FOOD CLUSTER ANALYSIS

BUSINESS OBJECTIVE

A food company wants to analyze client's preference in different countries by using the weight per product.

DATA SCIENCE TASK

We will perform a cluster analysis by using characteristics that describe each group.

SELECT DATA

We will use data of 26 countries where people consume the next 11kind of food.

[, 1] Country List of countries
[, 2] ReadMeat Weight (lbs.)
[, 3] WhiteMeat Weight (lbs.)
[, 5] Milk Weight (oz)
[, 6] Fish Weight (lbs.)
[, 7] Cereals Weight (oz)
[, 8] Starch Engine
[, 9] Nuts Weight (oz)
[,10] Fr&Veg Weight (lbs.)

DATA EXPLORATION

This is an example of the first five rows of data:

```
In [19]: data1.head()
```

Out[19]:

	Country	RedMeat	WhiteMeat	Eggs	Milk	Fish	Cereals	Starch	Nuts	Fr&Veg
0	Albania	10.1	1.4	0.5	8.9	0.2	42.3	0.6	5.5	1.7
1	Austria	8.9	14.0	4.3	19.9	2.1	28.0	3.6	1.3	4.3
2	Belgium	13.5	9.3	4.1	17.5	4.5	26.6	5.7	2.1	4.0
3	Bulgaria	7.8	6.0	1.6	8.3	1.2	56.7	1.1	3.7	4.2
4	Czechoslovakia	9.7	11.4	2.8	12.5	2.0	34.3	5.0	1.1	4.0

```
> library(cluster)
> head(ruspini)
   x  y
1  4 53
2  5 63
3 10 59
4  9 77
5 13 49
6 13 69
> plot(ruspini)
> summary(ruspini)
       x                y
 Min.   :  4.00   Min.   :  4.00
 1st Qu.: 31.50   1st Qu.: 56.50
 Median : 52.00   Median : 96.00
 Mean   : 54.88   Mean   : 92.03
 3rd Qu.: 76.50   3rd Qu.:141.50
 Max.   :117.00   Max.   :156.00
```

DATA ANALYSIS

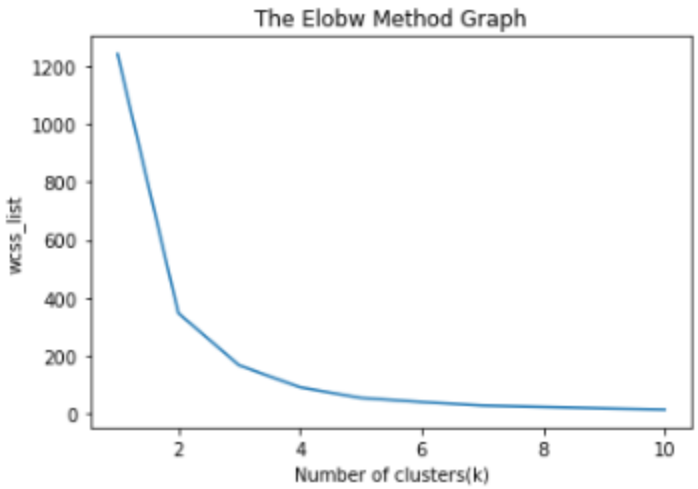
We applied the following steps in our analysis.

APPLY ANALYSIS

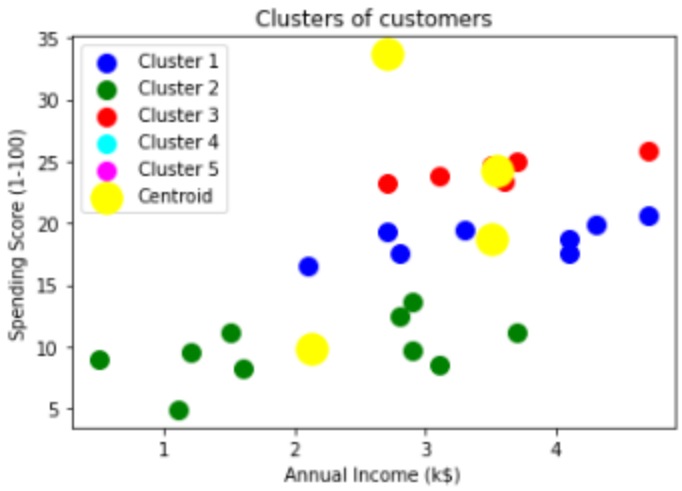
Three of the nine input variable seemed to be important in the model:

K-means Analysis

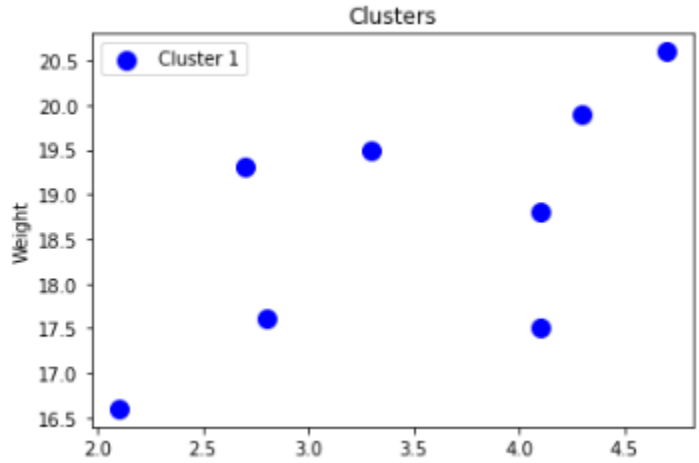
Elbow Plot



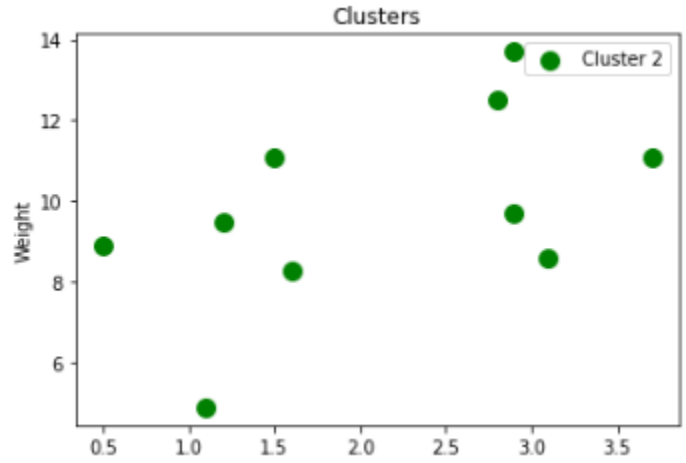
Cluster Solution



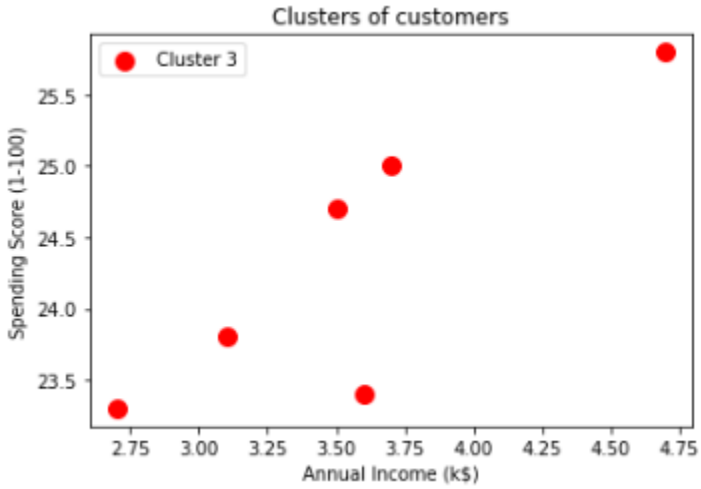
Cluster 1: has a medium weight for almost all of the products.



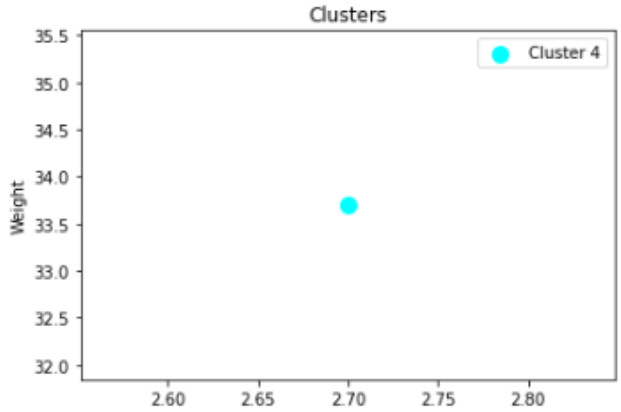
Cluster 2: has a medium high weight for almost all the products.



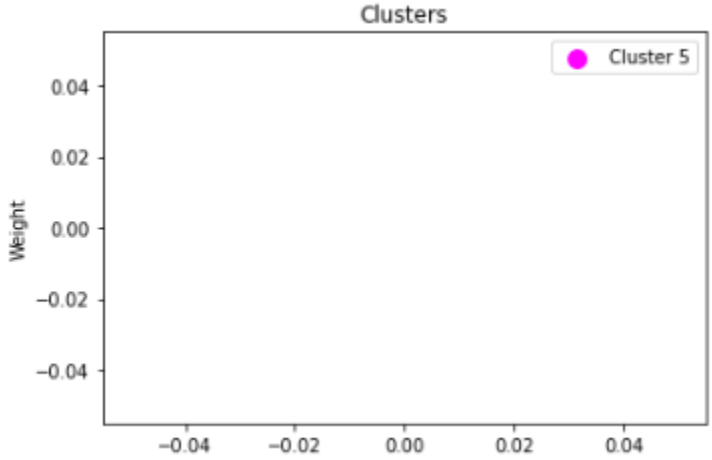
Cluster 3: has a medium lower weight for almost all of the products.

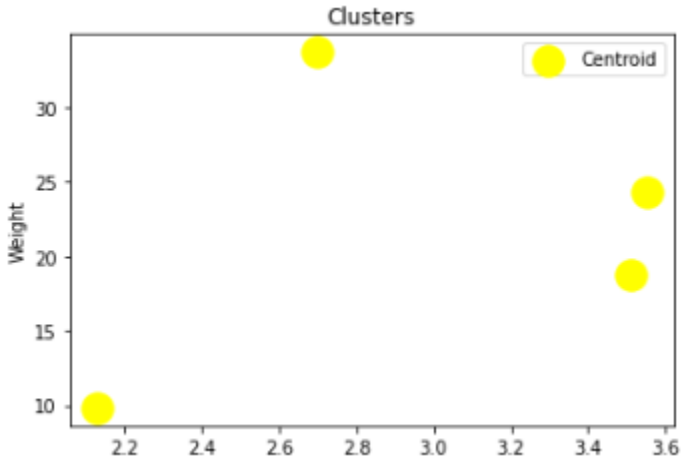


Cluster 4: has a lower weight for almost all of the products.



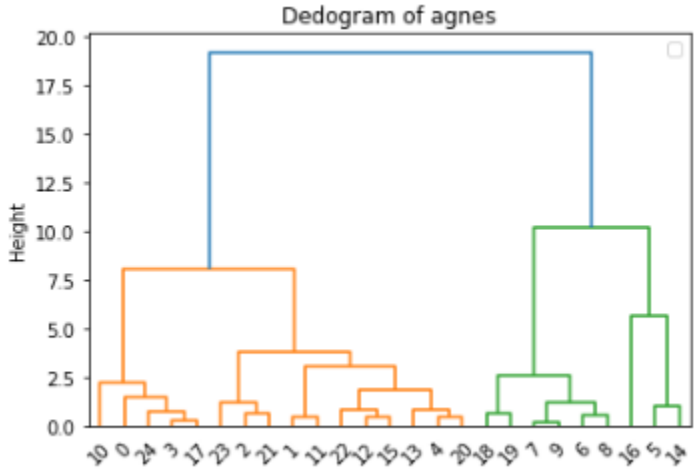
Cluster 4: is not having weight for the products.





Hierarchical Agglomerative Clustering Analysis

These are a ran agnes() using the wards method and created dendrograms with cluster solutions from 3 to 5 to see how the clusters show useful insights to make decisions.



DEPLOY MODEL

We would expect the preference weight per food in different countries.

ASSESS RESULTS

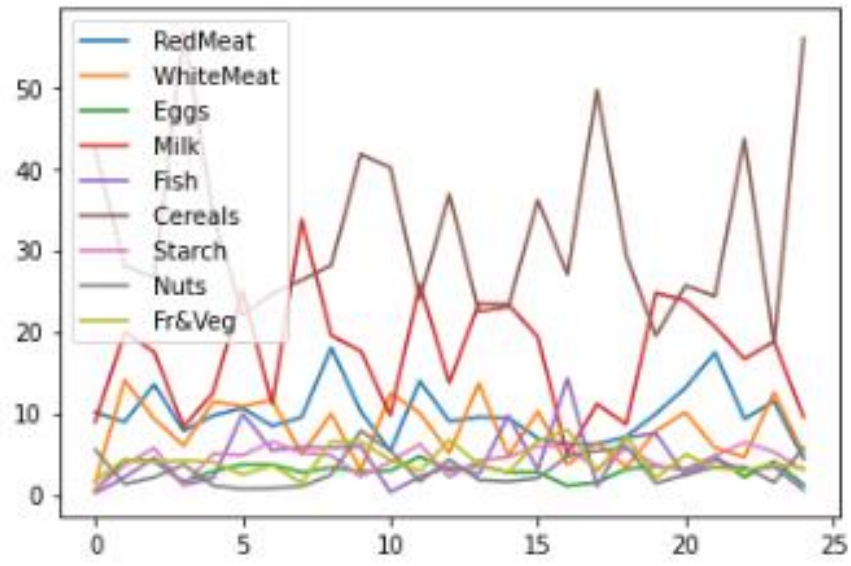
We will evaluate the response rate for the targeted customers compared to the response rate for the random sample to see if there was a greater response rate for the xyz.

STRENGTHS OF XYZ ANALYSIS

The methods show many many strengths:

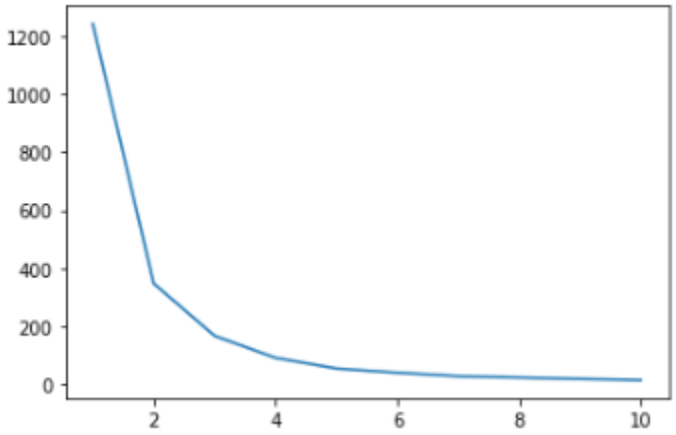
- Albania and Bulgaria have a high production of cereals in some countries to produce others.
- Austria has high production of white meat and milk.
- Belgium has a high production of read meat, white meat and fish.

APPENDIX



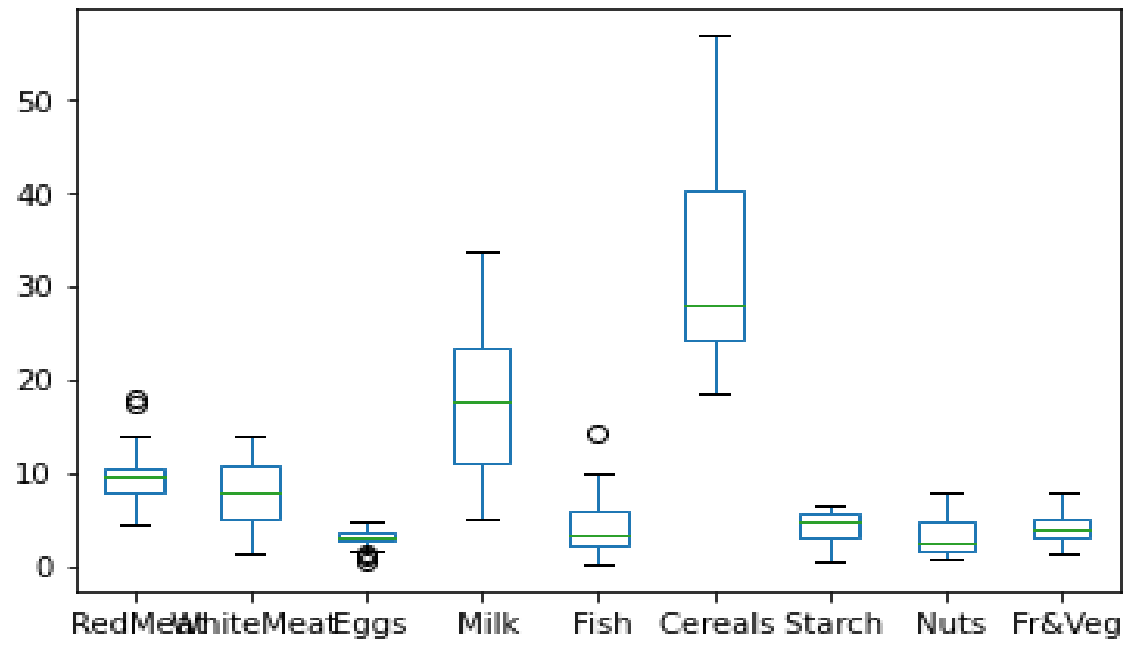
- Cereal and milk have the highest consumption.
- Nuts, eggs and Starch have the lower levels fruits and vegetables.

Kmeans




```
> ### K-Medoids with representative objects
> pam4 <- pam(ruspini, 5)
> pam4
Medoids:
   ID  x   y
10 10 19  65
32 32 44 149
52 52 99 119
47 47 78  94
70 70 69  21
Clustering vector:
 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40
 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  2  2  2  2  2  2  2  2  2  2  2  2  2
41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75
 2  2  2  3  3  4  4  4  3  3  3  3  3  3  3  3  3  3  3  5  5  5  5  5  5  5  5  5  5  5  5  5  5  5  5
Objective function:
   build     swap
12.09864 10.39579

Available components:
 [1] "medoids"    "id.med"     "clustering" "objective"  "isolation"  "clusinfo"   "silinfo"    "diss"       "call"
[10] "data"
```



Code

```
import numpy as nm
```

```
import matplotlib.pyplot as mtp
```

```
import pandas as pd
```

```
url = "protein.csv"
```

```
data1 = pd.read_csv(url)
```

```
from sklearn.cluster import KMeans
```

```
wcss_list= []  #Initializing the list for the values of WCSS
```

```
x = data1.iloc[:, [3, 4]].values
```

```
for i in range(1, 11):
    kmeans = KMeans(n_clusters=i, init='k-means++', random_state= 42)
    kmeans.fit(x)
    wcss_list.append(kmeans.inertia_)
```

```
mtp.plot(range(1, 11), wcss_list)
```

```
mtp.plot(range(1, 11), wcss_list)
mtp.title('The Elobw Method Graph')
mtp.xlabel('Number of clusters(k)')
mtp.ylabel('wcss_list')
mtp.show()
```

```
mtp.scatter(x[y_predict == 0, 0], x[y_predict == 0, 1], s = 100, c = 'blue', label = 'Cluster 1') #for first cluster
mtp.scatter(x[y_predict == 1, 0], x[y_predict == 1, 1], s = 100, c = 'green', label = 'Cluster 2') #for second cluster
mtp.scatter(x[y_predict== 2, 0], x[y_predict == 2, 1], s = 100, c = 'red', label = 'Cluster 3') #for third cluster
mtp.scatter(x[y_predict == 3, 0], x[y_predict == 3, 1], s = 100, c ='cyan', label = 'Cluster 4') #for fourth cluster
mtp.scatter(x[y_predict == 4, 0], x[y_predict == 4, 1], s = 100, c = 'magenta', label = 'Cluster 5') #for fifth cluster
mtp.scatter(kmeans.cluster_centers_[:, 0], kmeans.cluster_centers_[:, 1], s = 300, c = 'yellow', label = 'Centroid')

mtp.title('Clusters of customers')
mtp.xlabel('Annual Income (k$)')
mtp.ylabel('Spending Score (1-100)')
mtp.legend()
mtp.show()
```

```
import pandas as pd
import numpy as np
from matplotlib import pyplot as plt
from sklearn.cluster import AgglomerativeClustering
import scipy.cluster.hierarchy as sch
X = data1.iloc[:, [3, 5]].values
dendrogram = sch.dendrogram(sch.linkage(X, method='ward'))
plt.title('Dedogram of agnes')
plt.xlabel('')
plt.ylabel('Height')
plt.legend()
plt.show()
```

```
data1.plot()
```

```
data1.plot.box()
```