ST464 - Assignment 1 - Solutions

Sean O'Riogain (18145426)

27 February 2019

```
knitr::opts_chunk$set(echo = TRUE)
getwd()
```

```
## [1] "C:/Users/oriogain/Dropbox/Maynooth/ST674 - Machine Learning/Assignments"
```

```
suppressPackageStartupMessages(library(tidyverse))
```

Question 3

The file eupop.txt contains the population and percentage distribution by age for EU countries in 1999. The age categories are 0-14 years, 15-44 years, 45-64 years and 65 years and over.

```
eupop <- read.table("eupop.txt", header=T, row.names=1)
eupop <- eupop[,-5]
```

a. Construct the euclidean distance matrix of the percentage variables.

```
d<-dist(eupop,method='euclidian')
head(d)
```

```
## [1] 2.389561 3.488553 4.631414 2.709243 2.229350 2.578759
```

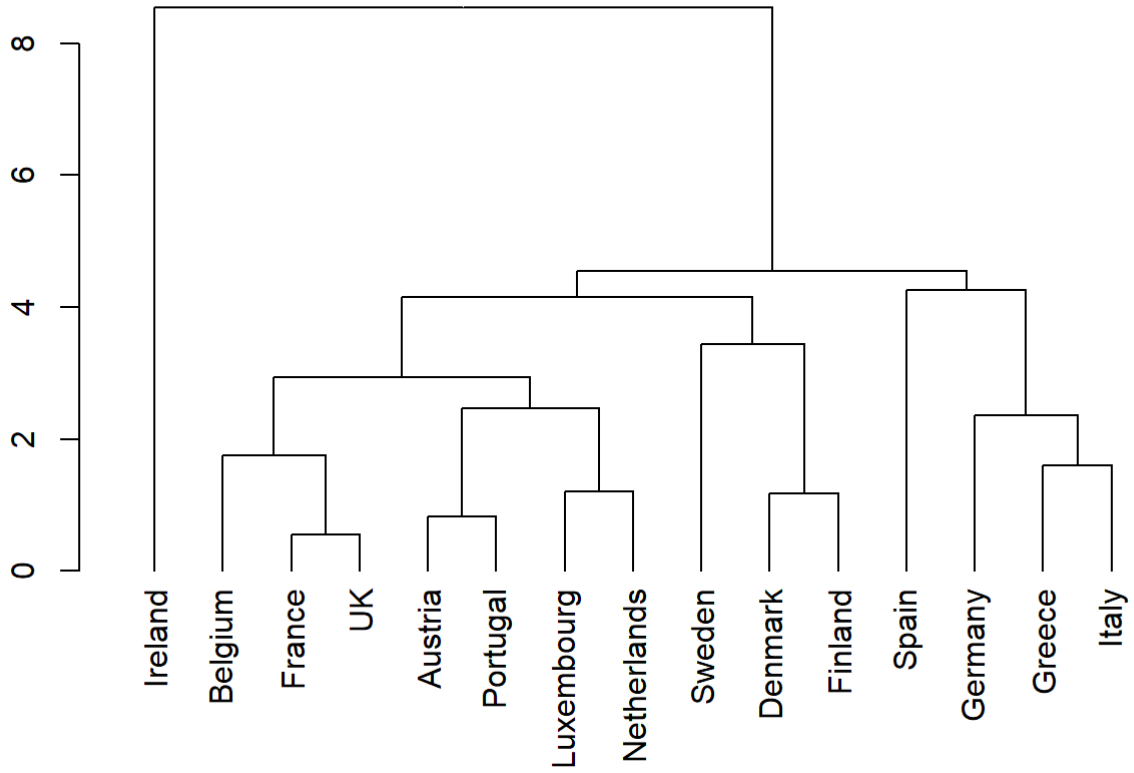
Use it to cluster the countries, using average linkage.

```
h<-hclust(d,method='average')
h
```

```
##
## Call:
## hclust(d = d, method = "average")
##
## Cluster method   : average
## Distance         : euclidean
## Number of objects: 15
```

Draw the dendrogram and interpret.

```
dg<-plot(as.dendrogram(h))
```




```
dg
```

```
## NULL
```

Looking at the above dendrogram from right to left, and in keeping with Tobler's 1st Law of geography, there seems to be a general grouping of countries geographically with some exceptions (in brackets):

1. **Southern (plus Germany, except Portugal);**
2. **Nordic;**
3. **Benelux (except Belgium);**
4. **Austria & Portugal (randomly);**
5. **Northwest (plus Belgium, except Ireland);**
6. **Ireland.**

Combining the above clusters a bit more we get:

1. **Southern (plus Germany, except Portugal);**
2. **Northern & Central (plus Portugal, except Germany and Ireland);**
3. **Ireland.**

Are there any outlier countries?

Ireland is a clear outlier, being the only country in a grouping (cluster) of its own.

Taking a closer look at the underlying data below, we can see that, for instance, Ireland is the only country to have a population percentage of greater than 20 in the 0-14 age group and less than 12 in the 65+ group….

```
rbind(eupop["Ireland",],head(eupop))
```

```
##            p014 p1544 p4564 p65.
## Ireland    22.2  46.2  20.3 11.3
## Austria    17.0  44.2  23.4 15.5
## Belgium    17.7  42.2  23.5 16.6
## Denmark    18.2  41.6  25.3 14.9
## Finland    18.4  40.8  26.1 14.7
## France     19.0  42.5  22.8 15.8
## Luxembourg 18.8  43.7  23.2 14.3
```

```
subset(eupop,p014>20)
```

```
##         p014 p1544 p4564 p65.
## Ireland 22.2  46.2  20.3 11.3
```

```
subset(eupop,p65.<12)
```

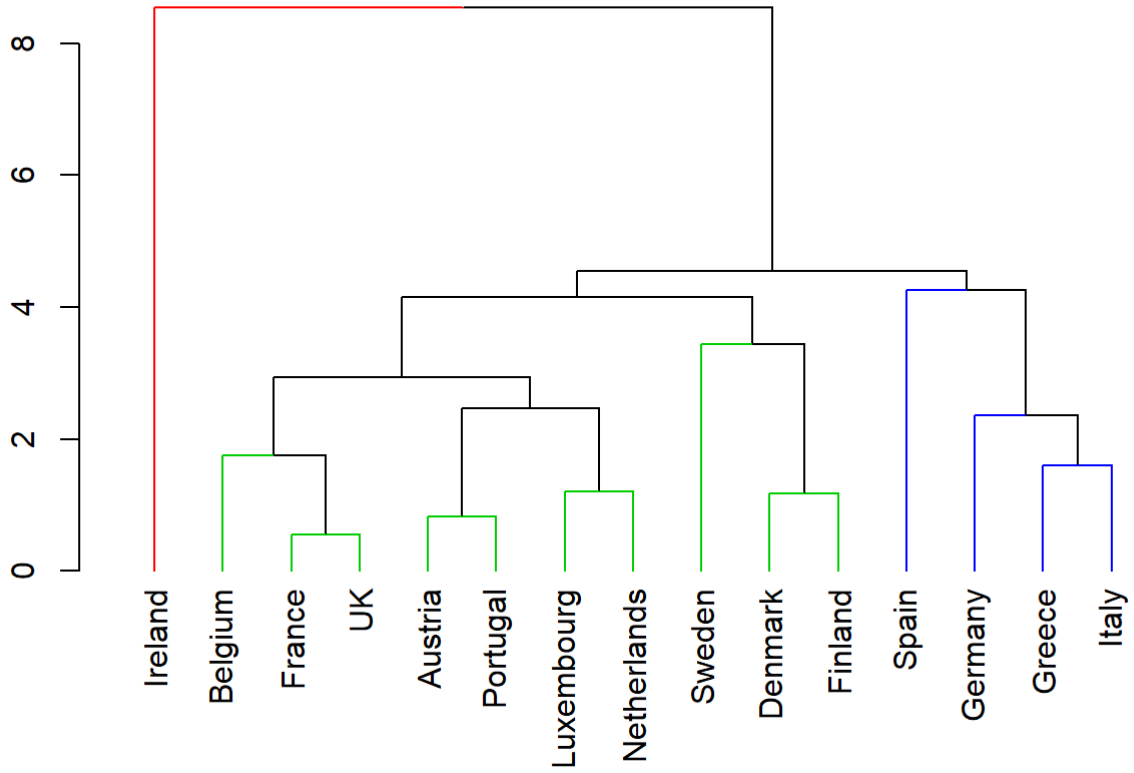
```
##         p014 p1544 p4564 p65.
## Ireland 22.2  46.2  20.3 11.3
```

b. Examine the 3-cluster solution.

```
suppressPackageStartupMessages(library(dendextend))
```

```
## Warning: package 'dendextend' was built under R version 3.5.2
```

```
col<-cutree(h, 3, order_clusters_as_data=F)
plot(color_branches(h, col=col+1))
```



Which countries belong to each of the three clusters?

```
col
```

```
##     Ireland     Belgium      France          UK     Austria    Portugal
##           1           2           2           2           2           2
##  Luxembourg Netherlands      Sweden     Denmark     Finland       Spain
##           2           2           2           2           2           3
##     Germany      Greece       Italy
##           3           3           3
```

Summarise the partitions with sumPartition (in h1code.R)

```
sumPartition(eupop, col)
```

```
## Final Partition
##
## Number of clusters  3
##
##           N.obs Within.clus.SS Ave.dist..Centroid Max.dist.centroid
## Cluster 1     1          0.000           0.000000          0.000000
## Cluster 2    10         59.290           2.314504          3.831579
## Cluster 3     4         85.225           4.136300          7.468517
##
##
## Cluster centroids
##
##       Cluster 1 Cluster 2 Cluster 3 Grand centrd
## p014  17        18.11     16.85     17.7
## p1544 44.2      42.35     44.7      43.1
## p4564 23.4      24.18     22.875    23.78
## p65.  15.5      15.4      15.575    15.45333
##
##
## Distances between Cluster centroids
##
##           Cluster 1 Cluster 2 Cluster 3
## Cluster 1 0.0000000  2.296301 0.7441438
## Cluster 2 2.2963014  0.000000 2.9738443
## Cluster 3 0.7441438  2.973844 0.0000000
```

Interpret your findings.

The first section (of 3) of the report above shows us that the data points in Cluster 2 are about two times more closely spaced than those in Cluster 3 (as per the ratio of their average and maximum distances from their respective centroids). As Cluster 1 contains only 1 data point which, therefore, is its centroid, whose its distance from itself is zero.

The third (and final) section tells us that Cluster 2 is (almost) equidistant from both Cluster 1 and Cluster 3, while Cluster 1 and Cluster 3 are in relatively closer proximity to each other.

c. Use the kmeans algorithm to find another 3-cluster grouping of countries.

```
km<-kmeans(eupop, 3, nstart=100)
str(km)
```

```
## List of 9
##  $ cluster     : Named int [1:15] 1 3 3 3 3 3 3 1 3 3 ...
##   ..- attr(*, "names")= chr [1:15] "Austria" "Belgium" "Denmark" "Finland" ...
##  $ centers     : num [1:3, 1:4] 15.8 22.2 18.5 44 46.2 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:3] "1" "2" "3"
##   .. ..$ : chr [1:4] "p014" "p1544" "p4564" "p65."
##  $ totss       : num 172
##  $ withinss    : num [1:3] 26.4 0 39.4
##  $ tot.withinss: num 65.8
##  $ betweenss   : num 106
##  $ size        : int [1:3] 6 1 8
##  $ iter        : int 2
##  $ ifault      : int 0
##  - attr(*, "class")= chr "kmeans"
```

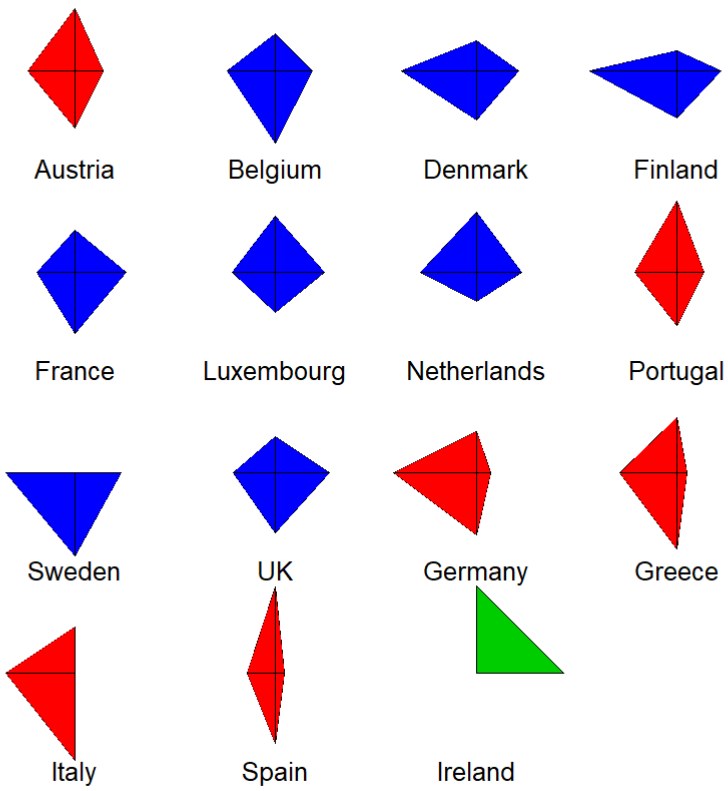
Which countries belong to each of the three clusters?

```
sort(km$cluster)
```

```
##     Austria    Portugal     Germany      Greece       Italy       Spain
##           1           1           1           1           1           1
##     Ireland     Belgium     Denmark     Finland      France  Luxembourg
##           2           3           3           3           3           3
## Netherlands      Sweden          UK
##           3           3           3
```

d. Construct a stars plot which shows the data and clustering obtained from kmeans.

```
stars(eupop, col.stars=km$cluster+1)
```



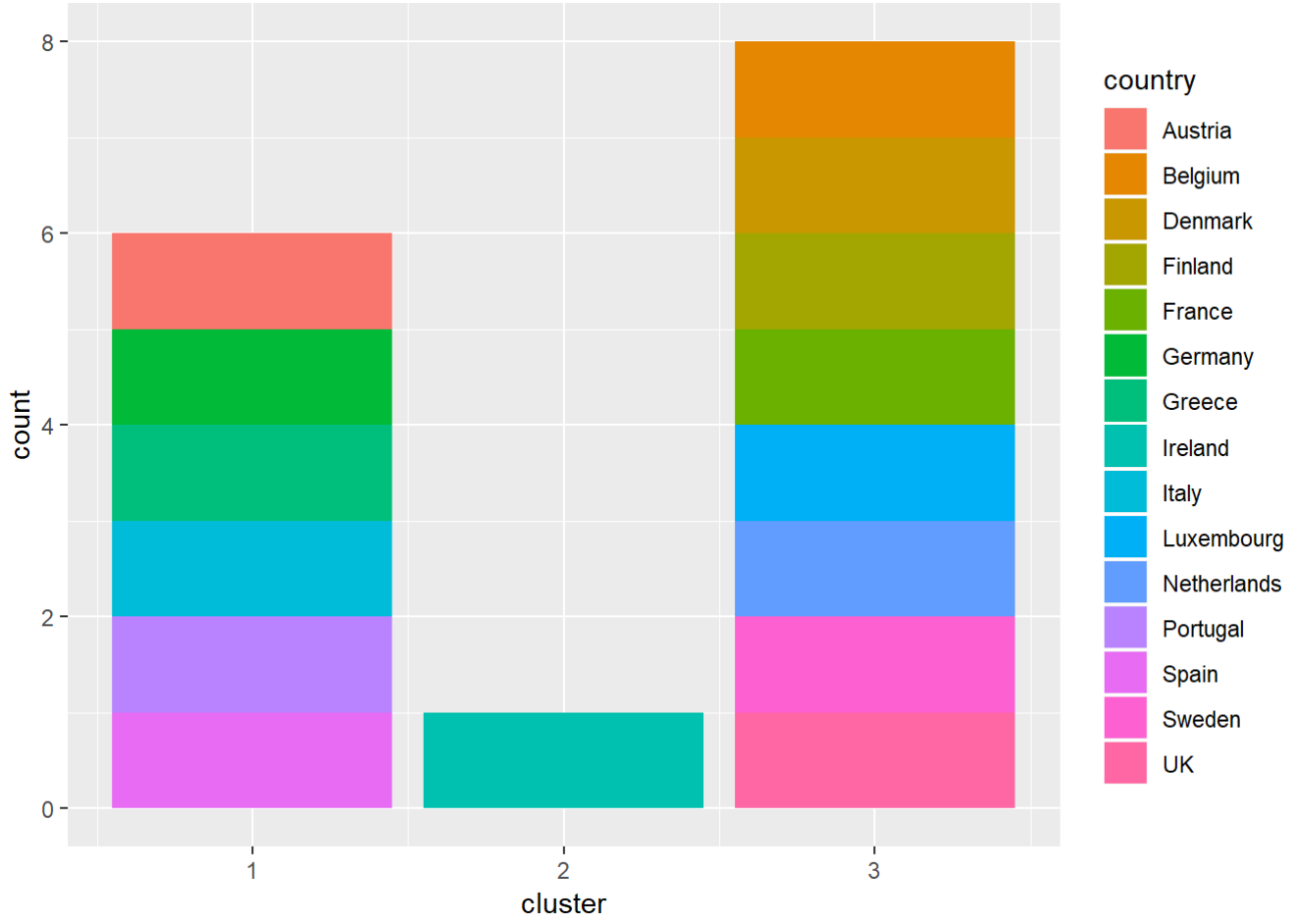
Optional:

Can you think of a better way of showing the clusters?

```
country<-rownames(as.matrix(km$cluster))
cluster<-km$cluster
df<-data.frame(country,cluster,eupop)
head(df)
```

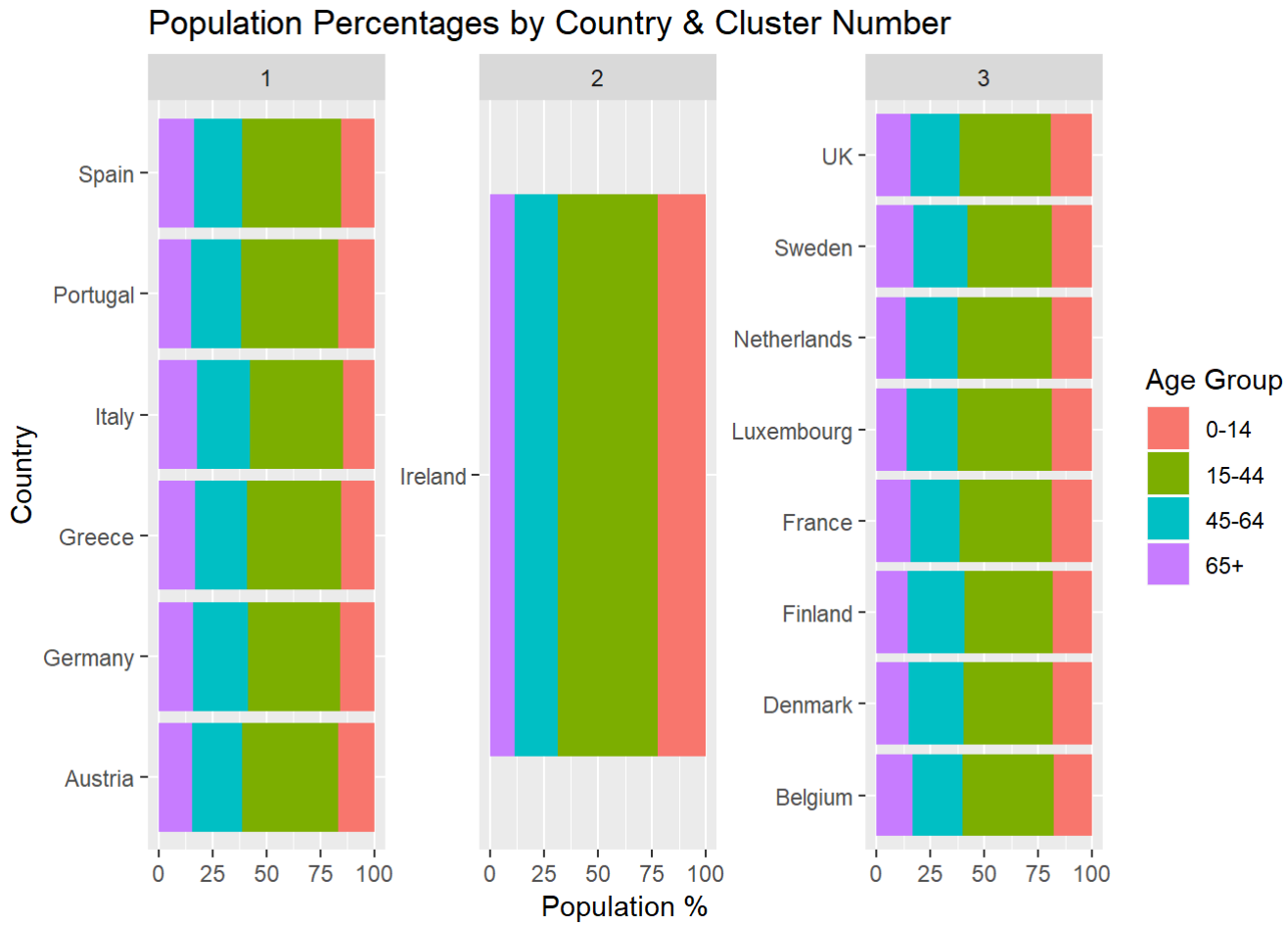
```
##              country cluster p014 p1544 p4564 p65.
## Austria       Austria       1 17.0  44.2  23.4 15.5
## Belgium       Belgium       3 17.7  42.2  23.5 16.6
## Denmark       Denmark       3 18.2  41.6  25.3 14.9
## Finland       Finland       3 18.4  40.8  26.1 14.7
## France         France       3 19.0  42.5  22.8 15.8
## Luxembourg Luxembourg       3 18.8  43.7  23.2 14.3
```

```
ggplot(data=df, aes(x=cluster, fill=country)) +
geom_bar()
```



Can you think of a way to present the data and the clustering results of both methods on the same graphical display?

```
dat<-gather(df, key=age_grp, value=pct_pop, p014, p1544, p4564, p65.,
            factor_key=T)

ggplot(dat, aes(x=country, y=pct_pop,
                fill=age_grp)) +
  geom_bar(stat="identity") +
  xlab("Country")+ylab("Population %") +
  scale_fill_discrete(name = "Age Group",
                      labels=c("0-14", "15-44", "45-64", "65+")) +
  facet_wrap(~cluster, scales="free") +
  ggtitle("Population Percentages by Country & Cluster Number") +
  coord_flip()
```



This does the required job for the results of the kmeans method. The results of the hclust method could be combined with them, having manipulated (stacked) them in a similar way - plus adding 3 to their cluster numbers. The titles of the facet panels would then also need to be manipulated (using facet_wrap's labeller parameter) to display suitable text - e.g 'Cluster 1 (kmeans)' for the first, 'Cluster 1 (hclust)' for the fourth, etc. nrow and column parameter settings could be used to display both sets of facet panels on separate rows, if required.

Question 4

Music data from class.

```
music<-read.csv("music.csv")
str(music)
```

```
## 'data.frame':    62 obs. of  8 variables:
##  $ X     : Factor w/ 62 levels "1","2","3","4",..: 18 25 45 36 26 44 23 48 37 43 ...
##  $ Artist: Factor w/ 7 levels "Abba","Beatles",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ Type  : Factor w/ 3 levels "Classical","New wave",..: 3 3 3 3 3 3 3 3 3 3 ...
##  $ LVar  : num  17600756 9543021 9049482 7557437 6282286 ...
##  $ LAve  : num  -90 -75.8 -98.1 -90.5 -89 ...
##  $ LMax  : int  29921 27626 26372 28898 27940 25531 14699 8928 22962 15517 ...
##  $ LFEner: num  106 103 102 102 100 ...
##  $ LFreq : num  59.6 58.5 124.6 48.8 74 ...
```

a. Run the k-means algorithm over the range k = 1,….,15 clusters and record the total within cluster sum of squares (TWSS). Let nstart = 25.

```
music.feat<-music[, 4:8]
music.feat<-scale(music.feat)
glimpse(music.feat)
```

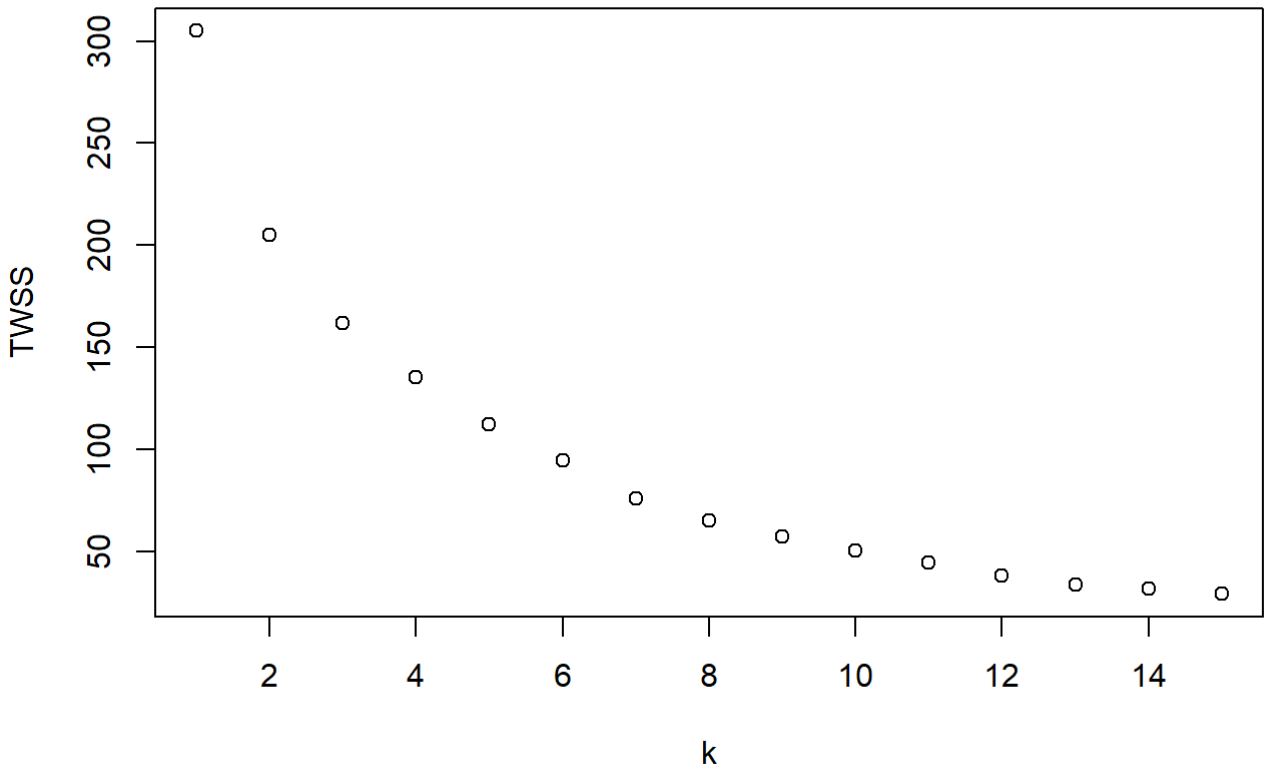
```
##  num [1:62, 1:5] -0.089 -0.394 -0.413 -0.469 -0.517 ...
##  - attr(*, "dimnames")=List of 2
##   ..$ : NULL
##   ..$ : chr [1:5] "LVar" "LAve" "LMax" "LFEner" ...
##  - attr(*, "scaled:center")= Named num [1:5] 2.00e+07 -7.81 2.25e+04 1.04e+02 2.31e+02
##   ..- attr(*, "names")= chr [1:5] "LVar" "LAve" "LMax" "LFEner" ...
##  - attr(*, "scaled:scale")= Named num [1:5] 2.64e+07 4.72e+01 8.76e+03 5.48 1.77e+02
##   ..- attr(*, "names")= chr [1:5] "LVar" "LAve" "LMax" "LFEner" ...
```

```
twss<-rep(0, 15)
kval<-c(1:15)

for(k in c(1:15)){
  km<-kmeans(music.feat, k, nstart=25)
  kval[k]<-k
  twss[k]<-km$tot.withinss
}

df<-data.frame(kval, twss)
head(df)
```

```
##   kval      twss
## 1    1 305.00000
## 2    2 205.17961
## 3    3 161.78557
## 4    4 135.09697
## 5    5 112.07876
## 6    6  94.42776
```

Plot k versus TWSS….

```
plot(x=df$kval, y=df$twss, xlab="k", ylab="TWSS")
```



Please note that I've chosen to plot TWSS versus k (instead of k versus TWSS), as TWSS is the response variable.

….and choose the best fitting number of clusters.

From the plot above, it looks like a k value of 6 would be a reasonable choice - on the basis that at that point, just over 1/3 of the way through the k-value range achieved just over a 2/3 reduction (305->94) in the TWSS value.

What do you observe?

As expected the value of TWSS reduces (non-linearly - almost quadratically, in this case) as that of k increases.

In this case, as the resultant curve is a relatively gradual, the position of an elbow isn't that clear-cut (unlike a similiar graph that we've seen in class).

b. Make a table of artist vs cluster solution from k = 5.

```
km<-kmeans(music.feat, 5, nstart=25)
str(km)
```

```
## List of 9
##  $ cluster     : int [1:62] 3 3 3 3 3 3 3 4 3 3 ...
##  $ centers     : num [1:5, 1:5] 1.521 -0.559 -0.322 -0.696 -0.228 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:5] "1" "2" "3" "4" ...
##   .. ..$ : chr [1:5] "LVar" "LAve" "LMax" "LFEner" ...
##  $ totss       : num 305
##  $ withinss    : num [1:5] 21.1 48.7 28.8 13.5 0
##  $ tot.withinss: num 112
##  $ betweenss   : num 193
##  $ size        : int [1:5] 15 17 19 10 1
##  $ iter        : int 3
##  $ ifault      : int 0
##  - attr(*, "class")= chr "kmeans"
```

```
tab<-cbind(km$cluster,as.character(music$Artist))
head(tab)
```

```
##      [,1] [,2]
## [1,] "3"  "Abba"
## [2,] "3"  "Abba"
## [3,] "3"  "Abba"
## [4,] "3"  "Abba"
## [5,] "3"  "Abba"
## [6,] "3"  "Abba"
```

```
table(tab[,2], tab[, 1], useNA="ifany")
```

```
##
##             1 2 3 4 5
##   Abba      0 0 9 1 0
##   Beatles   8 0 2 0 0
##   Beethoven 0 5 2 1 0
##   Eels      7 0 3 0 0
##   Enya      0 0 1 2 0
##   Mozart    0 6 0 0 0
##   Vivaldi   0 5 1 3 1
##   <NA>      0 1 1 3 0
```

Question 5

Protein data. We want to study the similarities and differences in the protein composition of the diets of different countries.

Read in the data and extract the feature variables and scale them.

```
# Read in the Protein data
protein<-read.csv("protein.csv")
str(protein)
```

```
## 'data.frame':    25 obs. of  10 variables:
##  $ Country  : Factor w/ 25 levels "Albania","Austria",..: 1 2 3 4 5 6 7 8 9 10 ...
##  $ RedMeat  : num  10.1 8.9 13.5 7.8 9.7 10.6 8.4 9.5 18 10.2 ...
##  $ WhiteMeat: num  1.4 14 9.3 6 11.4 10.8 11.6 4.9 9.9 3 ...
##  $ Eggs     : num  0.5 4.3 4.1 1.6 2.8 3.7 3.7 2.7 3.3 2.8 ...
##  $ Milk     : num  8.9 19.9 17.5 8.3 12.5 25 11.1 33.7 19.5 17.6 ...
##  $ Fish     : num  0.2 2.1 4.5 1.2 2 9.9 5.4 5.8 5.7 5.9 ...
##  $ Cereals  : num  42.3 28 26.6 56.7 34.3 21.9 24.6 26.3 28.1 41.7 ...
##  $ Starch   : num  0.6 3.6 5.7 1.1 5 4.8 6.5 5.1 4.8 2.2 ...
##  $ Nuts     : num  5.5 1.3 2.1 3.7 1.1 0.7 0.8 1 2.4 7.8 ...
##  $ Fr.Veg   : num  1.7 4.3 4 4.2 4 2.4 3.6 1.4 6.5 6.5 ...
```

```
# Extract only the feature variables
prot.feat<-protein[,-1]

# Standardise to feature values to values in the range 0 to 10
scale_0_10 <- function(x){((x-min(x))/(max(x)-min(x)))*10}
prot.feat<-apply(prot.feat,2,scale_0_10)
```

Using any methods that you choose from this course or otherwise, write a brief summary.

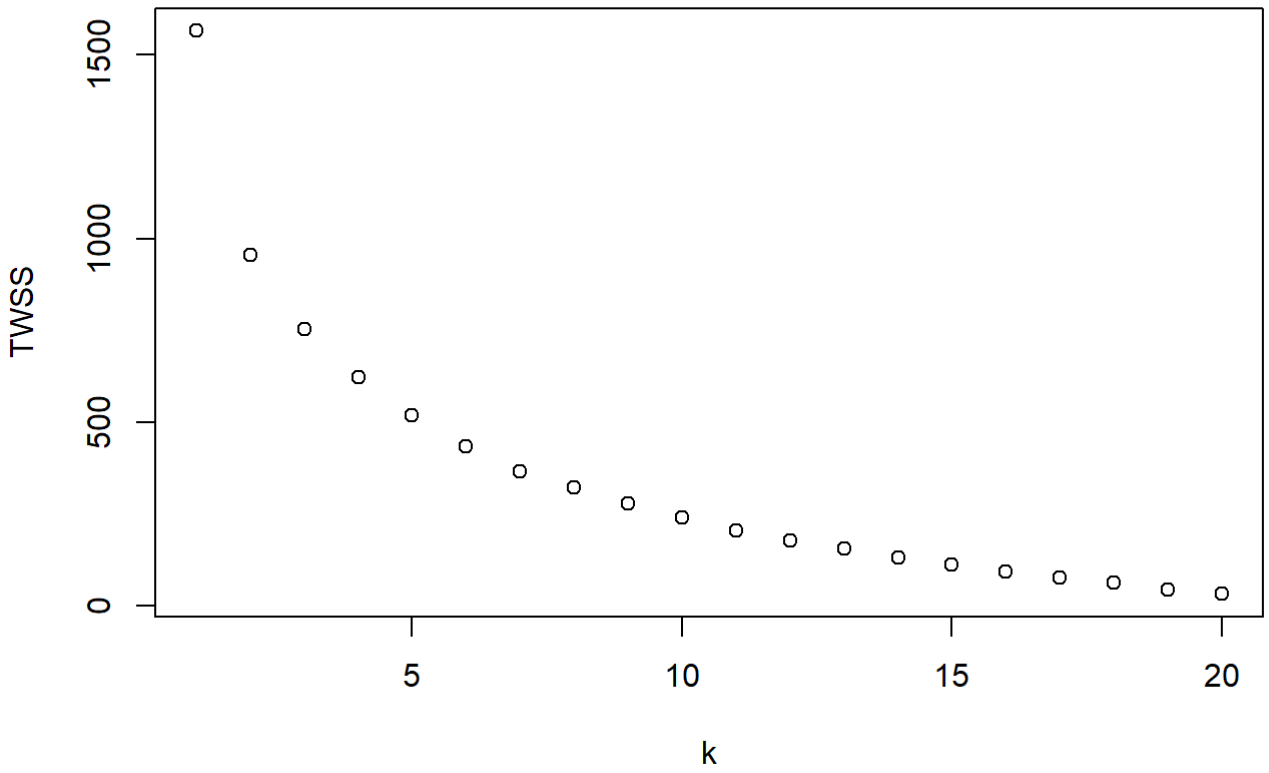
Let's use kmeans clustering to identify groups of similar diets and compare them to identify the difference between them.

The first step is to find the optimum number of clusters (k) to use.

```
# Find the optimum value of k to use

# Create the data frame for plotting
kval<-rep(0, 20)
twss<-rep(0, 20)
df<-data.frame(kval, twss)

# Populate the data frame
for(k in c(1:20)){
  km<-kmeans(prot.feat, k, nstart=100)
  df$kval[k]<-k
  df$twss[k]<-km$tot.withinss
}

head(df)
```

```
##   kval      twss
## 1    1 1565.2537
## 2    2  956.0524
## 3    3  754.9216
## 4    4  622.2614
## 5    5  519.7109
## 6    6  433.9588
```

```
# Plot twss versus k
plot(df$kval, df$twss, xlab="k", ylab="TWSS")
```



Looking at the plot above, 4 looks like a reasonable choice of k, where TWSS is reduced by 60% after only 25% of values of k assessed.

With only 25 observations in the Protein dataset, it won't make sense to have a large number of clusters anyway.

Now to create those 4 clusters….

```
# Create the kmeans cluster data
km<-kmeans(prot.feat, 4, nstart=100)

prot<-data.frame(km$cluster, protein$Country, prot.feat, stringsAsFactors=T)
colnames(prot)[1:2]<-c("Cluster", "Country")

sort(km$cluster)
```

```
##  [1] 1 1 1 1 1 1 2 2 2 2 3 3 3 3 4 4 4 4 4 4 4 4 4 4 4
```

In the table above, we can see that the countries fall into the following geographical/cultural (and political and economical) groupings:

1. **Nordic (x 4, EU/EEA-Capitalist-Tier 1);**
2. **Mediterranean (x 4, EU/EEA-Capitalist-Tier 2);**
3. **Slavic (x 6, Soviet Bloc-Communist-Tier 3);**
4. **Other (x 11, Various).**

- **Please note that these groupings are not necessarily listed in order of cluster number order (as that can vary from one execution to the next).**

As USSR, West Germany, East Germany and Yugoslavia are included in the 25 countries in the dataset, in all probability it dates from either before, or shortly after, the fall of the Soviet Union.

Note that the wealth tiering (from wealthiest to poorest) is a crude, anectotal indicator of my own creation.

Of course, at first glance, at least, this clustering seems to make sense because the diet of a particular country will depend on its geopgraphic location (e.g. more fish consumed if it has a coastline).

Geoography will also tend to influence the cultural, political and economic links between nations. There is also a link between political system (capitalism/democracy versus communism/dictatorship) and economic success.

This, in turn, will also influence the diet of a countries inhabitants, in that more of the expensive forms of protein (e.g. red meat) will tend to be consumed in the wealthier countries.

Let's see if increasing the number of clusters by 1 (k = 5) will split the 4th cluster above in some meaningful way…..

```
# Create the kmeans cluster data
km<-kmeans(prot.feat, 5, nstart=100)

# Combine the cluster numner and country with the protein feature data
prot<-data.frame(km$cluster, protein$Country, prot.feat, stringsAsFactors=T)
colnames(prot)[1:2]<-c("Cluster", "Country")

# Sort the countries by cluster
sort(km$cluster)
```

```
##  [1] 1 1 1 1 2 2 2 2 2 3 3 3 3 4 4 4 4 5 5 5 5 5 5 5 5
```

In the table above, we can now see an improved grouping of the countries into the following clusters:

1. **Nordic (x 4, EU/EEA-Capitalist-Tier 1);**
2. **Western (x 8, Eu/EEA-Capitalise-Tier 1);**
3. **Slavic (x 5, Soviet Bloc-Communist-Tier 3);**
4. **Balkan (x4, Soviet Bloc-Communist-Tier 4);**
5. **Mediterranean (x 4, EU/EEA-Capitalist-Tier 2).**

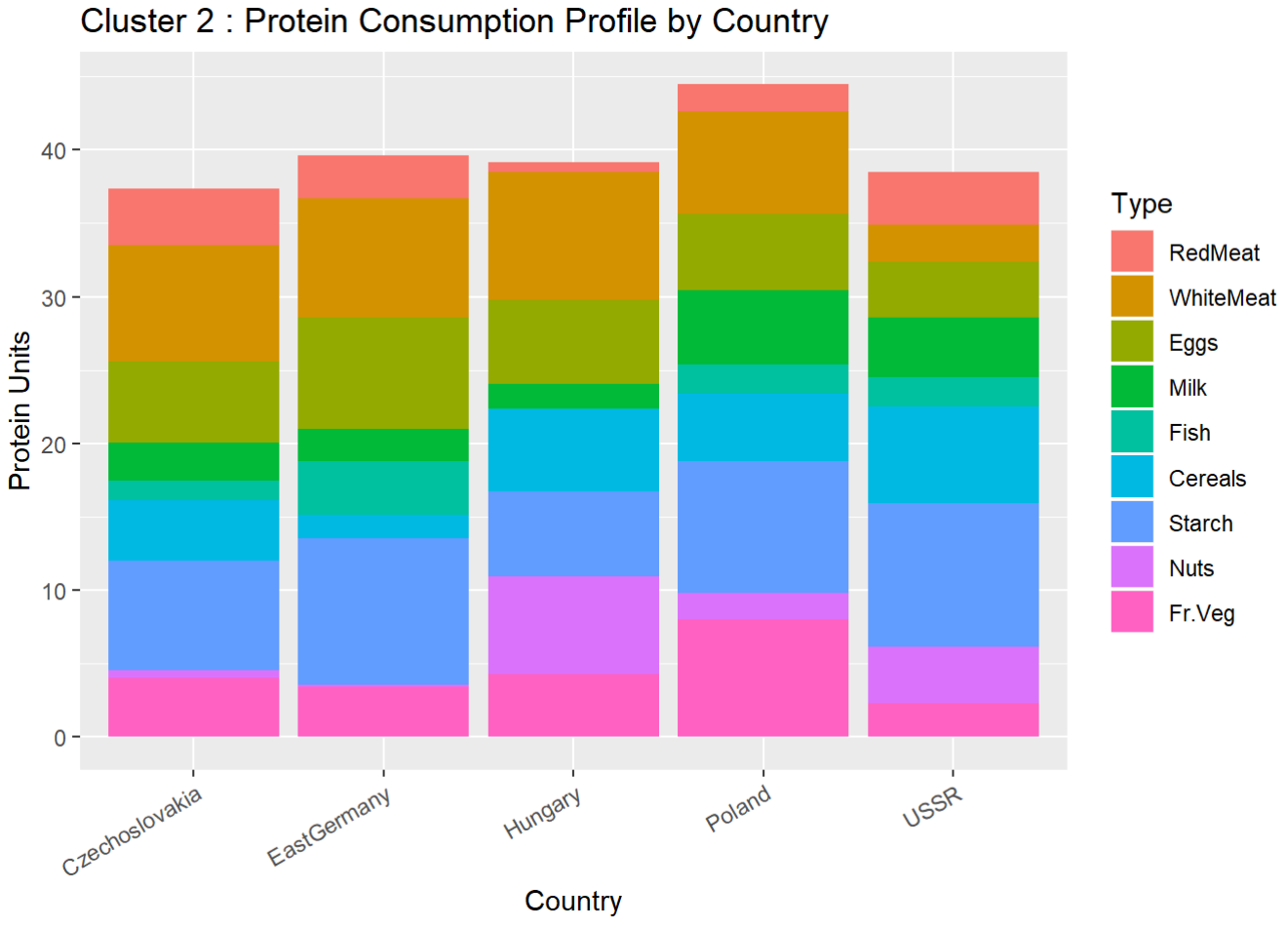
Please note that:

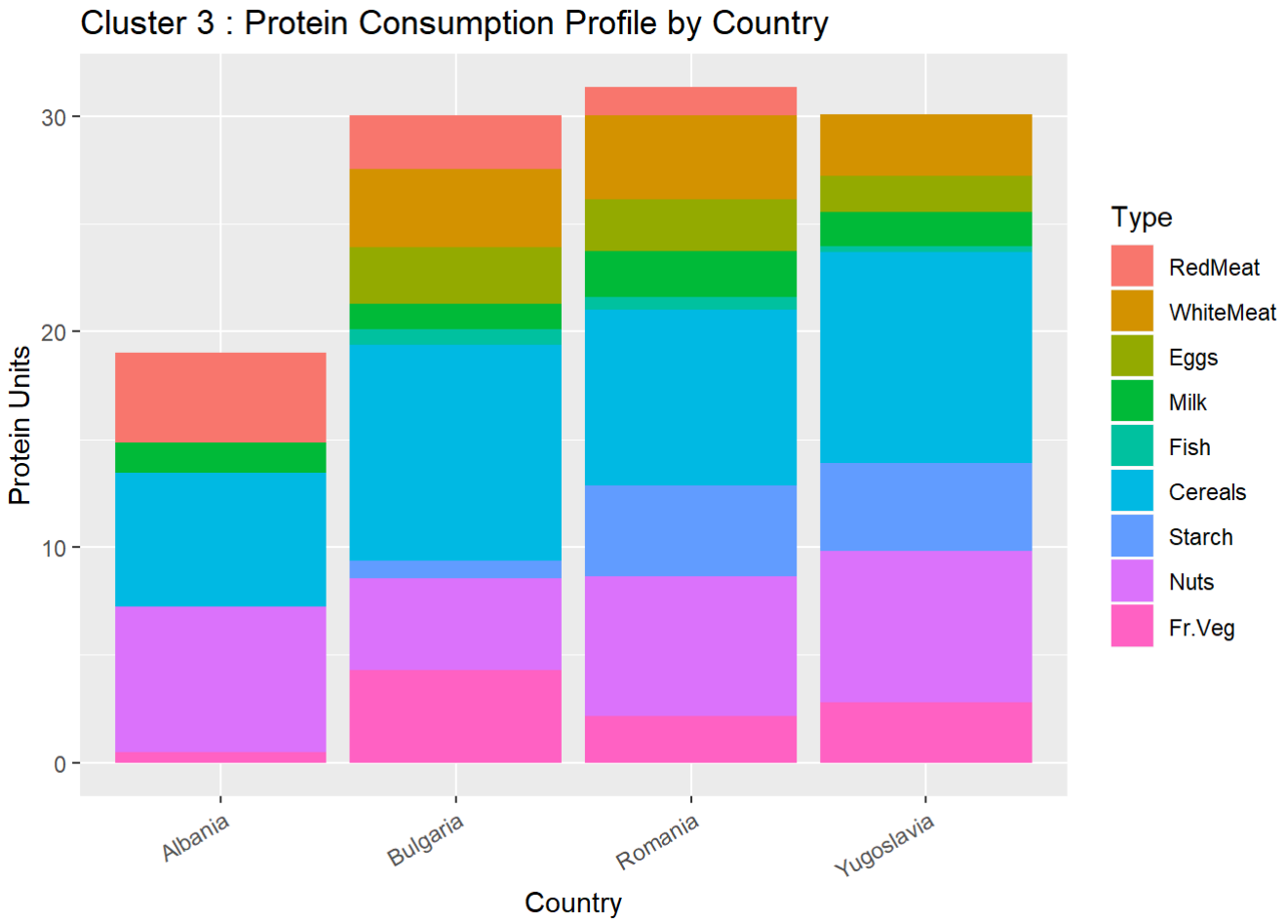
- **These groupings are not necessarily listed in order of cluster number order (as that can vary from one execution to the next).**
- **The cultural labels Slavic and Balkan are used in a very loose way, in that, for instance, Hungarians are not Slavs and Bulgaria is not in the Balkan Peninsula (although it does include the Balkan mountains).**

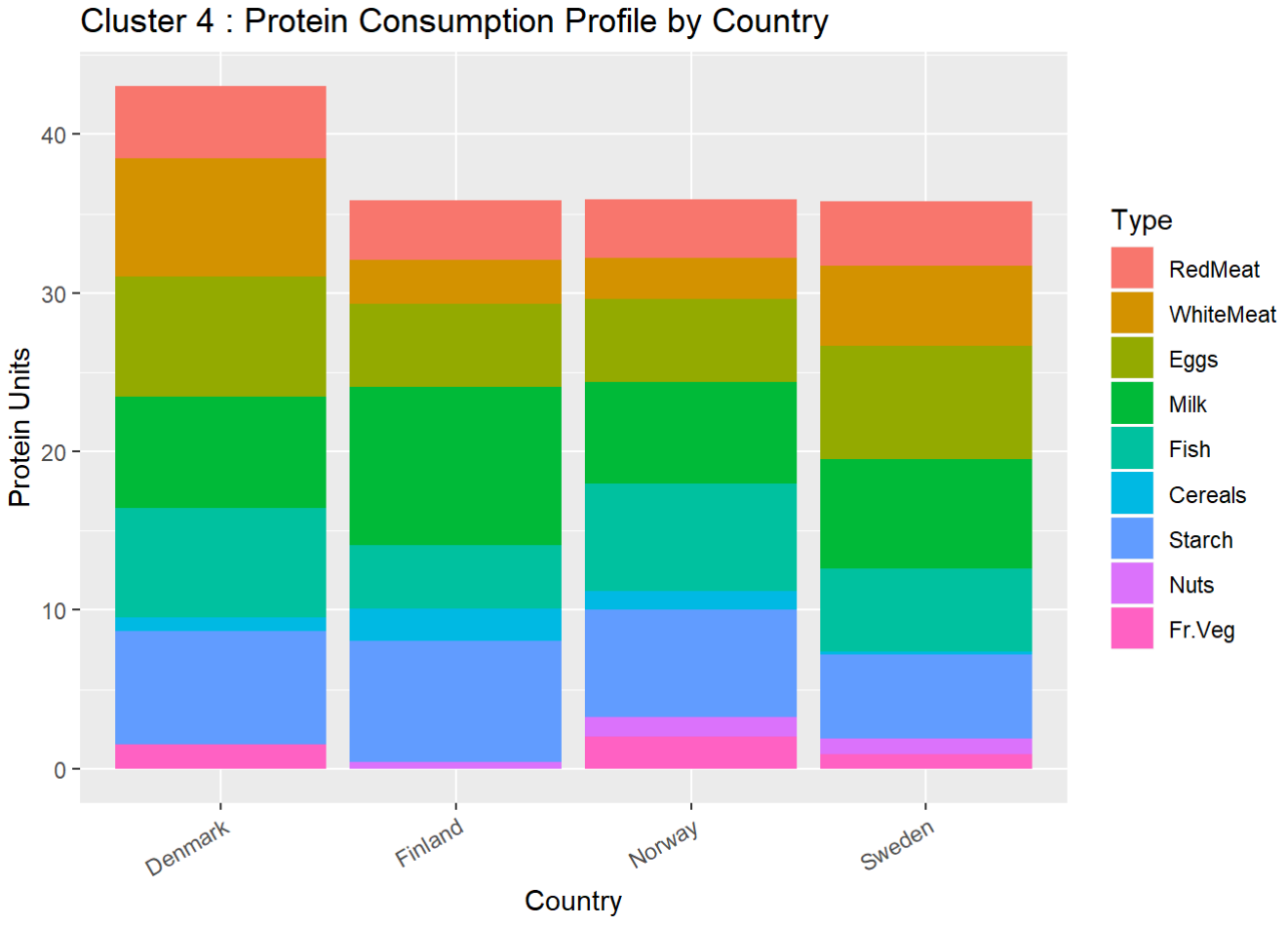
Now, let's use the Protein data to compare the dietary profiles of those 5 clusters……

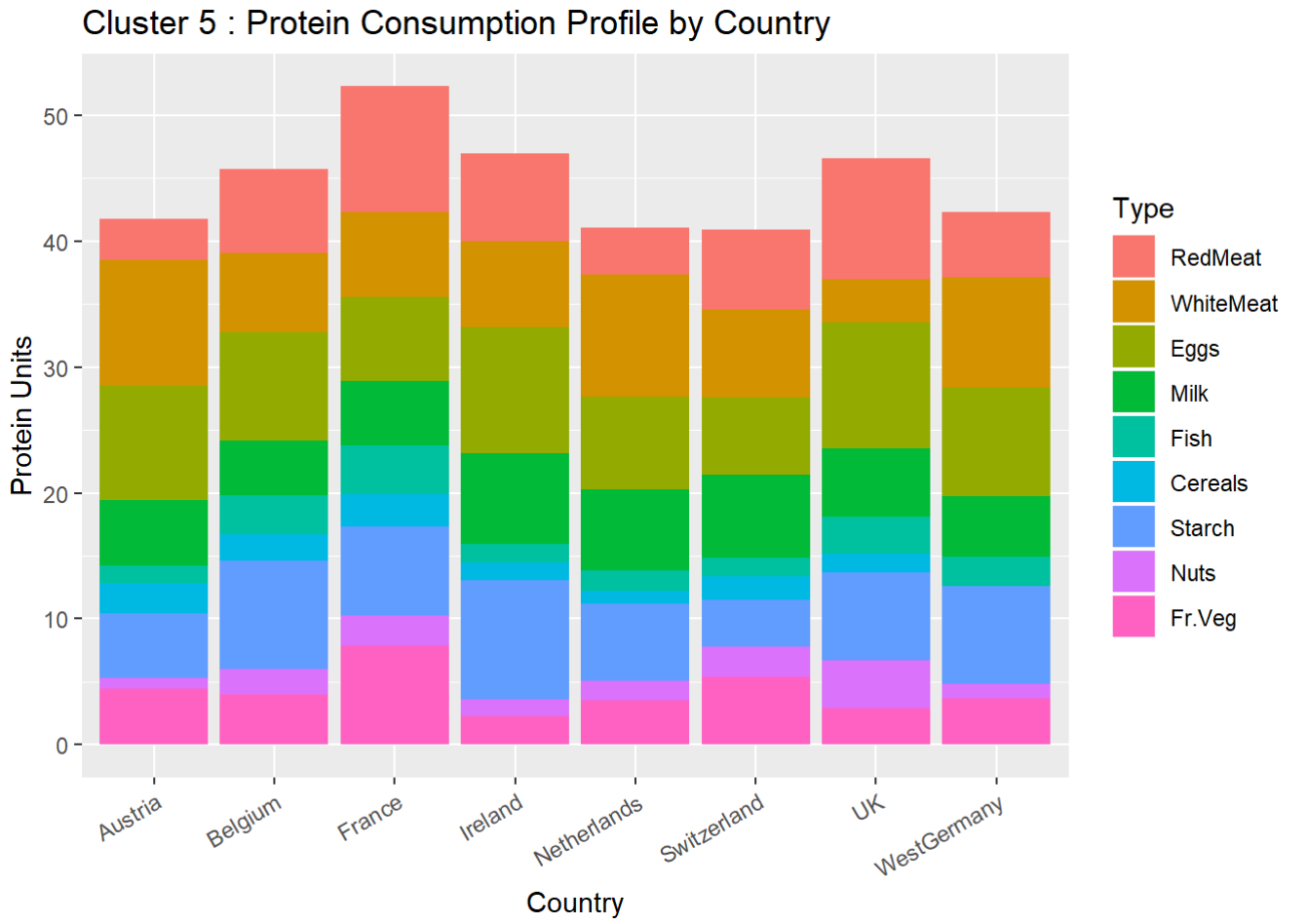
```
# Stack the features
dat<-gather(prot, key=Type, value=Units, RedMeat, WhiteMeat, Eggs, Milk,
            Fish, Cereals, Starch, Nuts, Fr.Veg, factor_key=T)

# Create a list for the plots
plist <- list()

# Generate, store and print the relevant plot for each cluster
for(i in c(1:5)){

  # Throw a page to ensure thata maximum of 2 plots appear on each page
  if(i != 1 & i %% 2 == 1){
    cat("\n\n\\newpage\n")
  }

  dat_clust<-filter(dat, Cluster==i)

  plist[[length(plist) + 1]]<-ggplot(dat_clust,
                                     aes(x=Country ,y=Units,
                                         fill=Type)) +
    geom_bar(stat="identity") +
    xlab("Country")+ylab("Protein Units") +
    ggtitle(paste("Cluster", dat_clust$Cluster,
                  ": Protein Consumption Profile by Country")) +
    theme(axis.text.x = element_text(angle = 30, vjust = 1, hjust=1))

  plot<-plist[[i]]
  print(plot)
}
```


Cluster 1 : Protein Consumption Profile by Country
Protein Units
0
10
20
30
40
Greece
Italy
Portugal
Spain
Country
Type
RedMeat
WhiteMeat
Eggs
Milk
Fish
Cereals
Starch
Nuts
Fr.Veg


Cluster 2 : Protein Consumption Profile by Country
Protein Units
0
10
20
30
40
Czechoslovakia
EastGermany
Hungary
Poland
USSR
Country
Type
RedMeat
WhiteMeat
Eggs
Milk
Fish
Cereals
Starch
Nuts
Fr.Veg


Cluster 3 : Protein Consumption Profile by Country
Protein Units
30
20
10
0
Albania
Bulgaria
Romania
Yugoslavia
Country
Type
RedMeat
WhiteMeat
Eggs
Milk
Fish
Cereals
Starch
Nuts
Fr.Veg


Cluster 4 : Protein Consumption Profile by Country
Protein Units
40
30
20
10
0
Denmark
Finland
Norway
Sweden
Country
Type
RedMeat
WhiteMeat
Eggs
Milk
Fish
Cereals
Starch
Nuts
Fr.Veg



The insights gleaned from the protein consumption profile graphs above can be summarised as follows:

1. **As is to be expected, the protein consumption profiles of countries in the same cluster are similar (with a small number of notable variations - see below).**
2. **EU/EEA countries in the Western and Mediterranean clusters - and Denmark in the Nordic cluster - have the highest overall protein consumptions - at 40-45 (standardised) units each on average.**
3. **The Balkan cluster of communist countries have the lowest overall protein consumptions - at less than 30 units each on average - where Albania (probably the poorest country of the 25) is a prominent low-side outlier with less than 20 units.**
4. **However, Albania doesn't appear to show any consumption of white meat, eggs, fish and starch, so this may be indicative of missing data and requires further examination.**
5. **In the same cluster, Yugoslavia doesn't appear to show any consumption of red meat, which also requires further investigation for the same reason.**
6. **In the Mediterranean cluster, Portugal shows no evidence of milk consumption, which also requires further investigation.**
7. **In the Slavic cluster, East Germany doesn't appear to register any consumption of nuts, which also needs to be investigated.**
8. **In the Nordic cluster, Finland appears to show no consumption of fruit and vegetables, which also warrants attention.**
9. **As was expected, on average, the countries in the 3 EU/EEA clusters get a higher proportion of their protein consumption from the higher-value food types, such as red and white meat and fish, than the less affluent communist countries in the other 2 clusters.**
10. **In the Western cluster, at over 50 units, France shows the highest consumption of protein of any of the 25 countries and, therefore, is a (slight) outlier at the high end.**
11. **In the same cluster, being neighbouring islands, and having a common language, a shared history and a similar culture, the protein consumption profile of Ireland and Britain (UK) is very similar with practically the same total unit consumption.**

12. **Countries in the Mediterranean and Nordic clusters exhibit higher than average fish consumption - due to their extensive coastlines, maritime traditions and large fishing fleets, presumably.**
13. **Countries in the Mediterranean cluster show a higher than average consumption of fruit and vegetables - due to their favourable climatic conditions, presumably.**
14. **Countries in the Nordic countries reveal a lower than average consumption of fruit and vegetables - due to their less favourable climatic conditions, presumably.**
15. **Countries in the Slavic (communist) cluster, on average, have an overall protein consumption that is only slightly below that of their counterparts in the Western cluster - with Poland having an overall protein consumption that reaches the Western average.**
16. **In the Nordic cluster, 3 of the 4 countries have an overall protein consumption that is below the average (with about 36 units each) for other EU/EEA countries. Denmark is the exception here (with about 43 units).**