PC1 (", v.rna[1], "%)"),
     ylab = paste0("PC2 (", v.rna[2], "%)"))
text(pca.rna$x[,1], pca.rna$x[,2], labels = colnames(rna.data), pos = c(rep(4,5), rep(2,5)
```

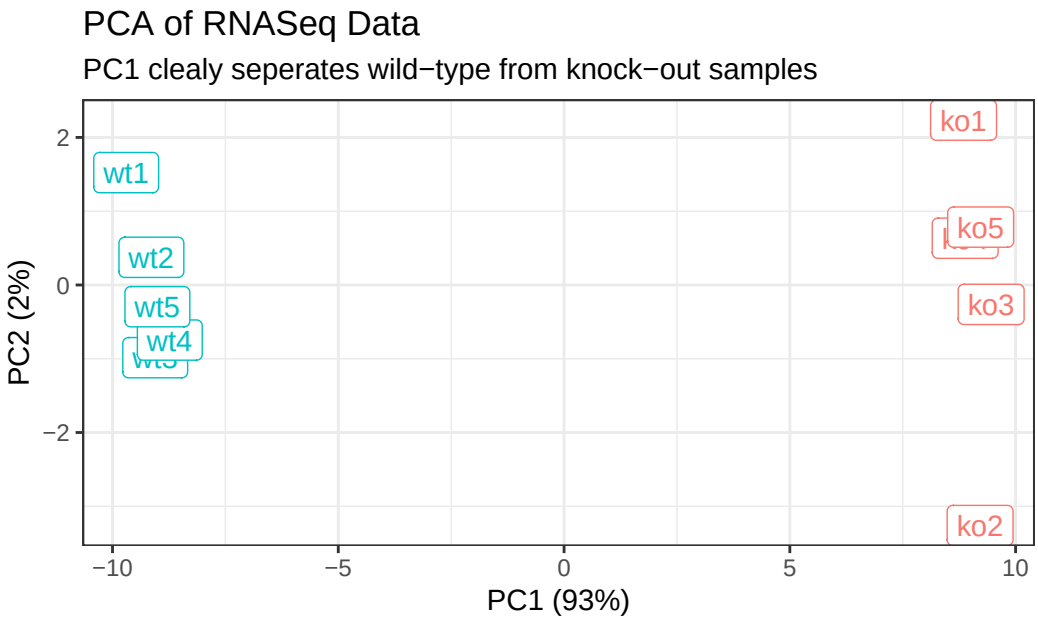


Using ggplot2

```
library(ggplot2)

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

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



Optional: gene loadings

```
gene_scores <- abs(pca.rna$rotation[,1])
gene_score_ranked <- sort(gene_scores, decreasing = TRUE)

names(gene_score_ranked[1:10])
```

```
 [1] "gene100" "gene66"  "gene45"  "gene68"  "gene98"  "gene60"  "gene21"
 [8] "gene56"  "gene10"  "gene90"
```