class 7

Kelly Isbell (A59019188)

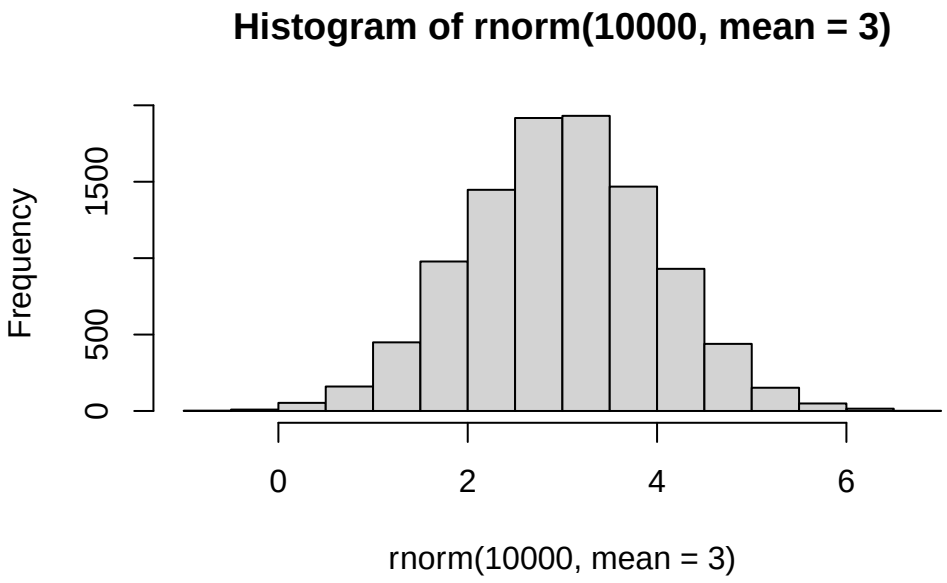
Clustering

k-means clustering, one of the most prevelent of all clustering

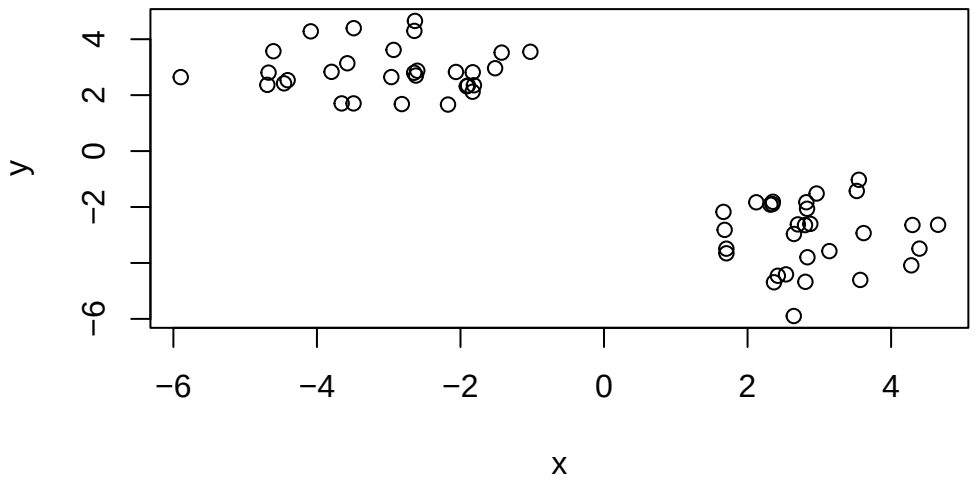
```
rnorm(10)
```

```
 [1] -0.7983101 -0.1704649  0.4686668  0.4041775  0.5077747  0.2092158
 [7] -0.6526601  0.5126342  0.7951054  1.1576468
```

```
hist(rnorm(10000, mean=3))
```



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




```r
#The main function in R for k-means clustering is called 'kmeans()'
k <- kmeans(x, centers=2, nstart=20)
k
```

```
K-means clustering with 2 clusters of sizes 30, 30

Cluster means:
          x         y
1 -3.005755  2.872068
2  2.872068 -3.005755

Clustering vector:
 [1] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 1 1 1 1 1 1
[39] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

Within cluster sum of squares by cluster:
[1] 60.46709 60.46709
```

```
 (between_SS / total_SS =  89.6 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
[6] "betweenss"    "size"         "iter"         "ifault"
```

#Q1. How many points are in each cluster?

```
k$size
```

```
[1] 30 30
```

#Q2. The clustering result i.e. membership vector?

```
k$cluster
```

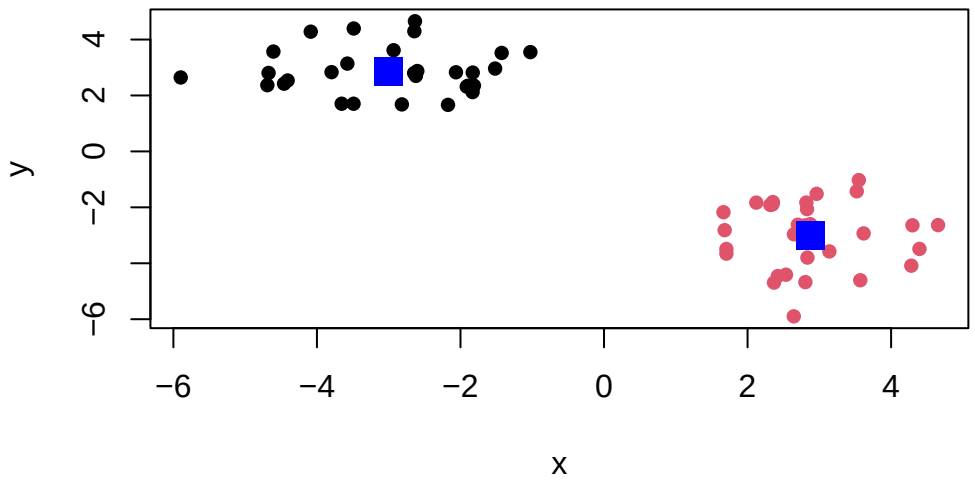
```
 [1] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 1 1 1 1 1 1
[39] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
```

```
k$centers
```

```
          x         y
1 -3.005755  2.872068
2  2.872068 -3.005755
```

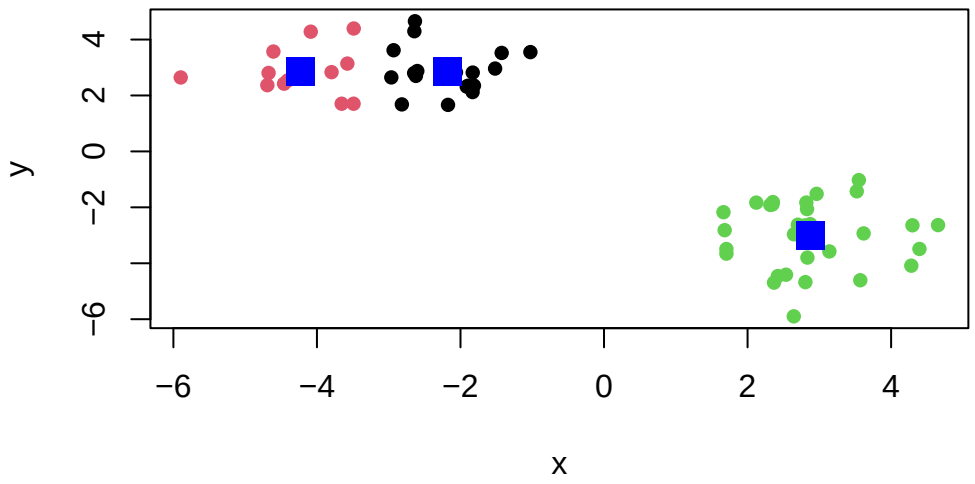
Q4. Make a plot of our data colored by clustering results with optionally the cluster centers shown

```
plot(x, col=k$cluster, pch=16)
points(k$centers, col="blue", pch=15, cex=2)
```



Q5. Run kmeans again but cluster into 3 groups and plot the results like we did above.

```
k3 <- kmeans(x, centers=3, nstart=20)
plot(x, col=k3$cluster, pch=16)
points(k3$centers, col="blue", pch=15, cex=2)
```



Hierarchial Clustering

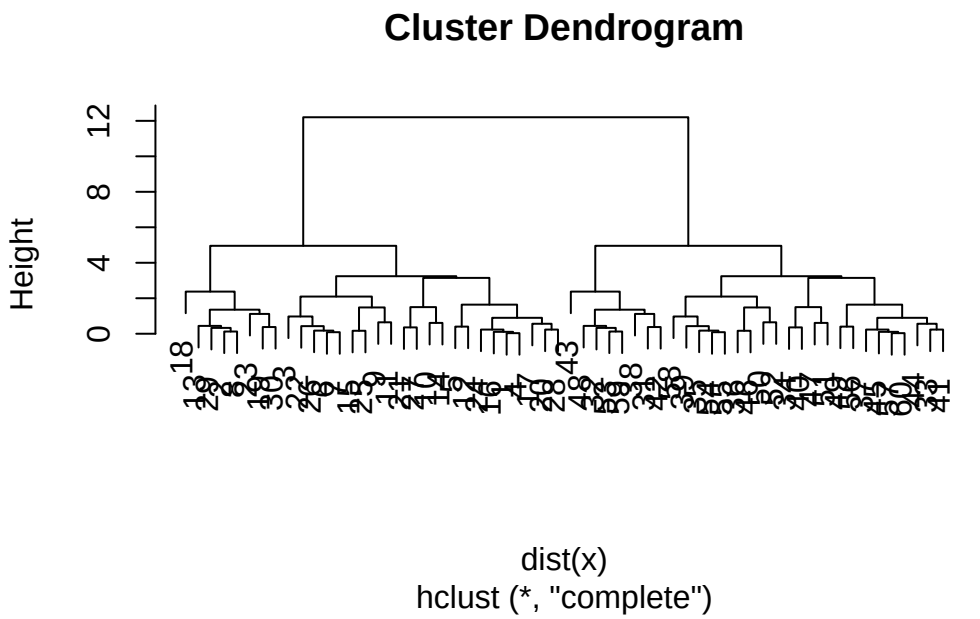
The main function in "base R" is called 'hclust()' It requires a distance matrix as input, not the raw data itself.

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

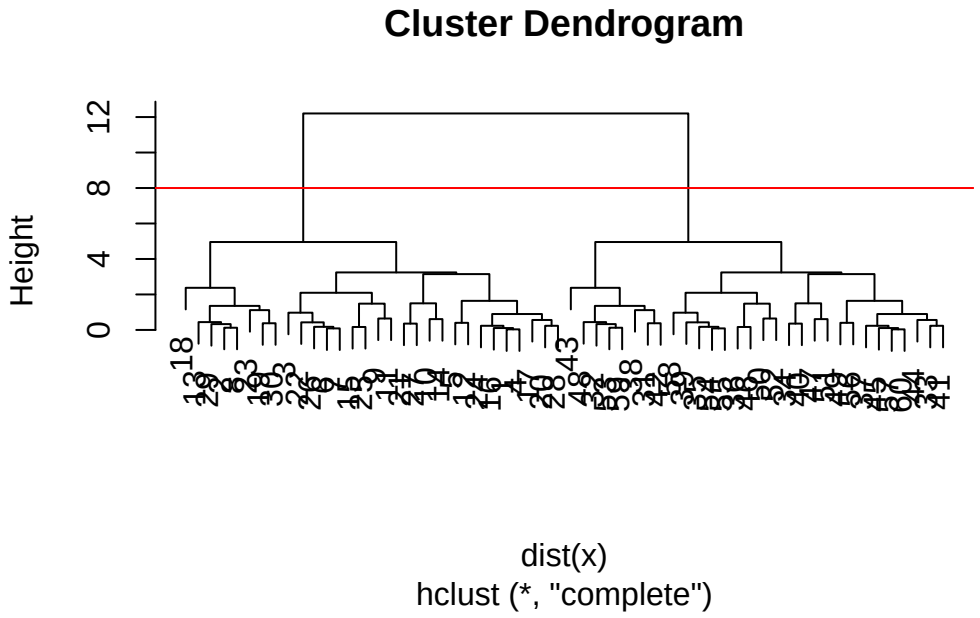
```
Call:
hclust(d = dist(x))

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

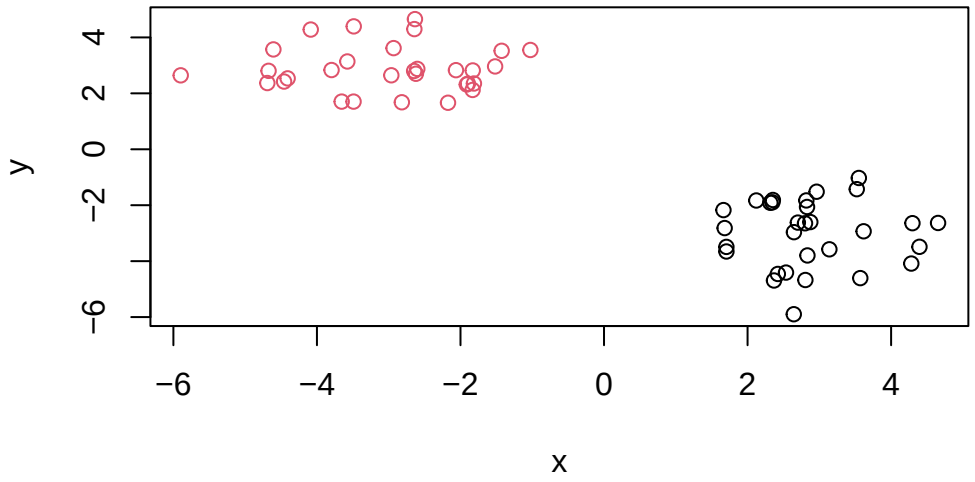
```
plot(hc)
```




```
plot(hc)
abline(h=8, col="red")
```



```
groups <- cutree(hc, h=8)
plot(x, col=groups)
```



Plot our hclust results in terms of our data colored by cluster membership

plot(hc, col=groups)

#Principal Component Analysis (PCA)

```
url <- "https://tinyurl.com/UK-foods"
x <- read.csv(url)
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
7      Fresh_potatoes      720   874      566      1033
8           Fresh_Veg      253   265      171       143
9           Other_Veg      488   570      418       355
10 Processed_potatoes      198   203      220       187
11      Processed_Veg      360   365      337       334
12        Fresh_fruit     1102  1137      957       674
13            Cereals     1472  1582     1462      1494
14          Beverages       57    73       53        47
15        Soft_drinks     1374  1256     1572      1506
16   Alcoholic_drinks      375   475      458       135
17      Confectionery       54    64       62        41
```

```
# rownames(x) <- x[,1]
# x <- x[,-1]
# head(x)
#If you run this code multiple times the dimensions will start to disappear.
```

```
nrow(x)
```

```
[1] 17
```

```
ncol(x)
```

```
[1] 5
```

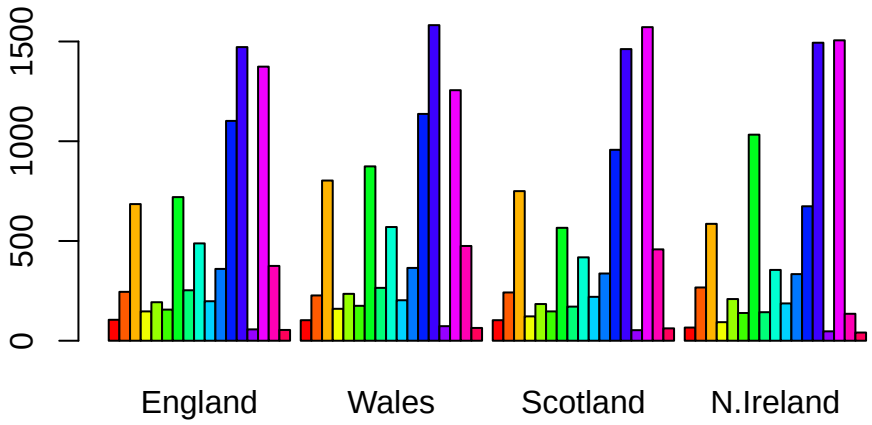
```
#Q1 There are 17 rows and 4 columns.
```

```
x<-read.csv(url, row.names=1)
head(x)
```

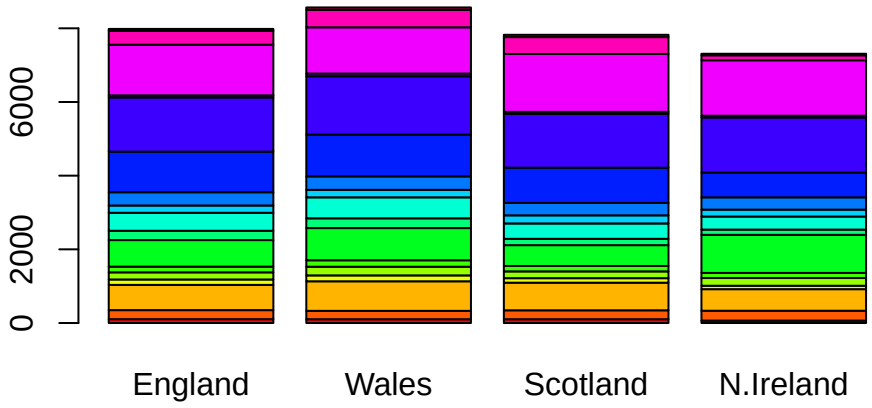
```
              England Wales Scotland N.Ireland
Cheese            105   103      103        66
Carcass_meat      245   227      242       267
Other_meat        685   803      750       586
Fish              147   160      122        93
Fats_and_oils     193   235      184       209
Sugars        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

```
# Q2 This approach is preferred and more robust. Repeated running of the other code leads
```

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

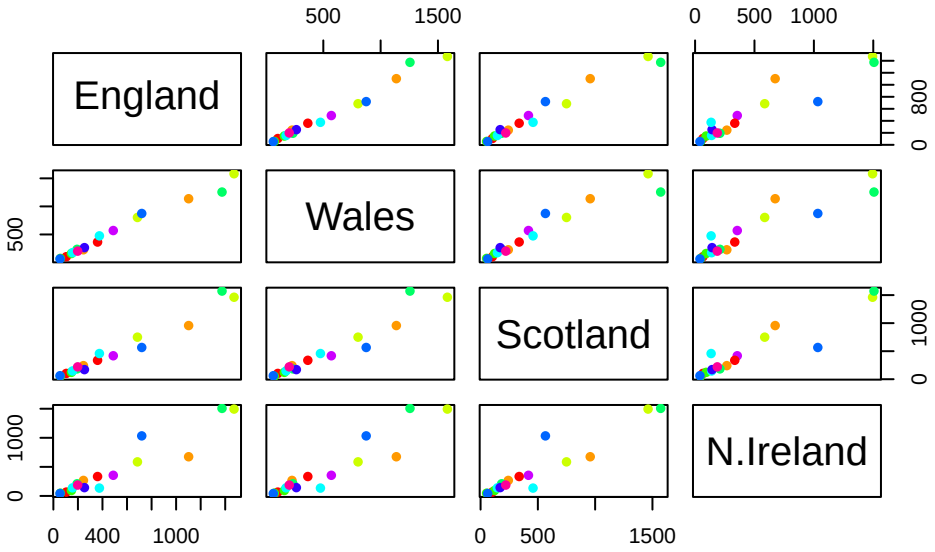



```
barplot(as.matrix(x), beside=F, col=rainbow(nrow(x)))
```




```
# Q3: changing the besides=T -> besides=F gives a stacked barplot
```

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




```
# Q5 This data compares food groups consumed between each possible pair of countries, ulti

# Q6 N. Ireland consumes significantly more fresh potatoes than the other countries.
```

The main function for PCA in base R si called 'prcomp()'

It wants the transpose of this data for analysis. 't()'

```
t(x)
```

```
           Cheese Carcass_meat  Other_meat  Fish Fats_and_oils  Sugars
England       105           245         685  147            193     156
Wales         103           227         803  160            235     175
Scotland      103           242         750  122            184     147
N.Ireland      66           267         586   93            209     139
           Fresh_potatoes  Fresh_Veg  Other_Veg  Processed_potatoes
England               720        253        488                 198
Wales                 874        265        570                 203
Scotland              566        171        418                 220
N.Ireland            1033        143        355                 187
           Processed_Veg  Fresh_fruit  Cereals  Beverages Soft_drinks
England              360         1102     1472         57         1374
```

```
Wales                365         1137     1582        73         1256
Scotland             337          957     1462        53         1572
N.Ireland            334          674     1494        47         1506
          Alcoholic_drinks  Confectionery
England                375             54
Wales                  475             64
Scotland               458             62
N.Ireland              135             41
```

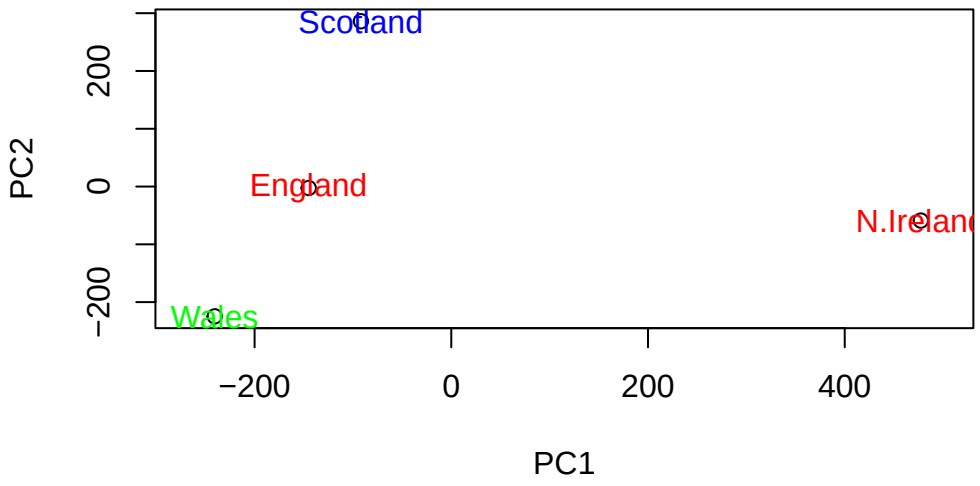
```r
pca <- prcomp(t(x))
summary(pca)
```

```
Importance of components:
                            PC1      PC2      PC3       PC4
Standard deviation     324.1502 212.7478 73.87622 3.176e-14
Proportion of Variance   0.6744   0.2905  0.03503 0.000e+00
Cumulative Proportion    0.6744   0.9650  1.00000 1.000e+00
```

One of the main results ppl look for is called the "scoreplot" a.k.a PC plot. PC1 vs PC2 plot

```r
# Q7 # Q8

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=rainbow(3))
```



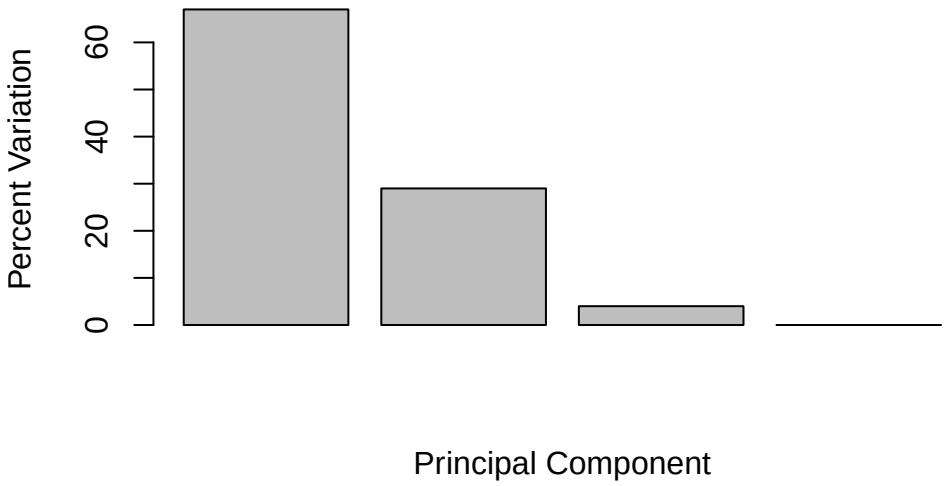

```
v <- round (pca$sdev^2/sum(pca$sdev^2)*100)
v
```

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

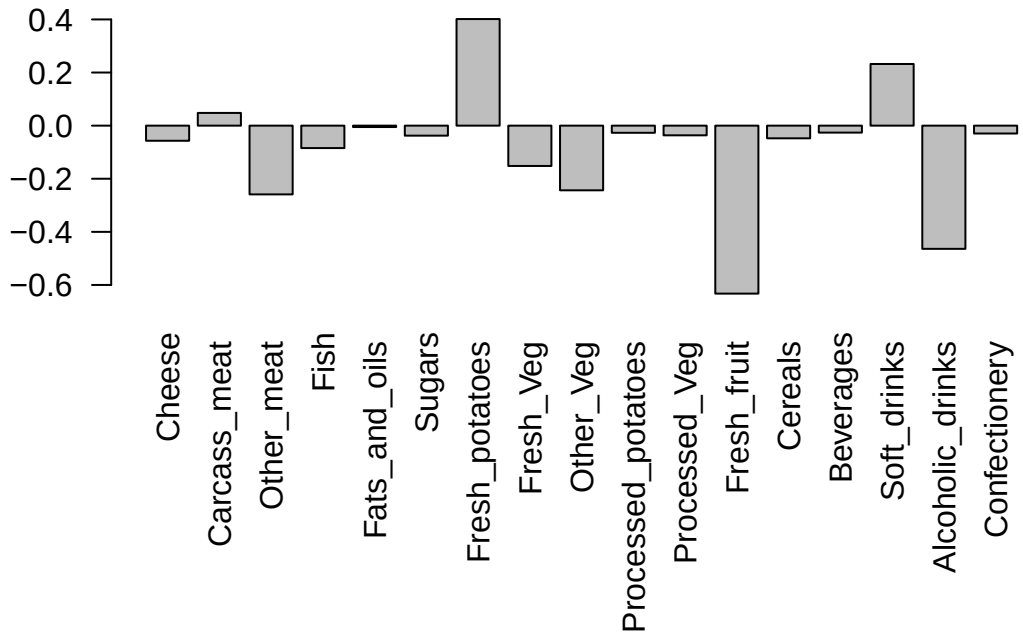
```
z <-summary(pca)
z$importance
```

```
                             PC1       PC2      PC3           PC4
Standard deviation     324.15019 212.74780 73.87622 3.175833e-14
Proportion of Variance   0.67444   0.29052  0.03503 0.000000e+00
Cumulative Proportion    0.67444   0.96497  1.00000 1.000000e+00
```

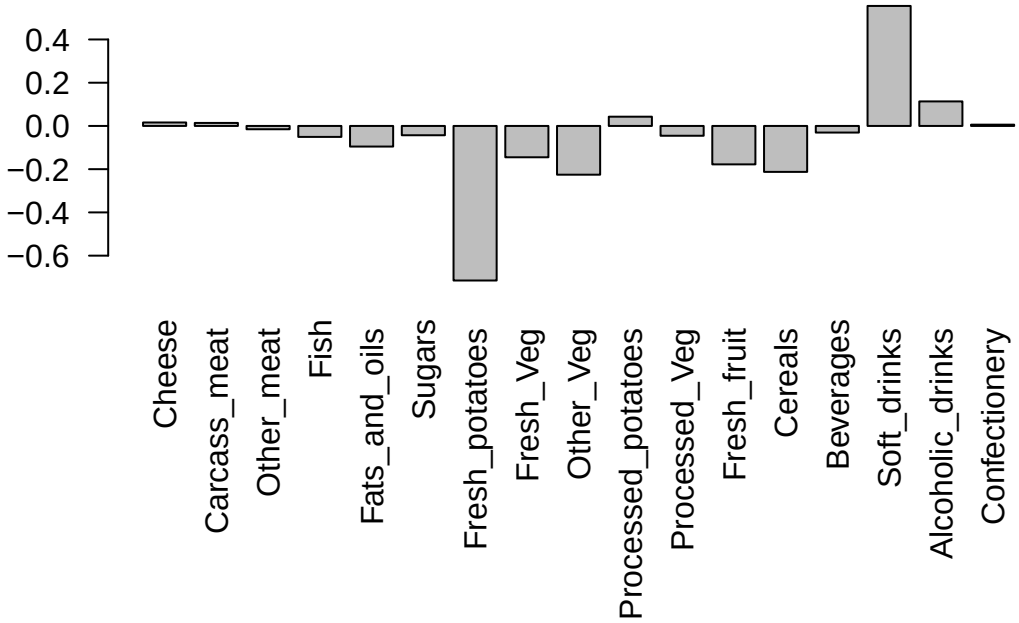
```
barplot(v, xlab="Principal Component", ylab="Percent Variation")
```




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



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




```
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
```

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

```
[1] 100
```

```r
ncol(rna.data)
```

```
[1] 10
```

```r
#Q10 There are 100 genes and 10 samples
```