Data Mining - Selection and Preparation of a Dataset

Dec 2023

Contents

Introduction

Obesity has emerged as a global health challenge with significant implications for public health. The prevalence of obesity has risen dramatically over the past few decades, leading to severe health consequences such as cardiovascular diseases, diabetes, and various metabolic disorders. Understanding and addressing obesity is crucial for developing preventive measures and personalized intervention strategies. In the context of our research, we focus on the application area of health and well-being, where the accurate classification of obesity levels can facilitate early detection and targeted interventions. Obesity poses a significant challenge to public health, with its prevalence steadily increasing over the past few decades. According to the World Health Organization (WHO), obesity has more than doubled worldwide since 1980. In 2016, 39% of adults aged 18 years and over were overweight, and 13% were obese. These statistics underscore the urgency of addressing obesity through advanced analytical approaches to mitigate its impact on public health.

The primary objective of this research is to develop a robust and accurate classification model using a Neural Network algorithm for categorizing gender into different obesity levels. By employing this analysis, we seek to contribute to the field of health informatics and provide a valuable tool for healthcare professionals to identify and address obesity-related concerns promptly.

The selection of a Neural Network algorithm for the classification of obesity levels is motivated by its interpretability, simplicity, and efficiency in handling both categorical and numerical data. Neural networks provide a transparent and intuitive framework for decision-making, enabling healthcare professionals to understand the factors influencing the classification outcomes. Additionally, Neural networks are well-suited for handling complex relationships between input features and the target variable. Given the multi-faceted nature of obesity, where multiple factors contribute to its classification, the Neural Network approach allows us to capture non-linear interactions and hierarchies within the data.

Data Selection

The data contains numerical data and continuous data, so it can be used for analysis based on algorithms of classification, and prediction. The data consist of the estimation of obesity levels in people from the countries of Mexico, Peru and Colombia, with ages between 14 and 61 and diverse eating habits and physical condition, data was collected using a web platform with a survey where anonymous users answered each question, then the information was processed obtaining 17 attributes and 2111 records.

Attribute Information:

The attributes related with eating habits are:
FAVC: Frequent consumption of high caloric food, FCVC: Frequency of consumption of vegetables, NCP: Number of main meals, CAEC: Consumption of food between meals, CH20: Consumption of water daily, CALC: Consumption of alcohol.

The attributes related with the physical condition are:
SCC: Calories consumption monitoring, FAF: Physical activity frequency, TUE: Time using technology devices, MTRANS: Transportation used.

Variables obtained:
Gender, Age, Height and Weight.

Obesity values are:
Underweight: Less than 18.5
Normal: 18.5 to 24.9
Overweight: 25.0 to 29.9.
Obese Obesity I: 30.0 to 34.9
Obesity II: 35.0 to 39.9
Obesity III: Higher than 40.

Exploratory Data Analysis

Required libraries

```
library(DataExplorer)
library(tidyverse)
library(ggplot2)
library(tidyr)
library(viridis)
library(plotly)
library(PerformanceAnalytics)
library(factoextra)
library(nnet)
library(NeuralNetTools)
library(gmodels)
library(caret)
```

Using the 'str' function, it was observed that the variables in the dataset consist of characters and numeric.

```
str(data)
```

```
## 'data.frame':    2111 obs. of  17 variables:
##  $ Gender                        : chr  "Female" "Female" "Male" "Male" ...
##  $ Age                           : num  21 21 23 27 22 29 23 22 24 22 ...
##  $ Height                        : num  1.62 1.52 1.8 1.8 1.78 1.62 1.5 1.64 1.78 1.72 ...
##  $ Weight                        : num  64 56 77 87 89.8 53 55 53 64 68 ...
##  $ family_history_with_overweight: chr  "yes" "yes" "yes" "no" ...
##  $ FAVC                          : chr  "no" "no" "no" "no" ...
##  $ FCVC                          : num  2 3 2 3 2 2 3 2 3 2 ...
##  $ NCP                           : num  3 3 3 3 1 3 3 3 3 3 ...
##  $ CAEC                          : chr  "Sometimes" "Sometimes" "Sometimes" "Sometimes" ...
##  $ SMOKE                         : chr  "no" "yes" "no" "no" ...
##  $ CH2O                          : num  2 3 2 2 2 2 2 2 2 2 ...
##  $ SCC                           : chr  "no" "yes" "no" "no" ...
##  $ FAF                           : num  0 3 2 2 0 0 1 3 1 1 ...
##  $ TUE                           : num  1 0 1 0 0 0 0 0 1 1 ...
##  $ CALC                          : chr  "no" "Sometimes" "Frequently" "Frequently" ...
##  $ MTRANS                        : chr  "Public_Transportation" "Public_Transportation" "Pu
blic_Transportation" "Walking" ...
##  $ NObeyesdad                    : chr  "Normal Weight" "Normal Weight" "Normal Weight" "Ov
erweight" ...
```

The ‘summary’ function was used to obtain the minimum and maximum, as well as measures of central tendency (mean, median) and spread (1st and 3rd quartiles) for each of these variables.

```
summary(data)
```

```
##     Gender               Age            Height          Weight      
##  Length:2111        Min.   :14.00   Min.   :1.450   Min.   : 39.00  
##  Class :character   1st Qu.:19.95   1st Qu.:1.630   1st Qu.: 65.47  
##  Mode  :character   Median :22.78   Median :1.700   Median : 83.00  
##                     Mean   :24.31   Mean   :1.702   Mean   : 86.59  
##                     3rd Qu.:26.00   3rd Qu.:1.768   3rd Qu.:107.43  
##                     Max.   :61.00   Max.   :1.980   Max.   :173.00  
##  family_history_with_overweight     FAVC                FCVC      
##  Length:2111                    Length:2111        Min.   :1.000  
##  Class :character               Class :character   1st Qu.:2.000  
##  Mode  :character               Mode  :character   Median :2.386  
##                                                    Mean   :2.419  
##                                                    3rd Qu.:3.000  
##                                                    Max.   :3.000  
##       NCP            CAEC              SMOKE                CH2O      
##  Min.   :1.000   Length:2111        Length:2111        Min.   :1.000  
##  1st Qu.:2.659   Class :character   Class :character   1st Qu.:1.585  
##  Median :3.000   Mode  :character   Mode  :character   Median :2.000  
##  Mean   :2.686                                         Mean   :2.008  
##  3rd Qu.:3.000                                         3rd Qu.:2.477  
##  Max.   :4.000                                         Max.   :3.000  
##      SCC                 FAF              TUE              CALC          
##  Length:2111        Min.   :0.0000   Min.   :0.0000   Length:2111       
##  Class :character   1st Qu.:0.1245   1st Qu.:0.0000   Class :character  
##  Mode  :character   Median :1.0000   Median :0.6253   Mode  :character  
##                     Mean   :1.0103   Mean   :0.6579                     
##                     3rd Qu.:1.6667   3rd Qu.:1.0000                     
##                     Max.   :3.0000   Max.   :2.0000                     
##     MTRANS           NObeyesdad       
##  Length:2111        Length:2111       
##  Class :character   Class :character  
##  Mode  :character   Mode  :character  
```

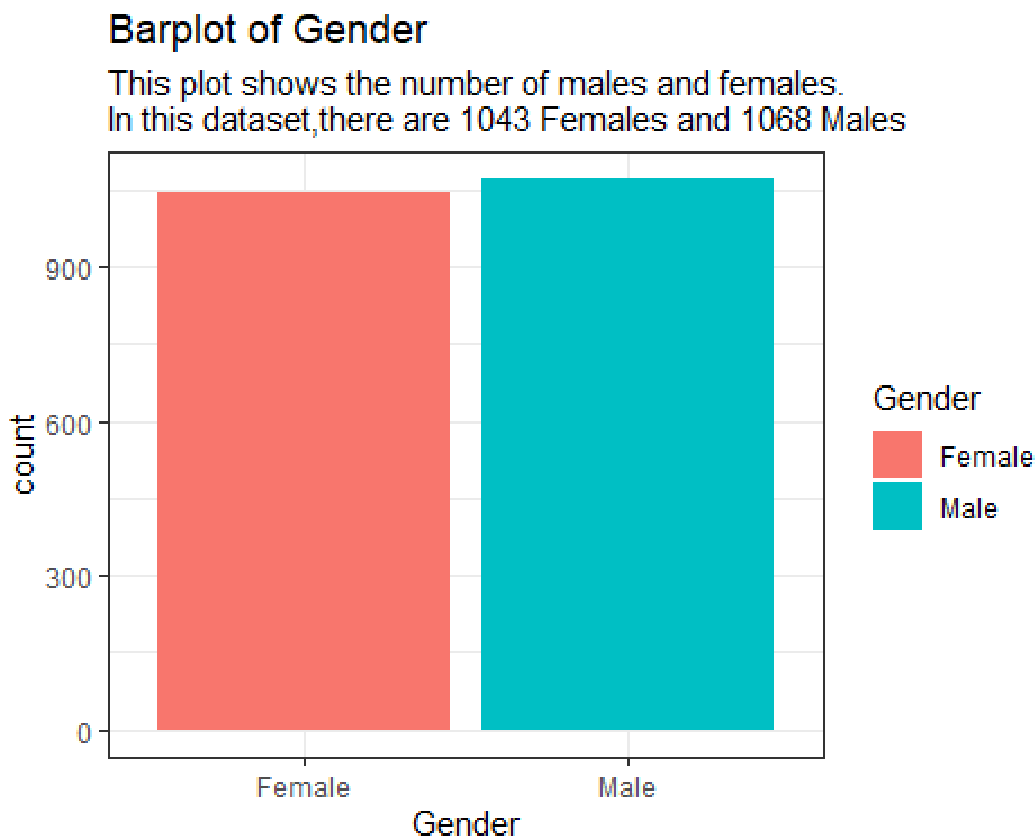
Converting the character data types to categorical

```
data$Gender <- as.factor(data$Gender)
data$family_history_with_overweight <-
as.factor(data$family_history_with_overweight)
data$FAVC <- as.factor(data$FAVC)
data$CAEC <- as.factor(data$CAEC)
data$SMOKE  <- as.factor(data$SMOKE)
data$SCC <- as.factor(data$SCC)
data$CALC <- as.factor(data$CALC)
data$MTRANS <- as.factor(data$MTRANS)
data$NObeyesdad <- as.factor(data$NObeyesdad)
```

Changing Height metric from Meter to Centimeter by multiply the value with 100

```
data$Height <- data$Height *100
```

```
ggplot(data, aes(x=Gender, fill = Gender))+
  theme_bw()+
  geom_bar()+
labs(title = "Barplot of Gender", subtitle = "This plot shows the number of m
ales and females.
In this dataset,there are 1043 Females and 1068 Males")
```



This dataset contains information on 1068 males(average height of approximately 176cm, average weight of 90.8kg) and 1043 females(average height of 164cm, average weight of 82.3kg). The average age of both males and females is approximately 24 years.

```
# the average age, weight, and height of the males and females in the dataset
their total number
data %>% group_by(Gender) %>% summarise("Gender Count"=n(),
                                         "Average Height"=mean(Height),
                                         "Average Weight"=mean(Weight),
                                          "Average Age"=mean(Age
```

```
## # A tibble: 2 × 5
##   Gender `Gender Count` `Average Height` `Average Weight` `Average Age`
##   <fct>           <int>            <dbl>            <dbl>         <dbl>
## 1 Female           1043             164.             82.3          24.0
## 2 Male             1068             176.             90.8          24.6
```

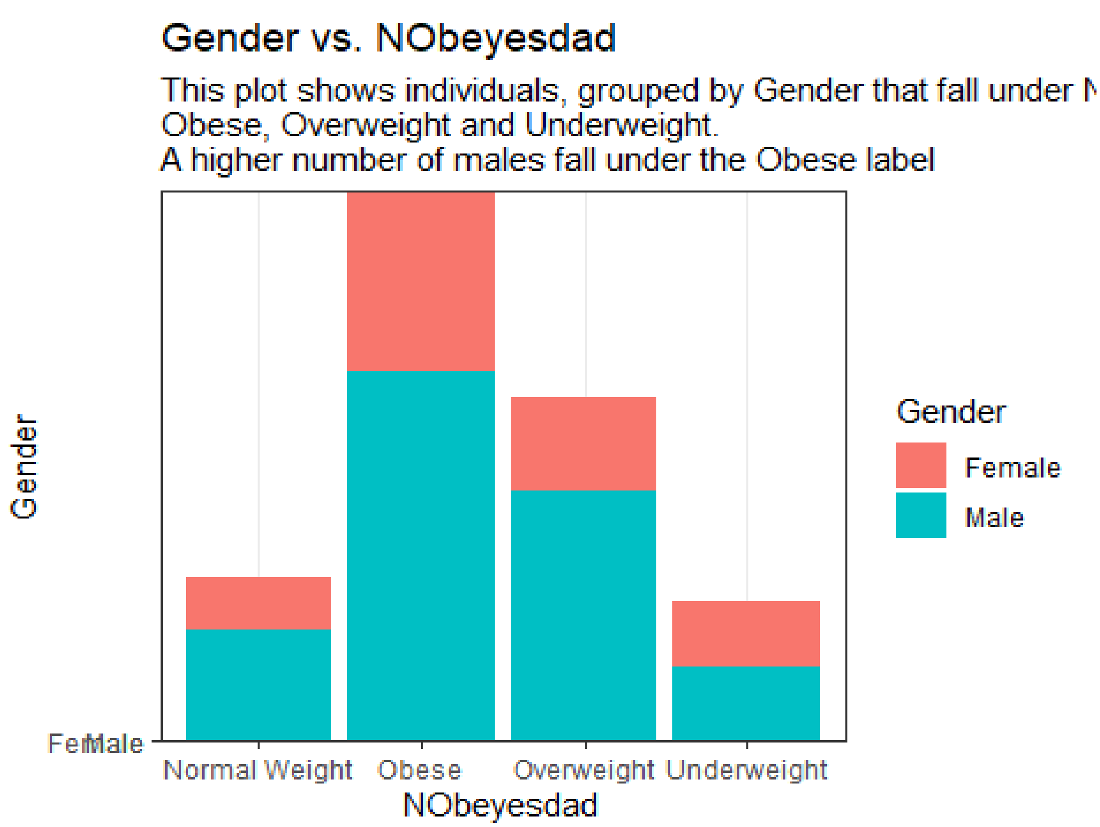
287 out of 2111 individuals are Normal Weight, 141 of which are Females and 146 Males.

```
# Filter the dataset when Label = Normal Weight. Name this filtered data "Nor
mal Weight"
Normal_Weight<- data %>% filter(NObeyesdad=="Normal Weight")%>% drop_na()
Normal_Weight%>% group_by(Gender)%>% summarise("Gender Count"=n())
```

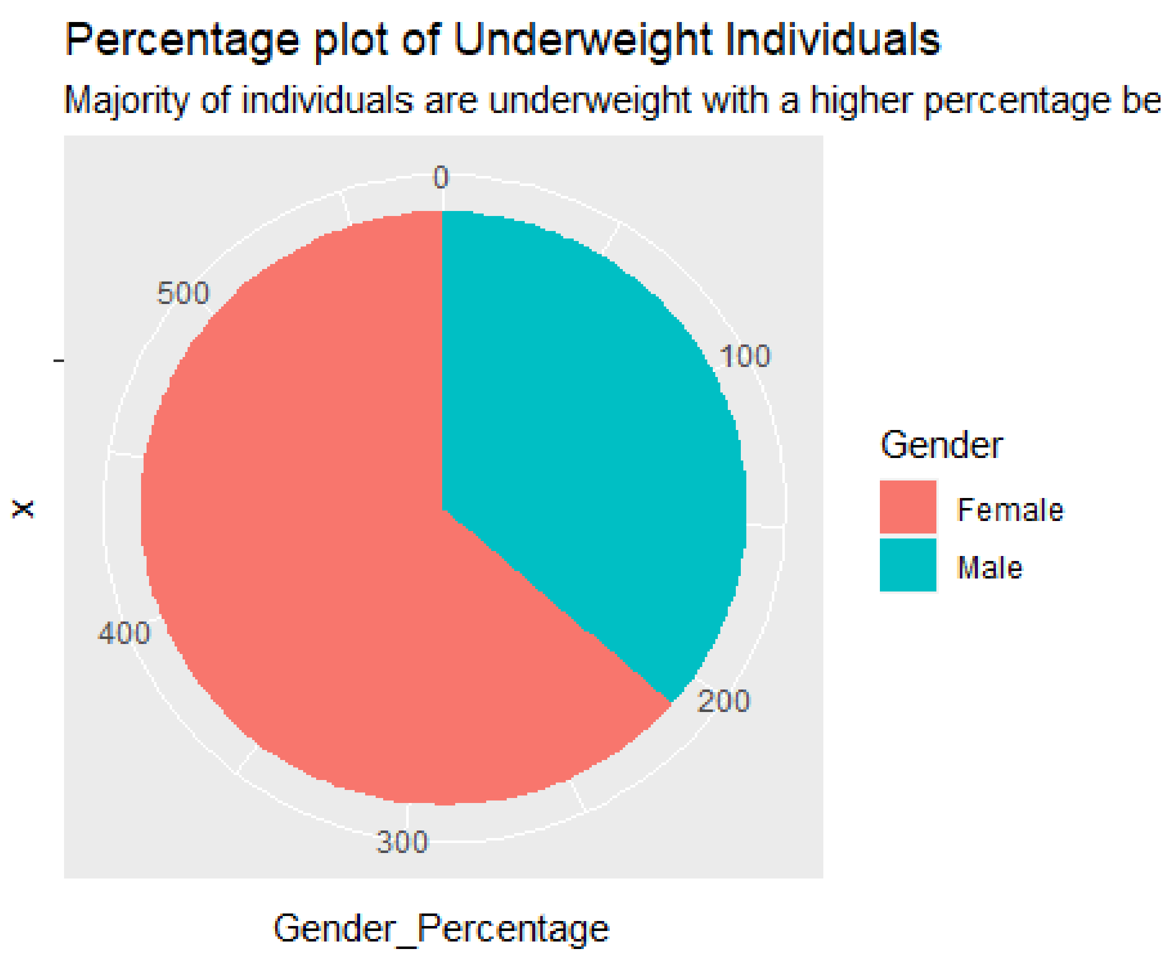
```
## # A tibble: 2 × 2
##   Gender `Gender Count`
##   <fct>           <int>
## 1 Female            141
## 2 Male              146
```

Classification of the Obesity Levels

```
ggplot(data=data)+
  geom_col(mapping=aes(fill=Gender, x=NObeyesdad, y=Gender), position="stack"
)+
   theme_bw()+
labs(title="Gender vs. NObeyesdad", subtitle="This plot shows individuals, gr
ouped by Gender that fall under Normal Weight,
Obese, Overweight and Underweight.
A higher number of males fall under the Obese label")
```



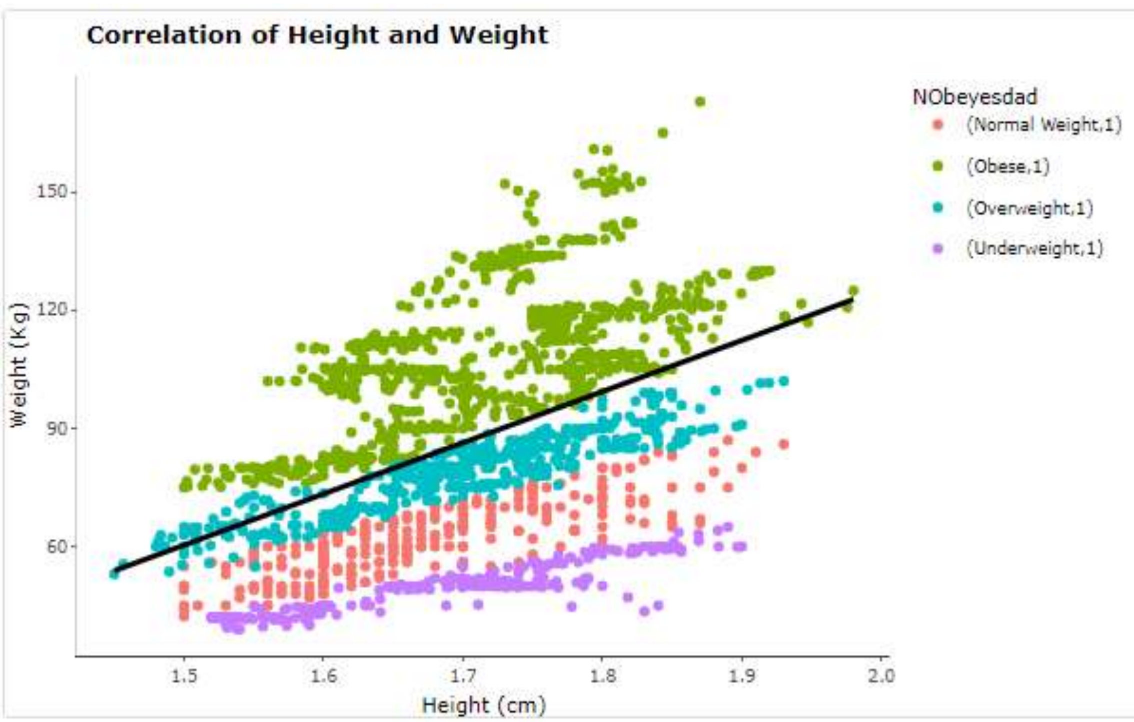
```
Underweight<- data %>% filter(NObeyesdad=="Underweight")%>% drop_na()
Underweight_Gender_Percentage<- Underweight %>% group_by(Gender) %>%
  summarise("Gender_Percentage"=(n()*100)/47)

annotation <- data.frame(
  x = c(""),
  y = c(15,50),
  label = c(" ", " "))

ggplot(data=Underweight_Gender_Percentage)+
  geom_bar(mapping=aes(x="", y=Gender_Percentage,fill=Gender),
                           stat="identity",width=1)+
                 coord_polar("y", start=0)+
labs(title="Percentage plot of Underweight Individuals", subtitle= "Majority
of individuals are underweight with a higher percentage being females.")+
geom_text(data=annotation, aes( x=x, y=y, label=label),                 ,
            color="Black",
            size=5 , angle=0, fontface="bold" )
```



Correlation Between Height and Weight in Type of Obesity

It can be deduced from below that the correlation (0.46) between height and weight is weakly positive.

```
obesity_cor <- data %>%
  select(c(NObeyesdad, Height, Weight))

plotob_cor <- ggplot(data = obesity_cor, mapping = aes(x = Height, y = Weight
, col = NObeyesdad))+
  geom_point(aes(col = NObeyesdad))+
  geom_smooth(method=lm , color="black", se=FALSE, formula = y~x) +
  scale_fill_viridis(discrete = T, option = "C") +
  labs(title = list(text = paste0('Correlation of Height and Weight')),
    x = "Height (cm)",
    y = "Weight (Kg)"
  ) +
  theme(legend.title = element_blank(),
        plot.title = element_text(face = "bold"),
        panel.background = element_rect(fill = "#ffffff"),
        axis.line.y = element_line(colour = "grey"),
        axis.line.x = element_line())
ggplotly(plotob_cor, tooltip = "text")
```



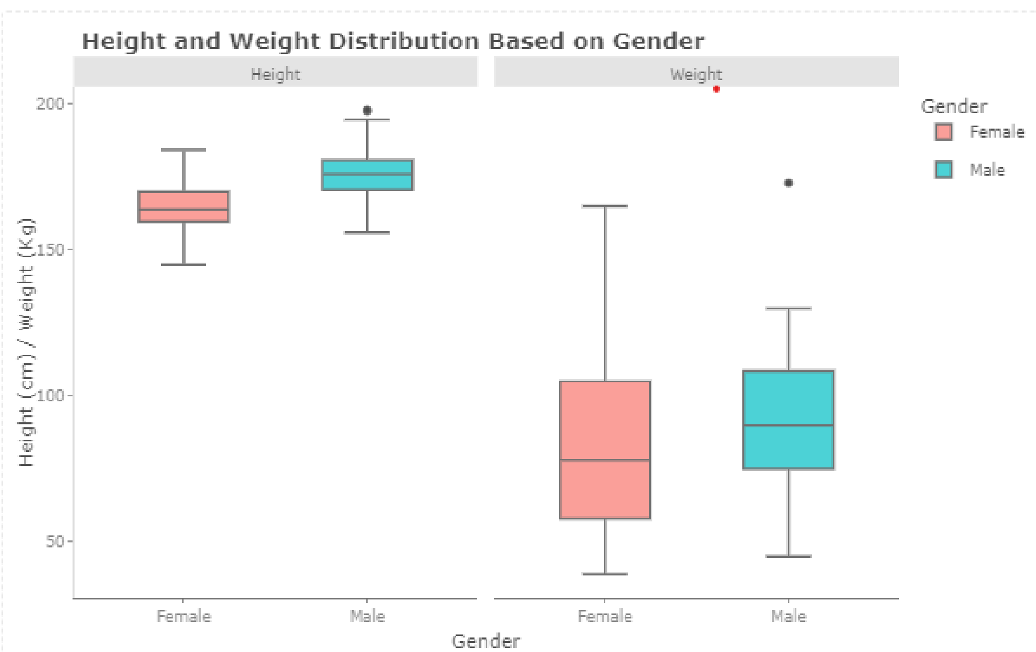

```
ob_corr <- cor(obesity_cor$Height, obesity_cor$Weight)
ob_corr
```

```
## [1] 0.4631361
```

Height and Weight Distribution based on Gender

The box plots show the distribution of Height and Weight based on Gender wise. The plots highlight that the median height of females in the sample is significantly lower than that of males, with a few of males surpassing 1.98 meters (outliers). In terms of their weights, though, the difference is not as significant. While, one individual with a weight of more than 165 kg is considered an outlier.

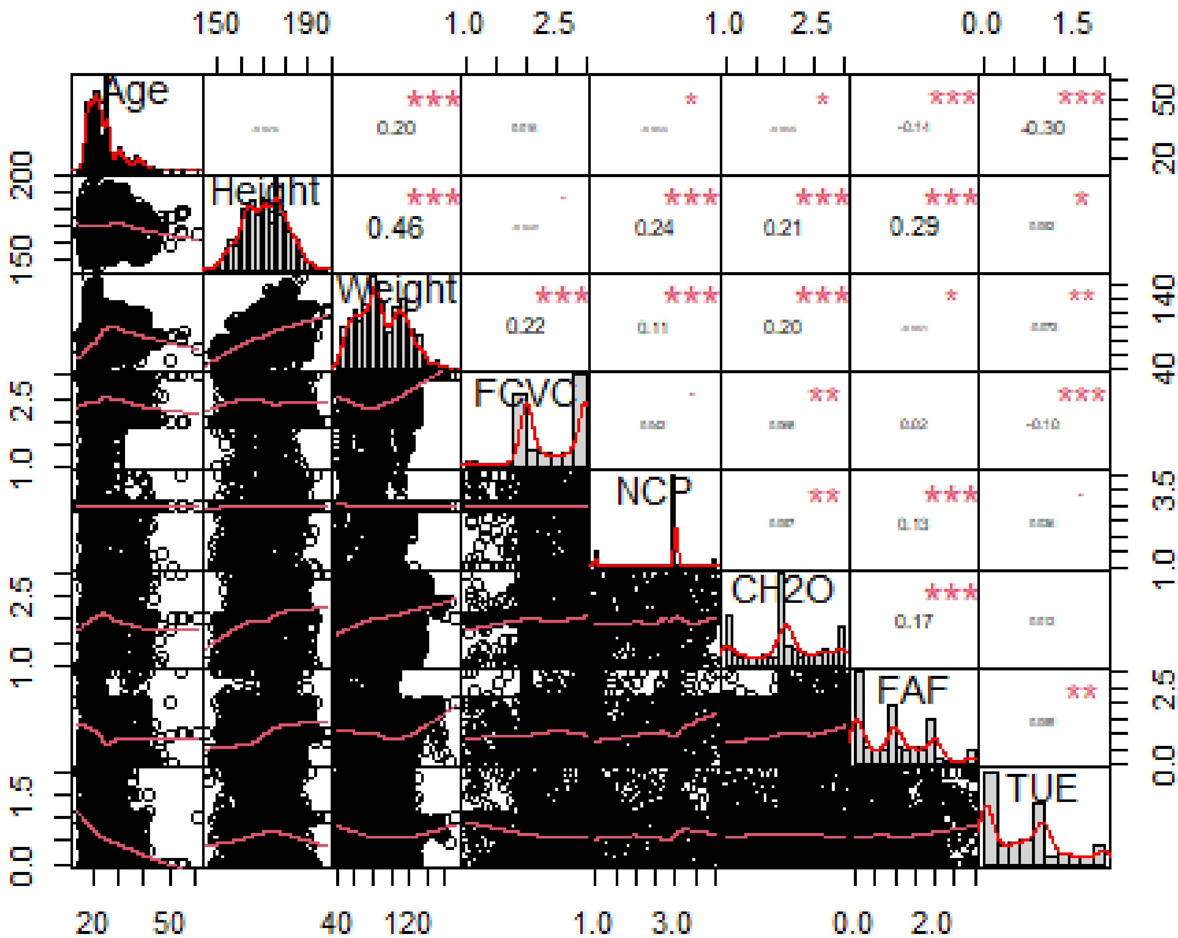
```
height_weight <- data %>%
  select(c(Gender, Height, Weight))
height_weight <- pivot_longer(data = height_weight,
                              cols = c("Height","Weight"),
                              names_to = "variabel")

plothw <- ggplot(data = height_weight, mapping = aes(x = Gender, y = value))+
  geom_boxplot(aes(fill=Gender), position = "dodge")+
  facet_wrap(vars(variabel)) + #memisahkan plot berdasarkan variable paramete
r
  labs(title = list(text = paste0('Height and Weight Distribution Based on Ge
nder')),
    x = "Gender",
    y = "Height (cm) / Weight (Kg)"
  ) +
  theme(legend.title = element_blank(),
        plot.title = element_text(face = "bold"),
        panel.background = element_rect(fill = "#ffffff"),
        axis.line.y = element_line(colour = "grey"),
        axis.line.x = element_line())

ggplotly(plothw, tooltip = "text")
```



Correlation Between the Numeric Variables

No record of very strong correlation between the numeric variables

```
chart.Correlation(data[,c(2,3,4,7,8,11,13,14)],histogram=TRUE, col="grey10", pch=1, main="Correlation")
```



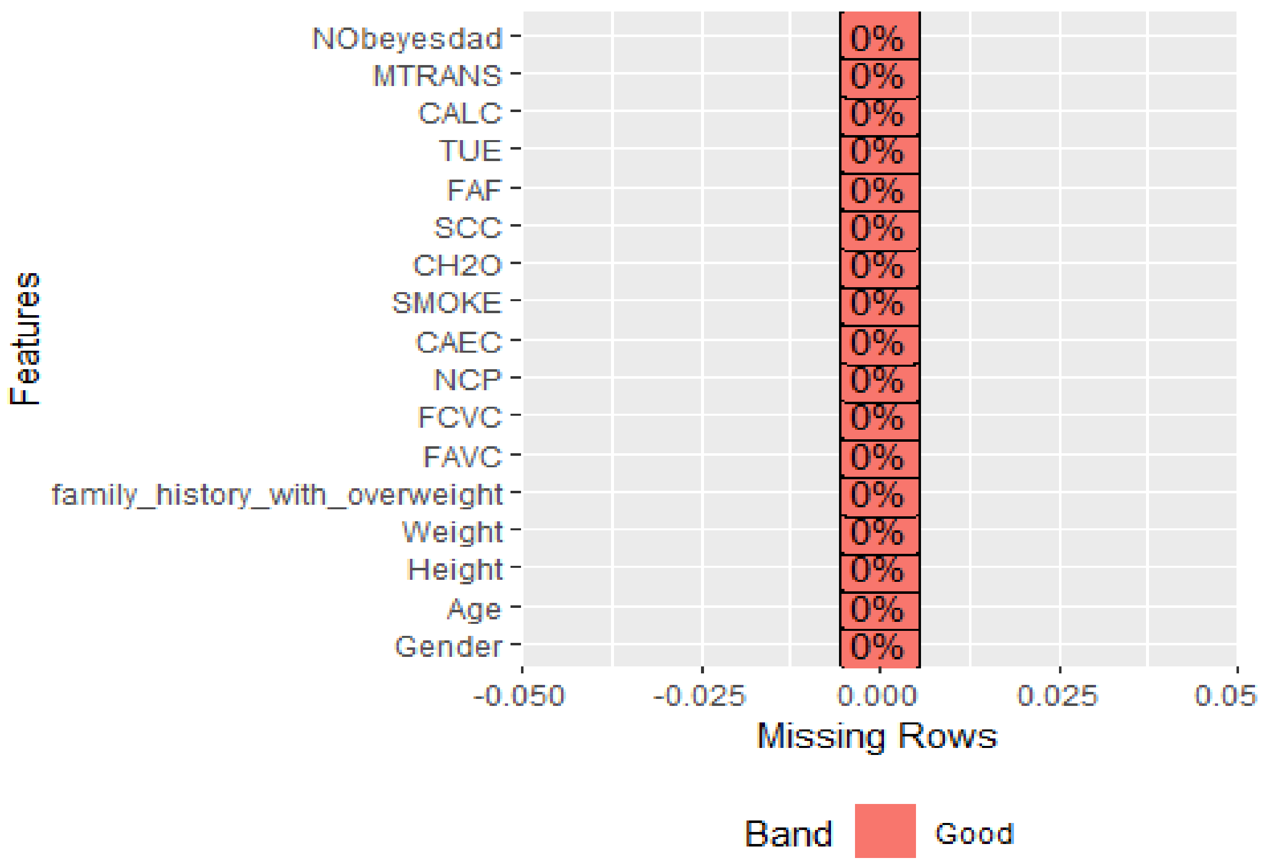
Preprocessing

We have six levels of obesity in the dataset, while there can be only four, hence, they are Underweight, Normal, Overweight and Obese. No spelling errors in the dataset.

Missing Values

There are no records of missing data, NA, in the dataset as shown below

```
plot_missing(data)
```



Principal Component Analysis (PCA)

Principal Component Analysis (PCA) can lead to the following issues when dealing with an excessive number of variables:

Increased computer throughput, resulting in computational challenges. Complications in visualizing complex data. Reduced efficiency due to the inclusion of variables that do not contribute to the analysis. Difficulty in interpreting data due to its increased complexity.

PCA employs standardized data to mitigate the impact of scale differences and prevent data distortion.

```
data_pca <- transform(data[, -1])
data_pca <- data_pca[, -4]
data_pca <- data_pca[, -4]
data_pca <- data_pca[, -6]
data_pca <- data_pca[, -6]
data_pca <- data_pca[, -7]
data_pca <- data_pca[, -9]
data_pca <- data_pca[, -9]
data_pca <- data_pca[, -9]
data_pca
```

```
##           Age   Height    Weight     FCVC      NCP     CH2O      FAF      TUE
## 1    21.00000 162.0000  64.00000 2.000000 3.000000 2.000000 0.000000 1.000000
## 2    21.00000 152.0000  56.00000 3.000000 3.000000 3.000000 3.000000 0.000000
## 2101 25.77756 162.8205 107.37870 3.000000 3.000000 2.506631 0.025787 0.484165
## 2102 25.72200 162.8470 107.21895 3.000000 3.000000 2.487070 0.067329 0.455823
## 2103 25.76563 162.7839 108.10736 3.000000 3.000000 2.320068 0.045246 0.413106
## 2104 21.01685 172.4268 133.03352 3.000000 3.000000 1.650612 1.537639 0.912457
## 2105 21.68237 173.2383 133.04394 3.000000 3.000000 1.610768 1.510398 0.931455
## 2106 21.28597 172.6920 131.33579 3.000000 3.000000 1.796267 1.728332 0.897924
## 2107 20.97684 171.0730 131.40853 3.000000 3.000000 1.728139 1.676269 0.906247
## 2108 21.98294 174.8584 133.74294 3.000000 3.000000 2.005130 1.341390 0.599270
## 2109 22.52404 175.2206 133.68935 3.000000 3.000000 2.054193 1.414209 0.646288
## 2110 24.36194 173.9450 133.34664 3.000000 3.000000 2.852339 1.139107 0.586035
## 2111 23.66471 173.8836 133.47264 3.000000 3.000000 2.863513 1.026452 0.714137
```

In the results of PCA, the cumulative proportion from PC1 to PC6 is about 87.3% (above 85%). It means that PC1~PC6 can explain 87% of the whole data.

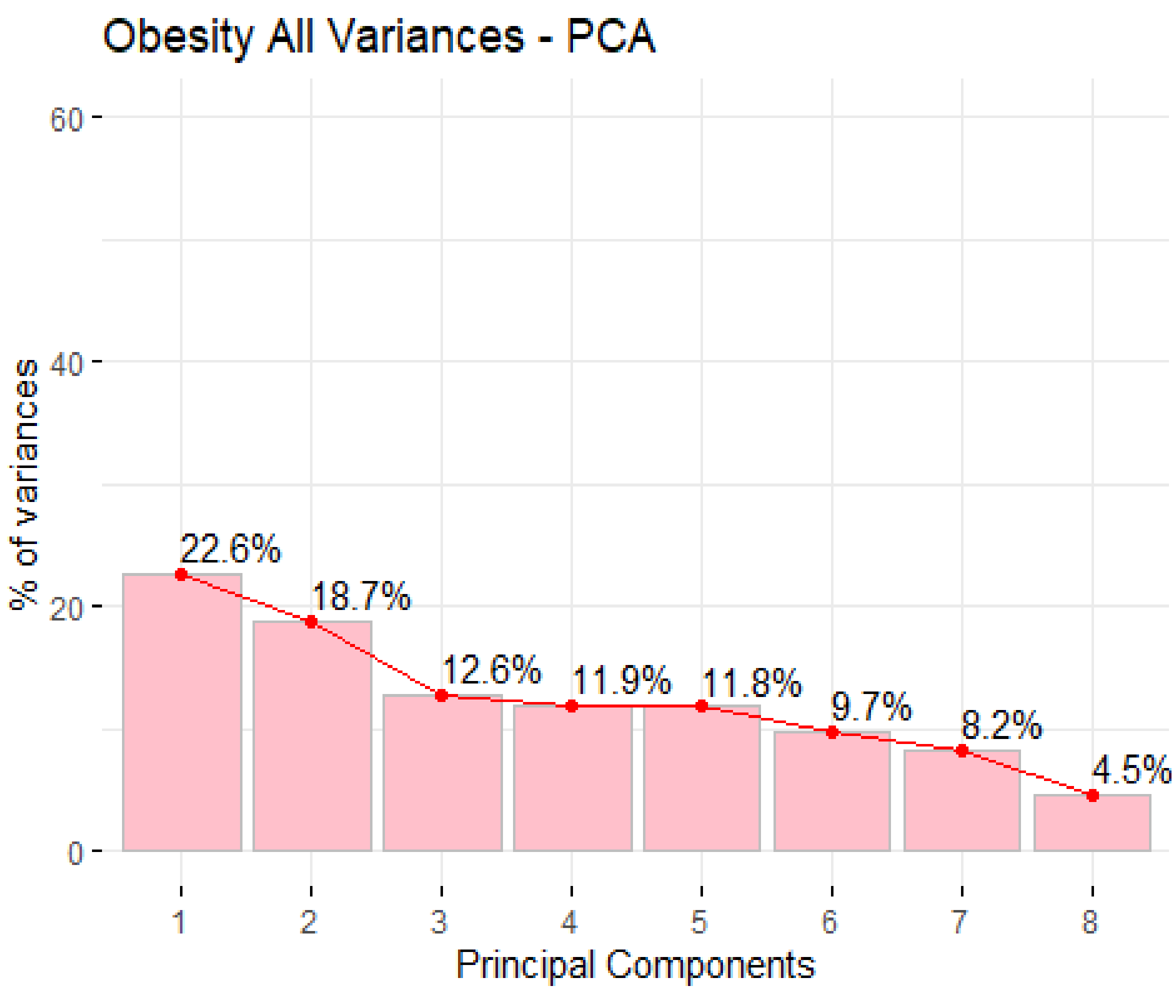
```
all_pca <- prcomp(data_pca, cor=TRUE, scale = TRUE)
summary(all_pca)
```

```
## Importance of components:
##                           PC1    PC2    PC3    PC4    PC5     PC6     PC7
## Standard deviation     1.3461 1.2217 1.0058 0.9751 0.9699 0.87965 0.80967
## Proportion of Variance 0.2265 0.1866 0.1265 0.1189 0.1176 0.09672 0.08195
## Cumulative Proportion  0.2265 0.4131 0.5395 0.6584 0.7760 0.87268 0.95463
##                            PC8
## Standard deviation     0.60246
## Proportion of Variance 0.04537
## Cumulative Proportion  1.00000
```

Screeplot

The percentage of variability explained by the principal components can be ascertained through screeplot. Line lies at point PC6.

```
fviz_eig(all_pca, addlabels=TRUE, ylim=c(0,60), geom = c("bar", "line"), barf
ill = "pink", barcolor="grey",linecolor = "red", ncp=10)+
labs(title = "Obesity All Variances - PCA",
         x = "Principal Components", y = "% of variances")
```

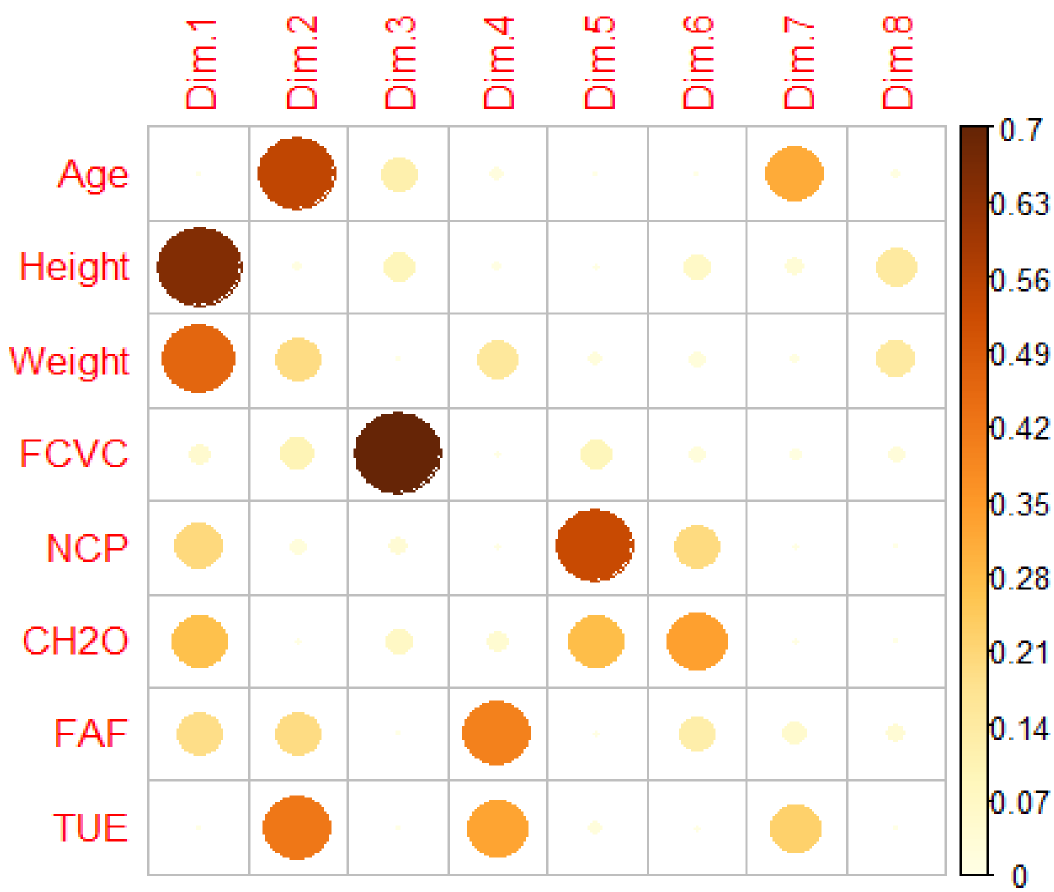


The PCA Variables

```
all_var <- get_pca_var(all_pca)
all_var

## Principal Component Analysis Results for variables
##  ===================================================
##   Name       Description
## 1 "$coord"   "Coordinates for the variables"
## 2 "$cor"     "Correlations between variables and dimensions"
## 3 "$cos2"    "Cos2 for the variables"
## 4 "$contrib" "contributions of the variables"
```

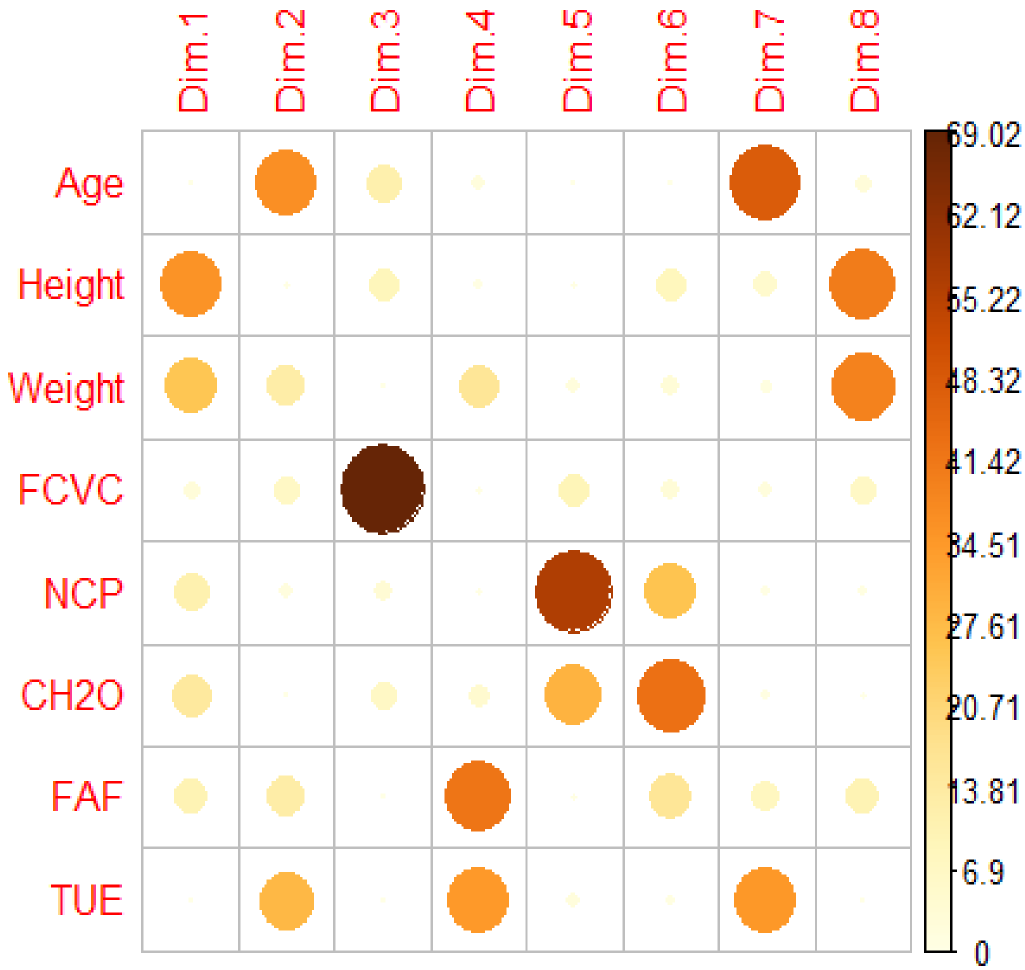
Quality of representation of PCA Correlation between variables and PCA

```
corrplot(all_var$cos2, is.corr=FALSE)
```



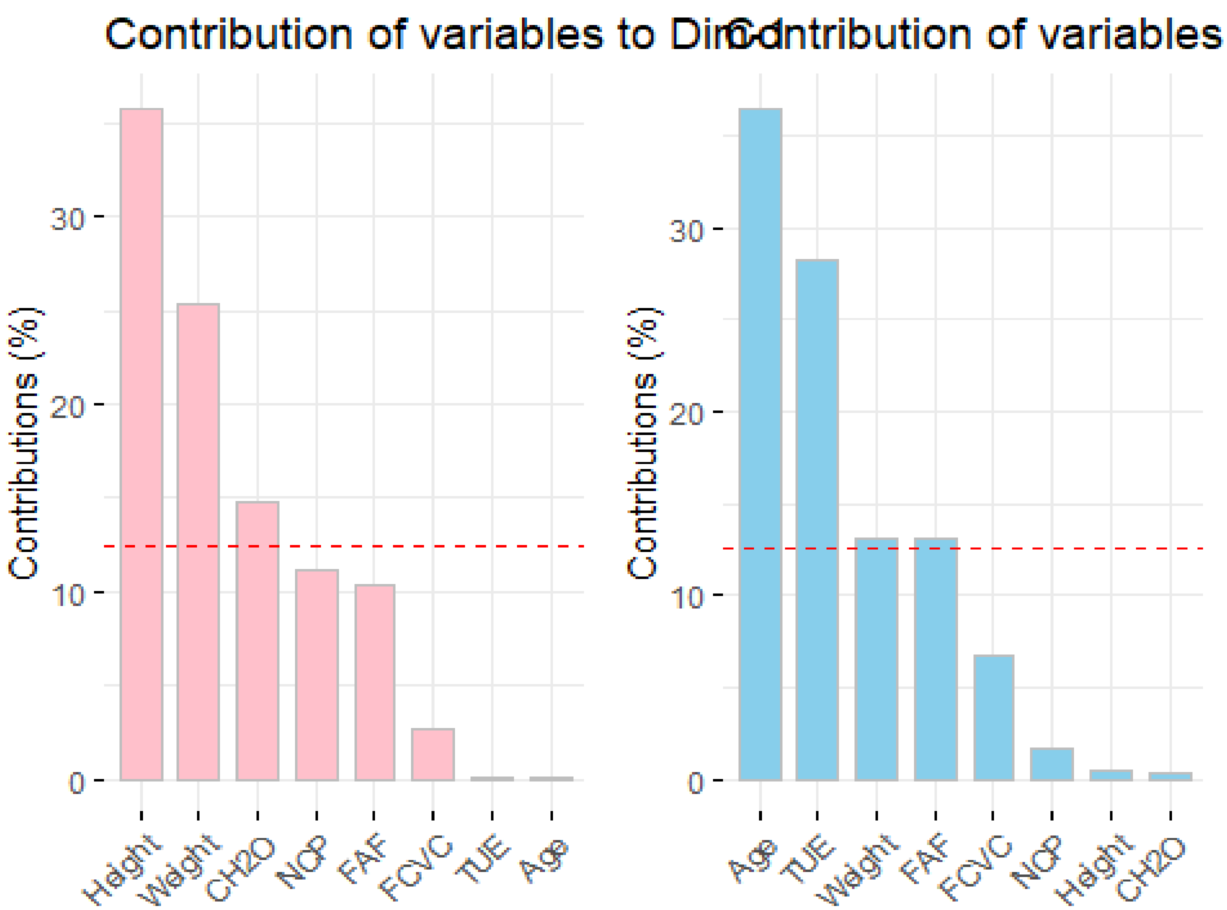
Contributions of variables to PCA To highlight the most contributing variables for each component

```
corrplot(all_var$contrib, is.corr=FALSE)
```



Contributions of variables to PC1 & PC2

```
p1 <- fviz_contrib(all_pca, choice="var", axes=1, fill="pink", color="grey",
top=10)
p2 <- fviz_contrib(all_pca, choice="var", axes=2, fill="skyblue", color="grey
", top=10)
grid.arrange(p1,p2,ncol=2)
```

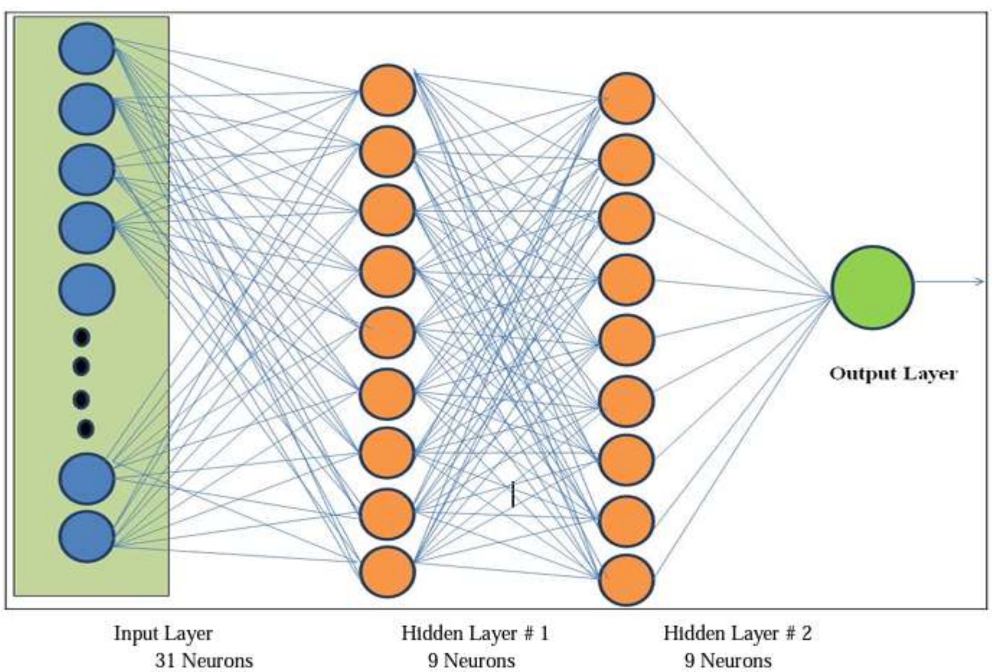


Algorithm Intuition

Neural Network

Neural networks (NN), a branch of machine learning also known as artificial neural networks (ANN), are computational models—essentially algorithms. These networks possess a remarkable capacity to derive meaning from imprecise or complex data, uncovering intricate patterns and identifying trends that may elude human comprehension or other conventional computer techniques. Neural networks have revolutionized our daily lives in numerous ways, exemplified by their integration into ridesharing apps, Gmail's intelligent email sorting, and product recommendations on platforms like Amazon.

One of the most groundbreaking features of neural networks is their ability to learn autonomously. These characteristic parallels the human brain, which comprises neurons—the fundamental units for transmitting information in both biological brains and neural networks. Alex Cardinell, Founder and CEO of Cortx, an artificial intelligence company specializing in natural language processing solutions, including an automated grammar correction application called Perfect Tense, points out, Human brains and artificial neural networks share similarities in their learning processes. In both cases, neurons continuously adjust their responses based on stimuli. When a task is executed correctly, neurons receive positive feedback and become more likely to trigger in similar future instances. Conversely, if neurons receive negative feedback, they learn to be less likely to trigger in subsequent instances. The whole of neurons in the input layer of the NN is equal to the number of characteristics in the dataset in its architecture. The hidden layer is another network component, with the number of hidden layers being counted as one layer. The NN architecture is illustrated below.



Data Splitting
Splitting the data set into Train and Test

A 60/40 train/test split is used to divide the dataset samples into training and testing sets. The rationale behind using this train/test split is due to the fact this is the more commonly used train/test split in machine learning.

```
index <- sample(2, nrow(data), replace=TRUE, prob = c(0.60, 0.40))
traindata <- data[index==1, ]
testdata <- data[index==2, ]
```

Model Fitting
Obesity Classification using Neural Network

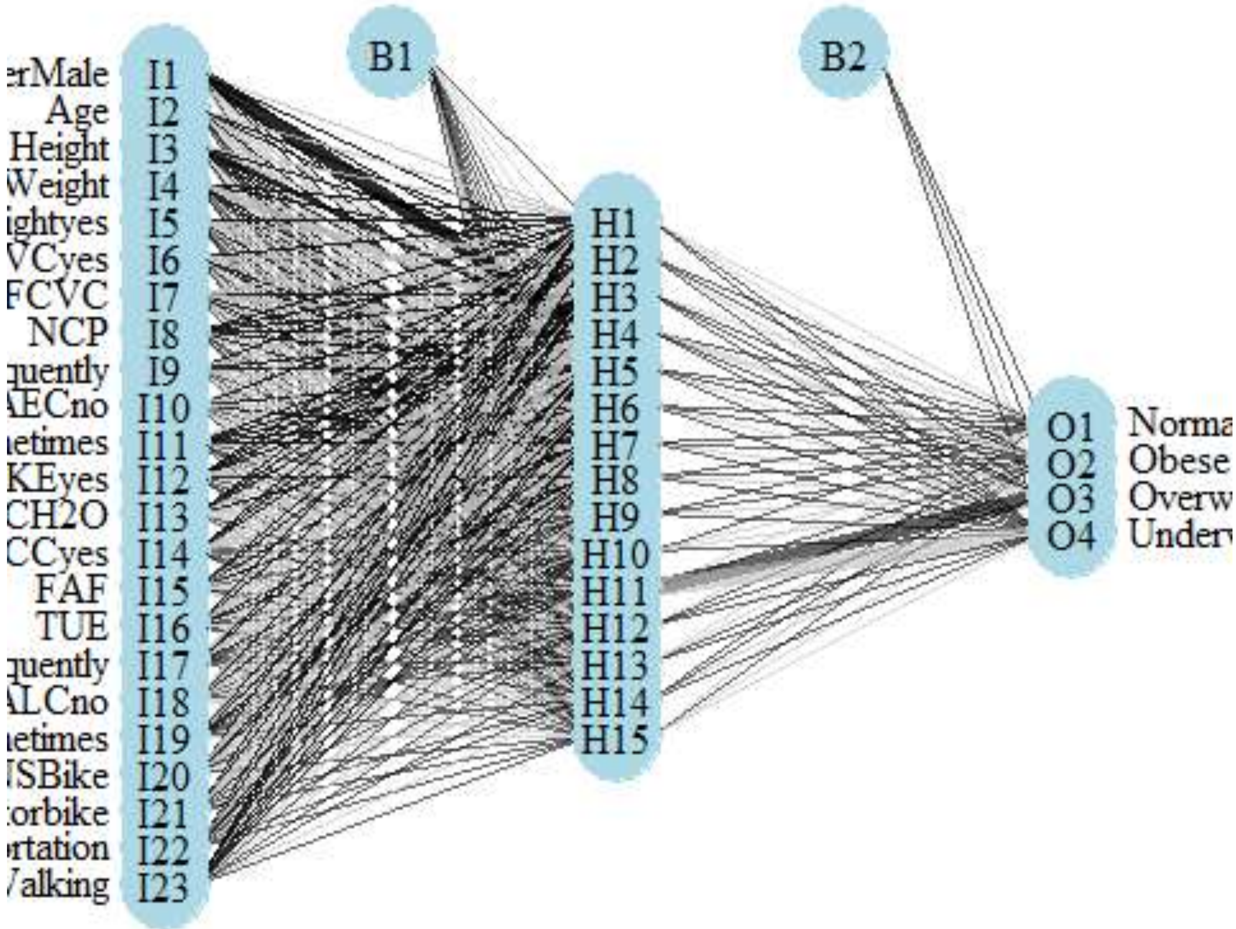
The numerical variables were fed into the machine learning model. The choice of machine learning classifiers is very important and plays an important role in classifying the output classes. A well-known machine learning classifier is implemented to investigate which classifier performs the best in classifying the different obesity levels. Thw classifier used is artificial neural network (ANN)

```
model_nnet <- nnet(NObeyesdad ~ ., data = traindata, size=15, rang = 1, decay
= 8e-4, maxit = 200)
```

```
## # weights:  424
## initial  value 2495.858569
## iter  10 value 1075.078740
## iter  20 value 866.988095
## iter  30 value 814.640830
## iter  40 value 812.659188
## iter  50 value 811.275568
## iter  60 value 547.133508
## iter  70 value 466.826987
## iter  80 value 382.994563
## iter  90 value 364.986935
## iter 100 value 361.176133
## iter 110 value 350.145225
## iter 120 value 346.736428
## iter 130 value 336.161362
## iter 140 value 330.976204
## iter 150 value 323.903893
## iter 160 value 316.425693
## iter 170 value 302.553769
## iter 180 value 217.952875
## iter 190 value 181.674827
## iter 200 value 155.234909
## final  value 155.234909
## stopped after 200 iterations
```

Neural Network Algorithm for Type of Obesity

```r
par(mar = numeric(4), family = 'serif')
plotnet(model_nnet, alpha = 0.6)
```



Prediction and Model Evaluation

```r
pred_nnet <- predict(model_nnet, testdata,type = c("class"))
```

After experimenting with the selected parameters in the training process and system testing, the accuracy level of the model is 96%, and the error is 4% i.e 96% of the test values are correctly classified, and misclassification rate is around 4%. The rate at which there was no information, i.e., “No Information Rate” is 45%.

Confusion matrix and Model Properties

The confusion matrix, which is often used to evaluate the performance of a classification model is provided below. In a confusion matrix, the rows represent the predicted classes, while the columns represent the actual classes or categories. Each cell in the matrix represents the number of observations that fall into a particular combination of actual and predicted obesity types.

		Predicted classes			
Classified as		Normal weight	Obese	Overweight	Underweight
Actual classes	Normal weight	113	0	1	2
	Obese	0	388	18	0
	Overweight	5	8	205	0
	Underweight	9	0	0	113

In this case, the model correctly classified 113 persons to have normal weight (True Positives), 388 to be obesed (True Negatives), with no false positive errors and negative errors.

```
pred = factor(pred_nnet, levels = c('Normal Weight', 'Obese', 'Overweight', '
Underweight'))
cm_nnet <- confusionMatrix(pred, testdata$NObeyesdad, positive = 'Normal Weig
ht')
cm_nnet
```

```
## Confusion Matrix and Statistics
##
##                 Reference
## Prediction      Normal Weight Obese Overweight Underweight
##   Normal Weight           113     0          1           2
##   Obese                     0   388         18           0
##   Overweight                5     8        205           0
##   Underweight               9     0          0         113
##
## Overall Statistics
##
##                Accuracy : 0.9501
##                  95% CI : (0.9334, 0.9637)
##     No Information Rate : 0.4594
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.9266
##
##  Mcnemar's Test P-Value : NA
##
```

```
## Statistics by Class:
##
##                      Class: Normal Weight Class: Obese Class: Overweight
## Sensitivity                        0.8898       0.9798            0.9152
## Specificity                        0.9959       0.9614            0.9796
## Pos Pred Value                     0.9741       0.9557            0.9404
## Neg Pred Value                     0.9812       0.9825            0.9705
## Prevalence                         0.1473       0.4594            0.2599
## Detection Rate                     0.1311       0.4501            0.2378
## Detection Prevalence               0.1346       0.4710            0.2529
## Balanced Accuracy                  0.9428       0.9706            0.9474
##                      Class: Underweight
## Sensitivity                      0.9826
## Specificity                      0.9880
## Pos Pred Value                   0.9262
## Neg Pred Value                   0.9973
## Prevalence                       0.1334
## Detection Rate                   0.1311
## Detection Prevalence             0.1415
## Balanced Accuracy                0.9853
```

Conclusion

Obesity is a serious health condition and can have severe consequences for health. It is increasing all over the world due to urbanization, economic development, and lifestyle changes and is considered an epidemic health problem. Therefore, it is vital to track the dietary habits and activity profiles of obese individuals to improve their quality of life and well-being. This study utilized a real-life dataset comprising various features related to dietary habits, physical conditions, activity profiles, and lifestyles.

In this study, we developed a neural network-based classification model for the pre diction of obesity levels based on physical activity levels and eating habits. The results shows that the prevalence of overweight and obesity is generally exceptionally high when compare to normal and underweight. Also, the reason why eating habits can lead to being obese or overweight may be that diets contain more and more high-calorie and, at the same time, high-fat foods, leading to a significant accumulation of fat in the body.

The proposed model was found to successfully obtain correct results that might decrease human mistakes in the diagnosis process and reduce the cost of cancer diagnosis. The approach presented in this study achieved an accuracy of 94%. The use of multiple NN models in a meta-learning framework allowed for better generalization and improved accuracy, particularly in detecting malignant tumors. The approach in medical imaging datasets could be extended to other types of cancer or medical conditions. The model produced by NN is more and it has the potential to make essential advancements in breast cancer prediction. Based on these findings, we can infer that machine learning techniques can automatically detect the disease with high accuracy.