Multivariate Statistical Analysis

Variation of Country Diets and COVID-19

Submitted to Professor Juan Serpa

Student Name: Atrin Morteza Ghasemi

Student McGill ID: 261005342

Student Name: Hadyan Fahreza

Student McGill ID: 261050224

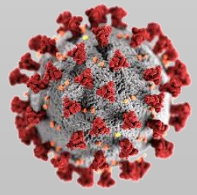


Table of Contents

1. Introduction

"COVID has impacted our lives significantly." This is the sentence we have been hearing for a very long time now and will continue to hear more as we hopefully enter the past-COVID era. And it is true! Many things have changed drastically and as we have gathered a ton of data regarding this pandemic, we are able to perform numerous analyses about this disease and use them to make informed decisions at personal, organizational, and government levels in many different areas.

In our project, we are specifically interested in identifying underlying diet patterns in different countries and tie that information to their COVID information to learn more about if and how a healthy eating style can combat COVID and have an impact of COVID mortality rates so we can adjust our diets accordingly on a personal level and governmental level. Governments can implement policies to effectively leverage such results and inform people on what they can adjust in their diet to better protect themselves and their loved ones against this disease.

In this project, we will be using the latest available version of COVID-19 Healthy Diet Dataset published in Kaggle containing food supply quantities of different countries and their related COVID cases information to perform our analysis. Our goal here is to first identify diet patterns in different countries by using their related data leveraging clustering techniques, and then use a prediction model based on this dietary information to detect the importance of each food category in predicting COVID death rates. Results of these models give us insights in existing patterns of diet in different countries and will also inform us of the impact of food categories and diet on COVID mortality.

2. Data Description

As mentioned, the dataset, "COVID-19 Healthy Diet Dataset" was obtained from Kaggle.com and it contains the percentage breakdown of average food intake for people in countries world-wide for different types of food. It also contains the percentage of undernourished and obese people as well as COVID-19 related information for each country such as percentage of confirmed cases, deaths, recovered, and active cases until June 2nd, 2021.

Continuing on that, we noticed that there are some NA entries in the dataset, specifically on the percentage of death rates of COVID-19 for several countries. To address the issue, we tried to find the real information on the internet and were successful for most of them. Other records were dropped .

Initially, after cleaning the dataset, we had 165 observations and 31 variables. We decided to take out variables which had very small values (average of the variable lower than 0.01) to reduce the dimensions and not losing any significant amount of information. Based on this criterion, we kept 19 variables in total. In this project, we are only interested in capturing dietary patterns in countries and the relation between these dietary information and our target variable COVID death rates. Thus, we removed other unused COVID information such as confirmed, recovered, active, and population.

We did some initial data exploration and plotted histogram of death rate through histograms to have an understanding of its shape and distribution (See Appendix 1). We can see that the distribution is similar to an exponential distribution.

Because the data set contains many dietary variables and it is not practical to plot and observe the relationship of each of these variables with all others (11 dietary variables would require 11 * 10 = 110 scatter plots), we decided to use PCA to analyze the structure of these variables. There are several takeaways based on the PCA diagram (See Appendix 2). To begin with, we can see that there are several variables that are highly correlated such as alcoholic beverages with meat, vegetal products with starchy roots, and sugar with meat. It shows that people who eat a lot of meat also consumes great number of alcoholic beverages and sugar and vice versa. In addition, people who eat a lot of vegetal products also consume great amounts of starchy roots.

Also, from the numerical analysis of the first component, we can see that most of the variability in countries' diets is from the three variables of Animal Products, Vegetal Products, and Milk having the biggest loadings in PCA1 (See Appendix 3).

Furthermore, it shows where the dietary pattern of different countries falls. The PCA arrows for each food type indicate the tendencies of the diets of these people. It can be seen that most countries fall on the left-end and right-end of the PCA diagram meaning there could be a group of countries' diets which consist of the combination of vegetal products, starchy roots, and cereals, and another group where the main diet is the combination of animal products, milk, and alcoholic beverages. Since the first two main components are capturing slightly more than half the

information of the dataset (37.59% for 1st component and 14.2% for 2nd component), this analysis gives us some preliminary insights into how countries' diets are grouped and we will go deeper on this analysis in the clustering section.

3. Model Selection & Methodology

3.1. Clustering countries based on dietary data

As mentioned, in the first step, we will try to group different countries into clusters based on their dietary information to detect underlying patterns of food intake. We will be using K-means algorithm to perform this clustering and to tackle the challenge of visualizing clusters when we have a lot of variables, we will leverage a chart type known as radar chart or spider chart.

We begin by selecting the subset of dietary variables in our dataset which we want to perform the clustering on. Since K-means is a distance-based algorithm and very sensitive to the scale of the variables, we first have to standardize these variables so they are in the same scale. We do that by using *scale* function in R. After standardization, we have to come up with the number of clusters we are looking for to tell K-means to group the data into that number of clusters. To do that, we leveraged a technique called Elbow method which is a very common technique to determine the appropriate number of clusters. We will try to choose the small value of K where the change in the total within sum of squares reduction becomes comparatively negligible. Based on the chart (See Appendix 4), we set the value of k to 4 which seems like the appropriate number based on the described criterion.

We train the K-means algorithm on the data by setting the value of k to 4 and also since the results depend on initial sorting, we set the value of nstart to 20 to run the algorithm 20 times and find the best cluster results among them. Then we label each row to its designated cluster based on the clustering model output.

Now that we have the clusters, apart from analyzing the numerical results, we also wanted to find a way to visualize these outputs. Since we have a lot of variables, it is not an easy task to visualize the results, but we were able to use a chart known as the radar chart or the spider chart to capture information of these clusters. We will extract insights out of these visualizations in the results and managerial conclusions section

3.2. Prediction Model on COVID-19 Death Rates

Following our clustering analysis, we want to formulate a prediction model that forecasts the COVID-19 death rate given the average percentage of food types. To begin with, we generated a random forest model to find out the importance of each predictor relative to the target death rate.

From the variable importance plot (See Appendix 5), we can see the ranking of importance based on percentage increase of mean squared error and increase of residual sum of squares if the subsequent food type is removed from the model. We decided to have two approaches for the prediction model; including all food types as predictors, and removing 5 least important food types: starchy roots, fruits, vegetables, sugar, and meat. Then, we will compare them based on the mean squared error generated by the model using each scenario.

Furthermore, we used gradient boosting to develop our prediction model. Before feeding the data to the model, we split the dataset into training and test dataset with the test dataset size of 0.3. Unlike the clustering task we did not standardize the predictors and feed the values directly to gradient boosting tree since it is not influenced by the scale of the variables .

Then we created gradient boosting model for both approaches, initially without specifying hyperparameters other than distribution = "gaussian", since we were doing regression tasks, and the number of trees. Then, we predicted the death rate using test dataset and calculated the mean squared error of each model using the prediction and the actual death rate contained in the test dataset. The model with the least mean squared error will be selected and that subsequent model will be used to discuss the relative importance of the predictors to the COVID-19 death rate.

4. Results

4.1. Clustering Results

As described in the model selection section, we grouped the data into 4 clusters. A summary table for the numerical cluster results is given the appendices section (See Appendix 6). After visualizing the data using the radar chart based on the mean of the four clusters for each variable (For radar chart of all cluster, See Appendix 7), we assigned a name to each cluster which represented clusters' primary characteristics. The names are defined to give some idea about each

cluster and to avoid calling them Cluster 1 to Cluster 4 during the entire analysis. Individual cluster charts are also drawn to more clearly present each cluster's characteristics (See Appendix 8). Clusters are described below:

- **Animal Consumers:** People in these countries heavily consume food produced by and from animals including animal products, meat, and milk (except seafood, they don't seem to like fish). Their diet is mainly based on these types of food and they don't really like vegetables, vegetal products, and they kind of hate starchy roots for some reason. Like alcohol a bit as well.
- **Animal Lovers :** On the other side of the spectrum, these people are all for animals. They don't like to eat animal products, meat, or milk at all. They enjoy their starchy roots and most importantly they are all for vegetal products. They also like cereals and a bit of fruits.
- **Adequately-Balanced (As all things should be):** People from these countries maintain an adequate balance of all food. They like good food whatever it may be as long as it's not starchy roots. They also enjoy a good level of sugary foods (pastries!) and an adequate level of alcohol.
- **Vegetable-Enjoyers:** These people really like vegetables and vegetal products. They enjoy them both very much and take life very seriously since they don't consume alcohol almost at all. They also like cereals but don't enjoy starchy roots.

Apart from the mean value of each cluster in each shown in the summary table, we can also look into the total observations in each cluster and the within cluster sum of squared errors which are given below:

Measure	Animal Lovers	Animal Consumers	Adequately-Balanced	Vegetable-Enjoyers
Total Number of Observations	38	41	37	49
Within-Cluster Sum of Squared Errors	191.006	310.384	292.851	180.234

This is from the best result of the 20 times the algorithm was run.

4.2 Prediction Model Results

With the regression task, using random forest and gradient boosting, we focused on finding out types of food which are important and types of food which are insignificant in predicting death rate, as well as getting the magnitude of importance of each type. Our main goal was to identify predictors who contribute the most to the prediction of the death rate to realize their importance in the mortality of COVID.

The mean squared error we got when we included all food types was 0.002, while the mean squared error we got when we excluded 5 least important predictors was 0.001. Consequently, the latter approach forecasts better with lesser predictors. Initially, after the first run, we were considering hyperparameter tuning, however, given the resulting small MSEs we decided not to do tuning to preserve generalizability. Since the model without 5 least significant predictors performed better, the analysis of feature importance will be drawn based on the predictors used for that model.

From the summary of gradient boosting model providing relative importance (See Below), it can be seen that milk has the greatest effect on the death rate prediction model. In addition to that, based on the initial correlations, it is known that the correlation between milk and death rate is 0.545. The food type with the second greatest effect on the COVID-19 death rate is alcoholic beverages. In addition to that, based on the initial correlations, it is known that the correlation between alcohol and death rate is 0.412. So, the result of the prediction model is aligned with the earlier findings we had about variable correlations. Top two variables are much more important the rest 4 and there is a significant drop in importance for all the others. Also, vegetal products' importance seems inconsequential.

Variable	Relative Importance
Milk	35.799
Alcoholic Beverages	24.668
Seafood	13.262
Animal Products	11.586
Cereals	8.229
Vegetal Products	0.331

Relative Importance of Variables in predicting COVID Death Rates for Gradient Boosting Tree Algorithm

We are keeping in mind that this analysis does not indicate causation. Further research might be needed to determine the actual causal effects of each type of food on the COVID-19 death rate.

5. Managerial Conclusions

5.1. Dietary Insights

In this project, we tried to identify underlying patterns of diets from different countries around the world. When we face a clustering task, unlike prediction tasks, we are not setting a pre-defined objective, but we try to gain insights into what patterns and structure exists in the data. In this case, we found out that we can group countries based on their diet into 4 clusters. People in these countries consume different amounts of different types of food. Our findings suggested that the diet of the majority people of countries can either be heavily dependent on vegetables and cereals, animal products, starchy roots and vegetal products, or a somewhat balanced diet between all types. We could probably have guessed about the first two diets before doing any analysis (which can happen in real-life analytics as well that the results of a rigorous analysis may confirm what we could have guessed), but the last two groups give us new and interesting insights. Some countries overall are maintaining a balance in their diets and there is a group of countries that use starchy roots and vegetal products as their main source of diet which we could not have guessed without this analysis.

Furthermore, we calculated the mean of obesity and undernourishment for these different clusters (See Appendix 9) to capture more insights from these clusters. As the table suggests, we can see that animal consumers have the highest obesity rates and lowest undernourishment rates. These countries are most likely the richest ones whose people can mostly afford food but they are quite unhealthy in their diets. We also see that the adequately-balanced countries are facing a serious undernourishment rate and this suggests that these people are maintaining a balanced diet maybe because they do not have access or cannot afford to eat meat and other animal products which are typically more expensive. The vegetable-enjoyers are similar to them, but with their increased obesity rate and decreased undernourishment, we can infer that they have a better situation regarding access to food and affordability, and for them, it is more a matter of choice to eat healthy food and have a good diet rather than a force of the situation.

Other than the governments who can use this information to discover countries similar to them diet-wise and inform their people to change their diets in a right direction based on their situation (every group is facing its own problems and have to implement corrective measures), food manufacturing companies and any company in this field can also leverage this information to find similar target countries which can potentially be a new market for them. If they are currently operating in some of the countries in one of these clusters, they can do research and target other countries within the clusters as well to expand their business.

5.2. Covid Insights

Our findings showed that milk is the most important food type in determining the COVID-19 death rate. Having a positive correlation with the death rate means that the more people in a country consume milk on their diet, the higher the death rate is. This is surprising because milk often regarded as a source of a lot of good nutrients, and there is little to no article and research done on the effects of milk to COVID-19 death rate.

Furthermore, alcohol beverages have the second most significant effect in determining COVID-19 death rate. Same as milk, it also has positive correlation with the death rate. This finding makes more sense because alcohol often times is regarded as a beverage that pose a lot of risks and negative effects on our wellbeing.

On the other hand, although having much lower importance, seafood, cereal, and vegetal products have negative correlation with the COVID-19 death rate. It means the more people in a country consume it, the lower the death rate would be, but not to the extent that milk and alcoholic beverages affect the death rate.

With the uncertainty of how long the virus is going to last, and with new variants appearing every now and then even with the enhanced immunity provided by vaccines, we may need to find the way to keep the death rate from COVID-19 as low as possible, and one possible way to do it is to getting the right nutrients to combat the virus and avoid substances that affect body negatively given the infection.

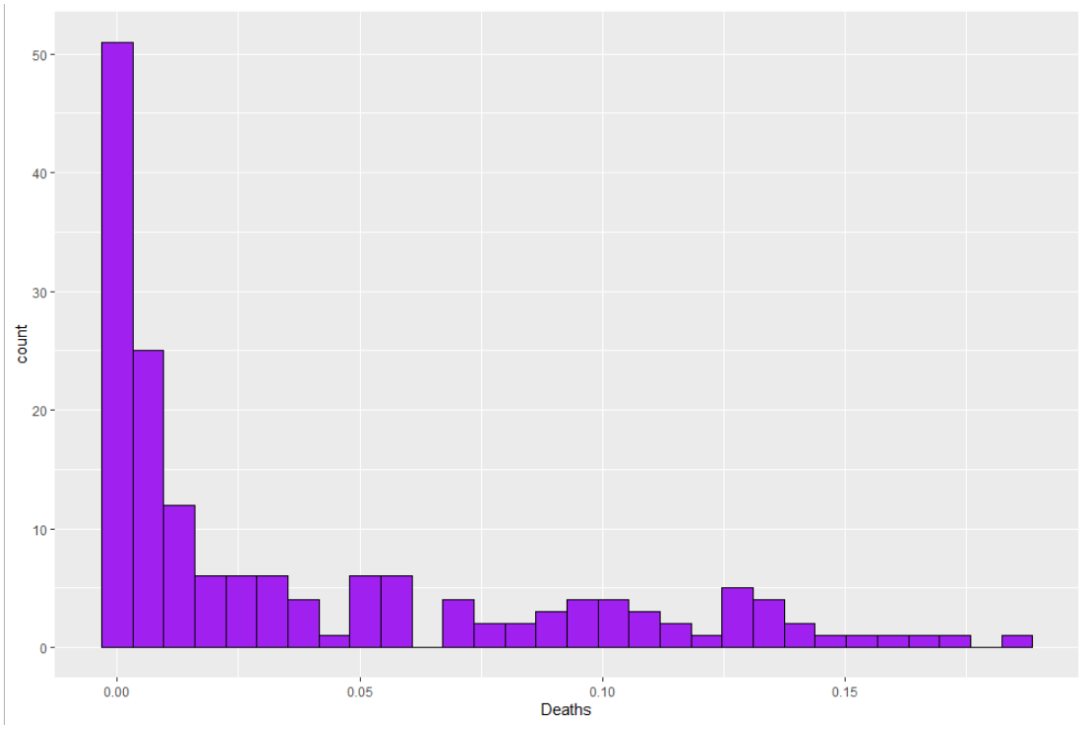
This information on relative importance of each food type could serve as an exploratory analysis for further research on how food types affect the chance of surviving from COVID-19

infection. It could also serve as the guidance for research prioritization and division. Because we know that some food types affect the death rate output of the model greatly in unfavorable way, researchers may want to scope down the focus on these food types and how to minimize the harms inflicted by these food types. With that said, we also know that some food types reduce the chance of dying from COVID-19 on the model we developed. Thus, it is also important to divert some of the resources to study this further to come up with dietary guidance and/or supplements that help people to gain essential nutrients needed to fight the virus.

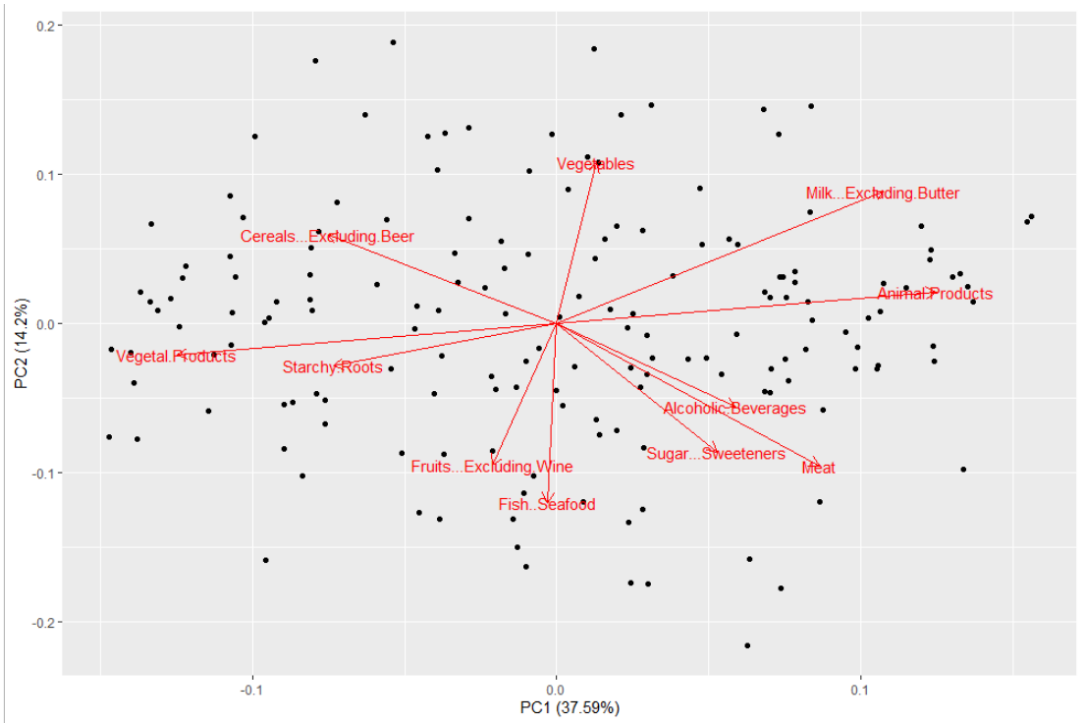
The prediction model we developed can also be used for the governments to test their approach after putting together a specific diet plan and awareness raising for their people constructed based on our clustering analysis results. It could give a rough estimate of the death rate if people actually followed the proposed plan. Observations can be made to verify if these measures are actually significantly affecting the COVID death rates and to make adjustments if needed.

6. Appendices

Appendix 1: Death Rates Histogram



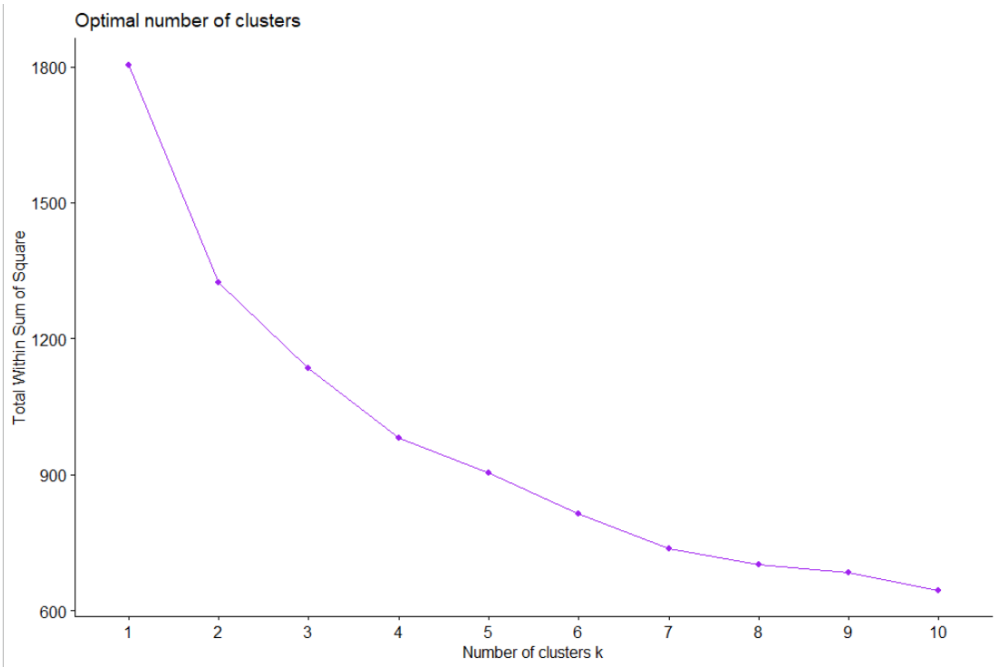
Appendix 2: PCA Chart



Appendix 3: PCA 1 Numerical Loadings

Variable	PCA1
Alcoholic.Beverages	0.225
Animal.Products	0.477
Cereals	-0.288
Seafood	-0.011
Fruits	-0.080
Meat	0.331
Milk	0.412
Starchy.Roots	-0.281
Sugar	0.202
Vegetables	0.050
Vegetal.Products	-0.477

Appendix 4: Elbow Chart



Elbow method chart to determine appropriate number of clusters

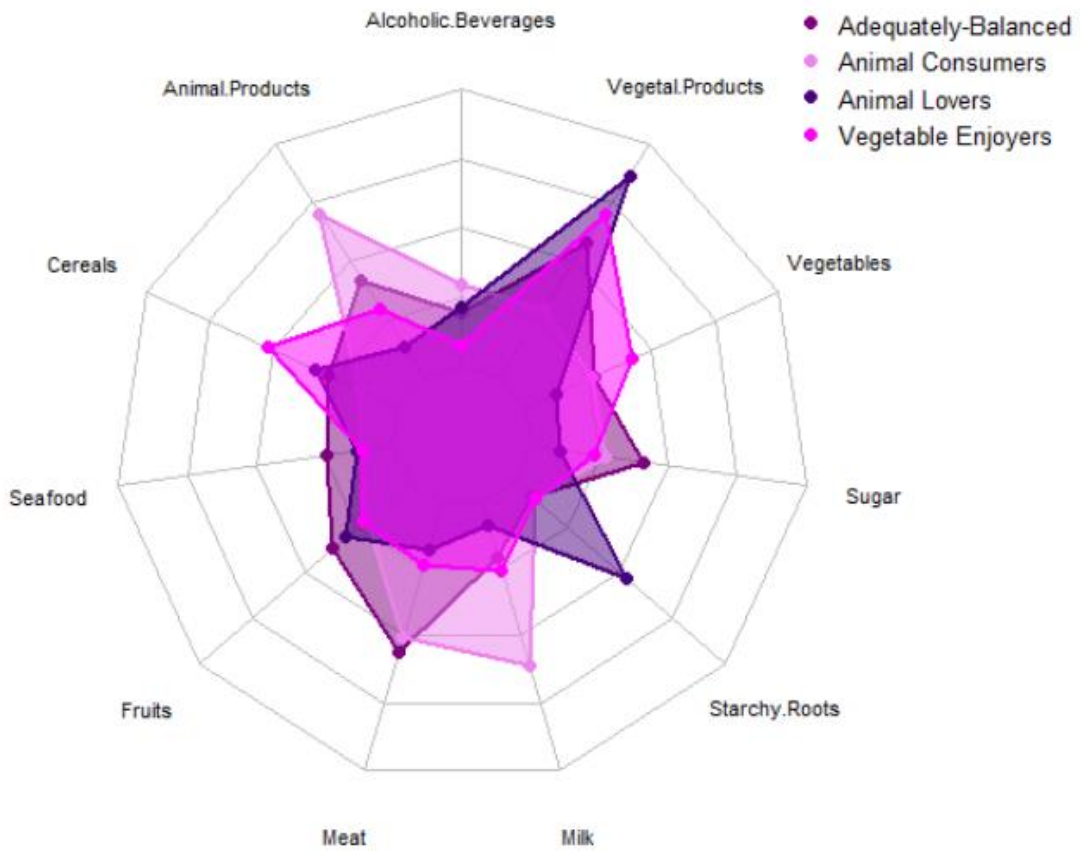
Appendix 5: Feature Importance

Variable	%Increase in MSE
Alcoholic.Beverages	15.383
Animal.Products	13.992
Cereals	6.244
Seafood	9.102
Fruits	3.598
Meat	6.694
Milk	20.057
Starchy.Roots	-2.162
Sugar	6.214
Vegetables	3.913
Vegetal.Products	14.831

Appendix 6 : Cluster Means Values for Each Variable

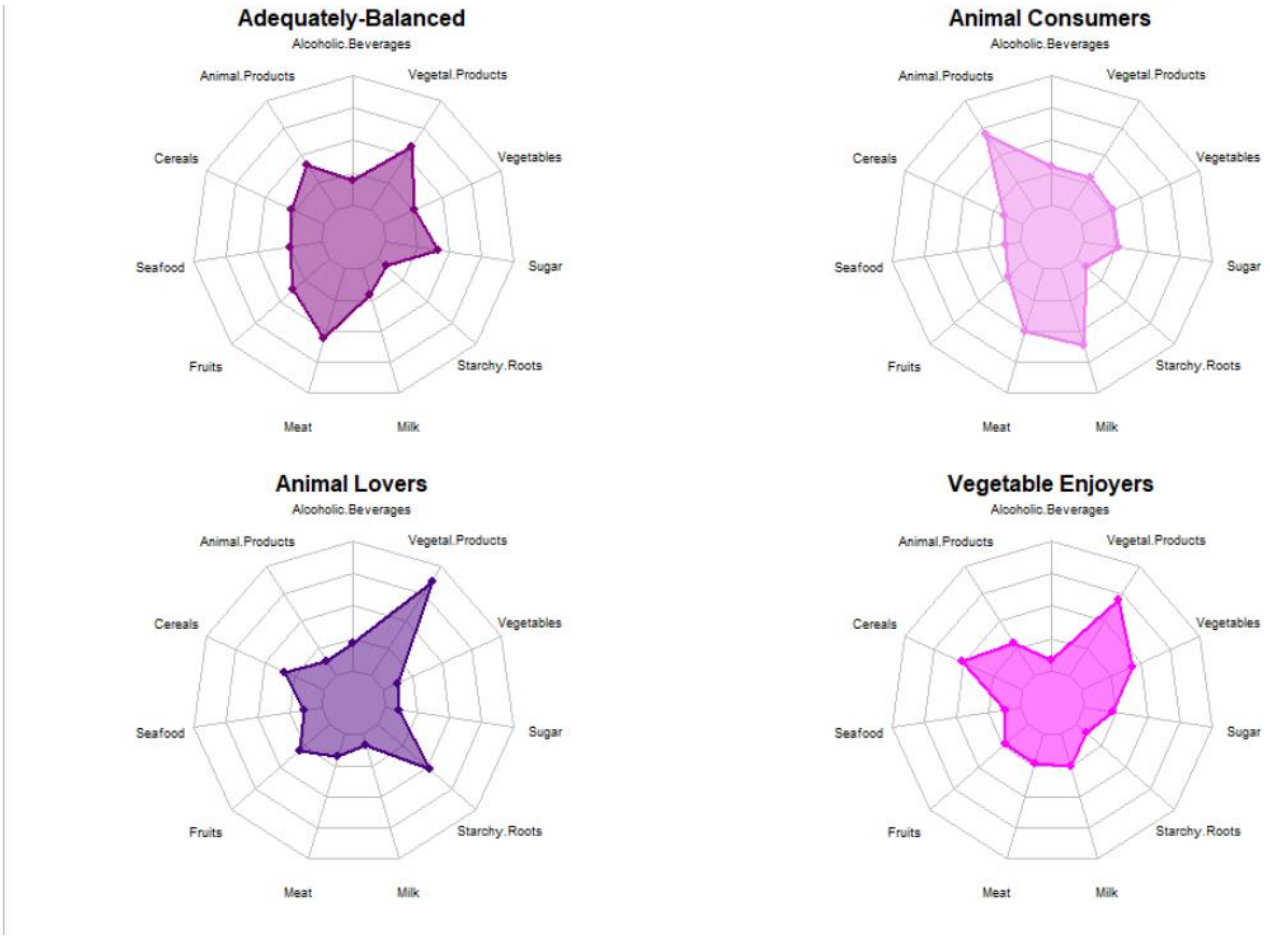
Variable	Animal Lovers	Animal Consumers	Adequately-Balanced	Vegetable-Enjoyers
Alcoholic.Beverages	-0.821	-0.003	0.001	0.638
Animal.Products	-0.421	-0.022	-1.159	1.221
Cereals	0.606	-0.093	0.417	-0.707
Seafood	-0.480	0.720	-0.002	-0.228
Fruits	-0.106	0.519	0.022	-0.368
Meat	-0.648	0.744	-0.883	0.546
Milk	-0.107	-0.488	-0.956	1.214
Starchy.Roots	-0.412	-0.453	1.423	-0.375
Sugar	-0.179	0.881	-0.869	0.057
Vegetables	1.087	-0.125	-0.835	-0.107
Vegetal.Products	0.421	0.022	1.159	-1.221

Appendix 7 : Radar Chart of all CLusters



Radar chart of all clusters together

Appendix 8: Radar Chart for each Cluster



Individual Radar charts for each cluster

Appendix 9: Mean of Obesity and Undernourishment

Variable	Animal Lovers	Animal Consumers	Adequately-Balanced	Vegetable-Enjoyers
Obesity	21.812	24.761	9.825	15.253
Undernourishment	8.396	3.215	23.967	13.280

7. Code

```
require(ggplot2)
library(ggfortify)
library(cluster)
library(fmsb)
library(GGally)
library(gridExtra)
library(randomForest)
library(gbm)
library(MASS)
library(klaR)
library(cowplot)

Food_Supply_Quantity_kg_Data <-
read.csv("C:/Users/Asus/Desktop/Food_Supply_Quantity_kg_Data.csv")

df = Food_Supply_Quantity_kg_Data

#================================================================
================================#
##### Data Pre-processing

## Drop records with multiple missing values
df = df[-c(53,110,81,82,156),]

## Gather information on missing values from the web and input into the dataframe
df$Obesity[df$Country == 'Taiwan*'] = 22.8
df$Confirmed[df$Country == 'Myanmar'] = 250000/54704000
df$Deaths[df$Country == 'Myanmar'] = 9000/54704000

df$Undernourished[df$Undernourished == '<2.5'] = 1.25
```

```
df$Undernourished[df$Country == 'Antigua and Barbuda'] = 20.5
df$Undernourished[df$Country == 'Bahamas'] = 5.1
df$Undernourished[df$Country == 'Grenada'] = 25.5
df$Undernourished[df$Country == 'Republic of Moldova'] = 4
df$Undernourished[df$Country == 'Saint Kitts and Nevis'] = 10.2
df$Undernourished[df$Country == 'Saint Lucia'] = 27.3
df$Undernourished[df$Country == 'Tajikistan'] = 33.2

## Identify weak variables and drop them (average below 1 percent)
useless_variables = c()

for( i in seq(2, 23)){
    if(mean ( df[,i] ) <=1 ){
      useless_variables = c(useless_variables, i)
    }
}

df <- df[ -c(3, 5, 7, 12, 13, 14, 15, 16, 18, 20, 21, 22) ]

df$Undernourished = as.double(df$Undernourished)

df <- df[, -c(15, 17, 18, 19) ]

#=================================================================
==============================#

###### Data Description

## Histogram
hist_mr = ggplot(data=df, aes(Deaths)) + geom_histogram(color='black', fill='purple')
```

```
## Correlation Matrix
cor_df = cor(df[,-c(1)])

## PCA
pca=prcomp(df[c(2:12)], scale=TRUE)
autoplot(pca, data = df[c(2:12)], loadings = TRUE, loadings.label = TRUE)

#=================================================================
================================#

#### Model Creation and Selection Part I: Clustering

df_clustering = df[,2:12]
rownames(df_clustering)=df$Country
df_clustering_std = as.data.frame(scale(df_clustering))

## Finding Optimal Number of Clusters with Elbow Method

#Elbow Method for finding the optimal number of clusters
# Compute and plot wss for k = 2 to k = 8.
set.seed(123)
fviz_nbclust(df_clustering_std, kmeans, method = "wss", linecolor = 'purple')

### Optimal Number of Clusters: 3 or 4

km.4=kmeans(df_clustering_std, 4, nstart = 20) #4 clusters
df_clustering_std$cluster=as.factor(km.4$cluster)

#=================================================================
================================#

#### Model Creation and Selection Part II: Prediction
```

```
#Splitting Data set to Training and Test Set

## 70% of the sample size
smp_size <- floor(0.7 * nrow(df))

## set the seed to make your partition reproducible
set.seed(123)
train_ind <- sample(seq_len(nrow(df)), size = smp_size)
train <- df[train_ind, ]
test <- df[-train_ind, ]

# Random Forest Regression with All Predictors, with Summary and Importance Plot
myforest=randomForest(Deaths~Alcoholic.Beverages+Animal.Products+Cereals...Excluding.Beer+Fish..Seafood+Fruits...Excluding.Wine+Meat+Milk...Excluding.Butter+Starchy.Roots+Sugar...Sweeteners+Vegetables+Vegetal.Products, ntree=1000, data=train, importance=TRUE, na.action = na.omit)
myforest
importance(myforest)
varImpPlot(myforest)

# Random Forest Regression without 5 Last Significant Predictors, with Summary and Importance Plot (put in the appendix)
myforest=randomForest(Deaths~Alcoholic.Beverages+Animal.Products+Cereals...Excluding.Beer+Fish..Seafood+Milk...Excluding.Butter+Vegetal.Products, ntree=1000, data=train, importance=TRUE, na.action = na.omit)
myforest
varImpPlot(myforest)

#Gradient Boosting Regression Model using All Predictors
gbmforest=gbm(Deaths~Alcoholic.Beverages+Animal.Products+Cereals...Excluding.Beer+Fish..Seafood+Fruits...Excluding.Wine+Meat+Milk...Excluding.Butter+Starchy.Roots+Sugar...Sweeteners+Vegetables+Vegetal.Products, data=train, distribution="gaussian", n.trees=10000, interaction.depth=4)
```

```
summary(gbmforest)

# Testing Gradient Boosting Regression Model using All Predictors with Test Set
predicted_score=predict(gbmforest, newdata=test)
mean((predicted_score - test$Deaths)^2)

# Gradient Boosting Regression Model without 5 Least Significant Predictors (put in the appendix)
gbmforest=gbm(Deaths~Alcoholic.Beverages+Animal.Products+Cereals...Excluding.Beer+Fish..Seafood+Milk...Excluding.Butter+Vegetal.Products, data=train, distribution="gaussian")
summary(gbmforest)

# Testing Gradient Boosting Regression Model without 5 Least Significant Predictors with Test Set
predicted_score=predict(gbmforest, newdata=test)
mean((predicted_score - test$Deaths)^2)

#=====================================================================
==================================#

## Create a Radar (Spyder) Diagram
# Create Dataframes to draw the Radar Chart for Clusters
max_min <- data.frame(
Alcoholic.Beverages = c( max(df_clustering_std$Alcoholic.Beverages), min(df_clustering_std$Alcoholic.Beverages)),
Animal.Products = c( max(df_clustering_std$Animal.Products), min(df_clustering_std$Animal.Products)),
Cereals = c( max(df_clustering_std$Cereals...Excluding.Beer), min(df_clustering_std$Cereals...Excluding.Beer)),
Seafood = c( max(df_clustering_std$Fish..Seafood), min(df_clustering_std$Fish..Seafood)),
Fruits = c( max(df_clustering_std$Fruits...Excluding.Wine), min(df_clustering_std$Fruits...Excluding.Wine)),
Meat = c( max(df_clustering_std$Meat), min(df_clustering_std$Meat)),
Milk = c( max(df_clustering_std$Milk...Excluding.Butter), min(df_clustering_std$Milk...Excluding.Butter)),
```

```
Starchy.Roots = c( max(df_clustering_std$Starchy.Roots), min(df_clustering_std$Starchy.Roots)),

Sugar = c( max(df_clustering_std$Sugar...Sweeteners),
min(df_clustering_std$Sugar...Sweeteners)),

Vegetables = c( max(df_clustering_std$Vegetables), min(df_clustering_std$Vegetables)),

Vegetal.Products = c( max(df_clustering_std$Vegetal.Products),
min(df_clustering_std$Vegetal.Products))

)


rownames(max_min) <- c("Max", "Min")

clusters <- data.frame(

Alcoholic.Beverages =
c(mean(df_clustering_std[df_clustering_std$cluster==1,"Alcoholic.Beverages"]),
                    mean(df_clustering_std[df_clustering_std$cluster==2,"Alcoholic.Beverages"]),
                    mean(df_clustering_std[df_clustering_std$cluster==3,"Alcoholic.Beverages"]),
                    mean(df_clustering_std[df_clustering_std$cluster==4,"Alcoholic.Beverages"])
                    ),
Animal.Products = c(mean(df_clustering_std[df_clustering_std$cluster==1,"Animal.Products"]),
                    mean(df_clustering_std[df_clustering_std$cluster==2,"Animal.Products"]),
                    mean(df_clustering_std[df_clustering_std$cluster==3,"Animal.Products"]),
                    mean(df_clustering_std[df_clustering_std$cluster==4,"Animal.Products"])
                    ),
Cereals = c(mean(df_clustering_std[df_clustering_std$cluster==1,"Cereals...Excluding.Beer"]),
                    mean(df_clustering_std[df_clustering_std$cluster==2,"Cereals...Excluding.Beer"]),
                    mean(df_clustering_std[df_clustering_std$cluster==3,"Cereals...Excluding.Beer"]),
                    mean(df_clustering_std[df_clustering_std$cluster==4,"Cereals...Excluding.Beer"])
                    ),
Seafood = c(mean(df_clustering_std[df_clustering_std$cluster==1,"Fish..Seafood"]),
                    mean(df_clustering_std[df_clustering_std$cluster==2,"Fish..Seafood"]),
                    mean(df_clustering_std[df_clustering_std$cluster==3,"Fish..Seafood"]),
                    mean(df_clustering_std[df_clustering_std$cluster==4,"Fish..Seafood"])
                   ),
 Fruits = c(mean(df_clustering_std[df_clustering_std$cluster==1,"Fruits...Excluding.Wine"]),
```

```
                mean(df_clustering_std[df_clustering_std$cluster==2,"Fruits...Excluding.Wine"]),
                mean(df_clustering_std[df_clustering_std$cluster==3,"Fruits...Excluding.Wine"]),
                mean(df_clustering_std[df_clustering_std$cluster==4,"Fruits...Excluding.Wine"])
                ),
Meat = c(mean(df_clustering_std[df_clustering_std$cluster==1,"Meat"]),
                 mean(df_clustering_std[df_clustering_std$cluster==2,"Meat"]),
                 mean(df_clustering_std[df_clustering_std$cluster==3,"Meat"]),
                 mean(df_clustering_std[df_clustering_std$cluster==4,"Meat"])
                 ),
Milk = c(mean(df_clustering_std[df_clustering_std$cluster==1,"Milk...Excluding.Butter"]),
                mean(df_clustering_std[df_clustering_std$cluster==2,"Milk...Excluding.Butter"]),
                mean(df_clustering_std[df_clustering_std$cluster==3,"Milk...Excluding.Butter"]),
                mean(df_clustering_std[df_clustering_std$cluster==4,"Milk...Excluding.Butter"])
                ),
Starchy.Roots = c(mean(df_clustering_std[df_clustering_std$cluster==1,"Starchy.Roots"]),
                 mean(df_clustering_std[df_clustering_std$cluster==2,"Starchy.Roots"]),
                 mean(df_clustering_std[df_clustering_std$cluster==3,"Starchy.Roots"]),
                 mean(df_clustering_std[df_clustering_std$cluster==4,"Starchy.Roots"])
                 ),
Sugar = c(mean(df_clustering_std[df_clustering_std$cluster==1,"Sugar...Sweeteners"]),
                 mean(df_clustering_std[df_clustering_std$cluster==2,"Sugar...Sweeteners"]),
                 mean(df_clustering_std[df_clustering_std$cluster==3,"Sugar...Sweeteners"]),
                 mean(df_clustering_std[df_clustering_std$cluster==4,"Sugar...Sweeteners"])
                 ),
    Vegetables = c(mean(df_clustering_std[df_clustering_std$cluster==1,"Vegetables"]),
                 mean(df_clustering_std[df_clustering_std$cluster==2,"Vegetables"]),
                 mean(df_clustering_std[df_clustering_std$cluster==3,"Vegetables"]),
                 mean(df_clustering_std[df_clustering_std$cluster==4,"Vegetables"])
                 ),
Vegetal.Products = c(mean(df_clustering_std[df_clustering_std$cluster==1,"Vegetal.Products"]),
                mean(df_clustering_std[df_clustering_std$cluster==2,"Vegetal.Products"]),
```

```
                mean(df_clustering_std[df_clustering_std$cluster==3,"Vegetal.Products"]),
                mean(df_clustering_std[df_clustering_std$cluster==4,"Vegetal.Products"])
                )
    )

rownames(clusters) <- c("Adequately-Balanced", "Animal Consumers","Animal
Lovers","Vegetable Enjoyers")
spyder_df = rbind(max_min, clusters)

## Draw the Radar  Chart
#install.packages('fmsb')
spyder_df <- spyder_df[c("Max", "Min", "Adequately-Balanced", "Animal Consumers","Animal
Lovers","Vegetable Enjoyers"), ]

## Create Beautiful Radar Chart
create_beautiful_radarchart <- function(data, color = "#00AFBB",
                                        vlabels = colnames(data), vlcex = 0.7,
                                        caxislabels = NULL, title = NULL, ...){
radarchart(
      data, axistype = 0,
      # Customize the polygon
      pcol = color, pfcol = scales::alpha(color, 0.5), plwd = 2, plty = 1,
      # Customize the grid
      cglcol = "grey", cglty = 1, cglwd = 0.8,
      # Customize the axis
      axislabcol = "grey",
      # Variable labels
      vlcex = vlcex, vlabels = vlabels,
      caxislabels = caxislabels, title = title, ...
    )
}
```

```
# Reduce plot margin using par()
op <- par(mar = c(1, 2, 2, 2))
par(mfrow=c(1,1))

# Create the radar charts
create_beautiful_radarchart(
    data = spyder_df,
    color = c("#800080", "#EE82EE", "#4B0082","#FF00FF")
)

# Add an horizontal legend
legend(
    x = "topright", legend = rownames(spyder_df[-c(1,2),]), horiz = FALSE,
    bty = "n", pch = 20 , col = c("#800080", "#EE82EE", "#4B0082","#FF00FF"),
    text.col = "black", cex = 0.8, pt.cex = 1.5
)
par(op)

# Define colors and titles
colors <- c("#800080", "#EE82EE", "#4B0082","#FF00FF")
titles <- rownames(spyder_df[-c(1,2),])

# Reduce plot margin using par()
# Split the screen in 3 parts
op <- par(mar = c(1, 1, 1, 1))
par(mfrow = c(2,2))

# Create the radar chart
for(i in 1:4){
    create_beautiful_radarchart(
```

```
    data = spyder_df[c(1, 2, i+2), ],
    color = colors[i], title = titles[i]
     )
}
par(op)

#=================================================================
================================#

#### Get mean of Obesity and Undernourished for different clusters

df$cluster = as.factor(km.4$cluster)
Obesity_mean_cluster_1 = mean(df[df$cluster == 1,"Obesity"])
Undernourished_mean_cluster_1 = mean(df[df$cluster == 1,"Undernourished"])

Obesity_mean_cluster_2 = mean(df[df$cluster == 2,"Obesity"])
Undernourished_mean_cluster_2 = mean(df[df$cluster == 2,"Undernourished"])

Obesity_mean_cluster_3 = mean(df[df$cluster == 3,"Obesity"])
Undernourished_mean_cluster_3 = mean(df[df$cluster == 3,"Undernourished"])

Obesity_mean_cluster_4 = mean(df[df$cluster == 4,"Obesity"])
Undernourished_mean_cluster_4 = mean(df[df$cluster == 4,"Undernourished"])
```