04-02-2020

ADVANCED STATISTICS PROJECT

CEREAL DATA FACTOR ANALYSIS,LESLIE SALT DATA SET, ALL GREENS FRANCHISE

PRESENTED BY: SHILPA GIRIDHAR

CONTENTS

CEREAL DATA FACTOR ANALYSIS

The data file labeled Cereal has the following information

As part of a study of consumer consideration of ready-to-eat cereals sponsored by Kellogg Australia, Roberts and Lattin (1991) surveyed consumers regarding their perceptions of their favorite brands of cereals. Each respondent was asked to evaluate three preferred brands on each of 25 different attributes. Respondents used a five point Likert scale to indicate the extent to which each brand possessed the given attribute.

For the purpose of this assignment, a subset of the data collected by Roberts and Lattin, reflecting the evaluations of the 12 most frequently cited cereal brands in the sample (in the original study, a total of 40 different brands were evaluated by 121 respondents, but the majority of brands were rated by only a small number of consumers).

In total, 116 respondents provided 235 observations of the 12 selected brands. The 25 attributes and 12 brands are listed below

Cereal Brand	Attributes 1-12	Attributes 13-25
All Bran	Filling	Family
Cerola Muesli	Natural	Calories
Just Right	Fibre	Plain
Kellogg's corn flakes	Sweet	Crisp
Komplete	Easy	Regular
Nutrigrain	Salt	Sugar
Purina Muesli	Satisfying	Fruit
Rice Bubbles	Energy	Process
Special K	Fun	Quality
Sustain	Kids	Treat
Vitabrit	Soggy	Boring
Weetbix	Economical	Nutritious
	Health	

REQUIREMENTS

Topic	Marks
Problem 1- Cereal	**24**
1) Exploratory Data Analysis	
a) Basic data summary, Univariate, Bivariate analysis, graphs	4.5
2) PCA/FA	
a) Perform PCA/FA and Interpret the Eigen Values (apply Kaiser Normalization Rule)	12
b) Output Interpretation Tell which all factors needs to be shortlisted along with their importance and which ones needs to ignored. Name the factors with correct explanations.	7.5

EXPLORATORY DATA ANALYSIS

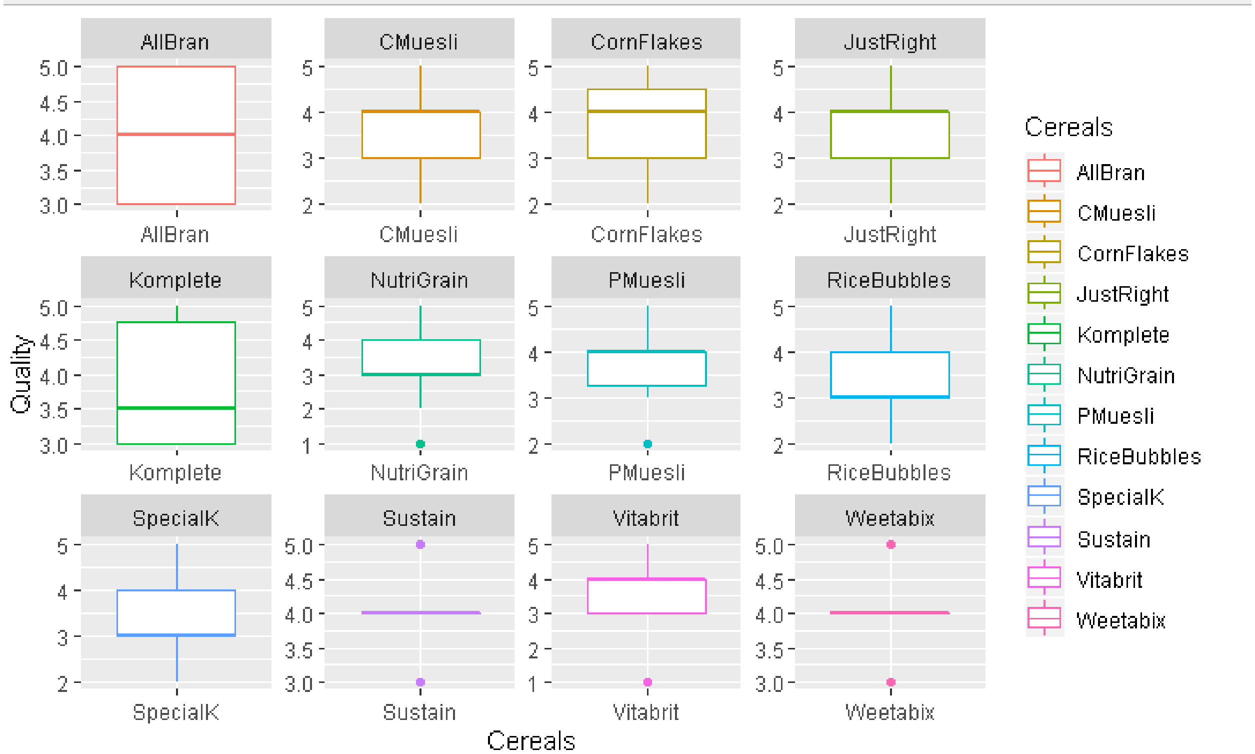
- The data shows that there are 235 observations and 26 variables
- Data rating is based on 5-point Likert Scale, a type of psychometric response scale in which responders specify their level of agreement to a statement typically in five points: (1) Strongly disagree; (2) Disagree; (3) Neither agree nor disagree; (4) Agree; (5) Strongly agree.
- Column Names of Dataset (cereal) are as follows -

```
## > [1] "Cereals"    "Filling"   "Natural"   "Fibre"     "Sweet"     "Easy"       "Salt"
## > [8] "Satisfying" "Energy"    "Fun"       "Kids"      "Soggy"     "Economical" "Health"
## > [15] "Family"    "Calories"  "Plain"     "Crisp"     "Regular"   "Sugar"      "Fruit"
## > [22] "Process"   "Quality"   "Treat"     "Boring"    "Nutritious"
```

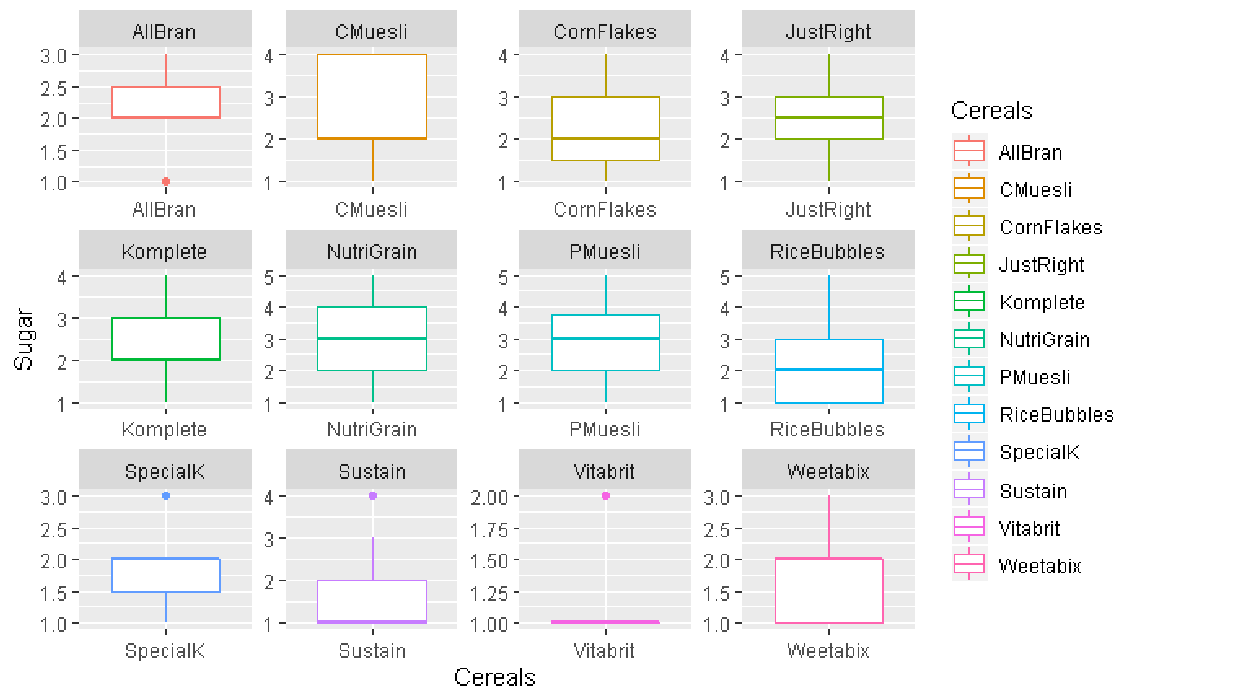
- The names of the brand are contained in the "Cereals" column
- Performed the str and summary function grouped by each brand to get an understanding of the data structure and average values (mean, median, mn, and max)
- Used the "library(DataExplorer)" and use the function "plot_missing(cereal)" to ascertain that there are No missing data
- Used the "library(ggplot2)" and use the function ggplot function to check if there are outliers. The data shows outliers in many occasions.

Example:

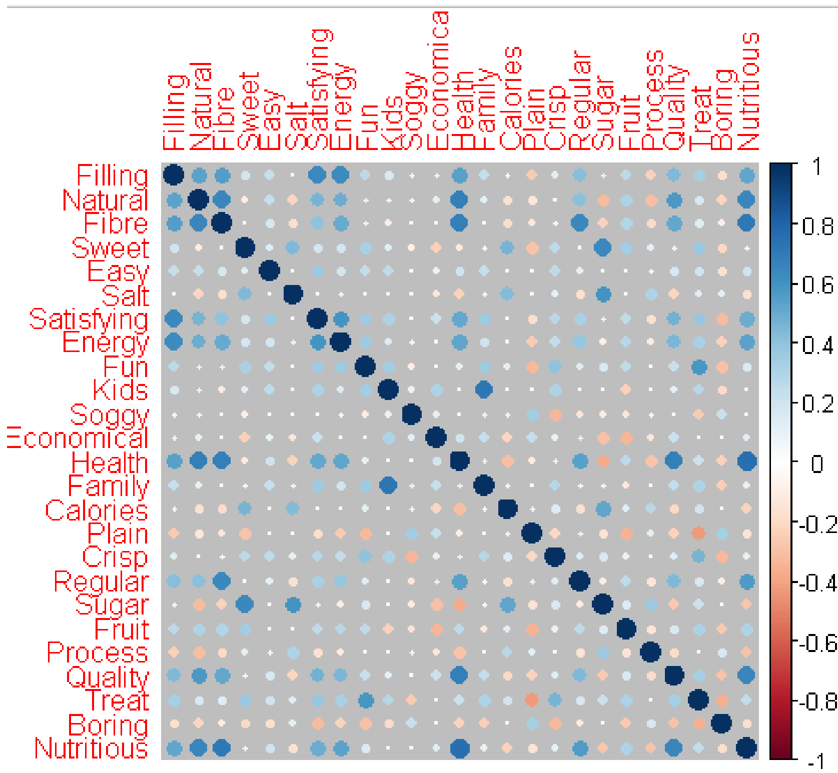
ggplot(cereal, aes(x=Cereals, y=Quality, fill=Quality, color=Cereals)) + geom_boxplot() + facet_wrap(~Cereals, scale="free")



ggplot(cereal, aes(x=Cereals, y=Sugar, fill=Sugar, color=Cereals)) + geom_boxplot() + facet_wrap(~Cereals, scale="free")



- Used the library(corrplot) to plot correlation and check for high correlation between the variables

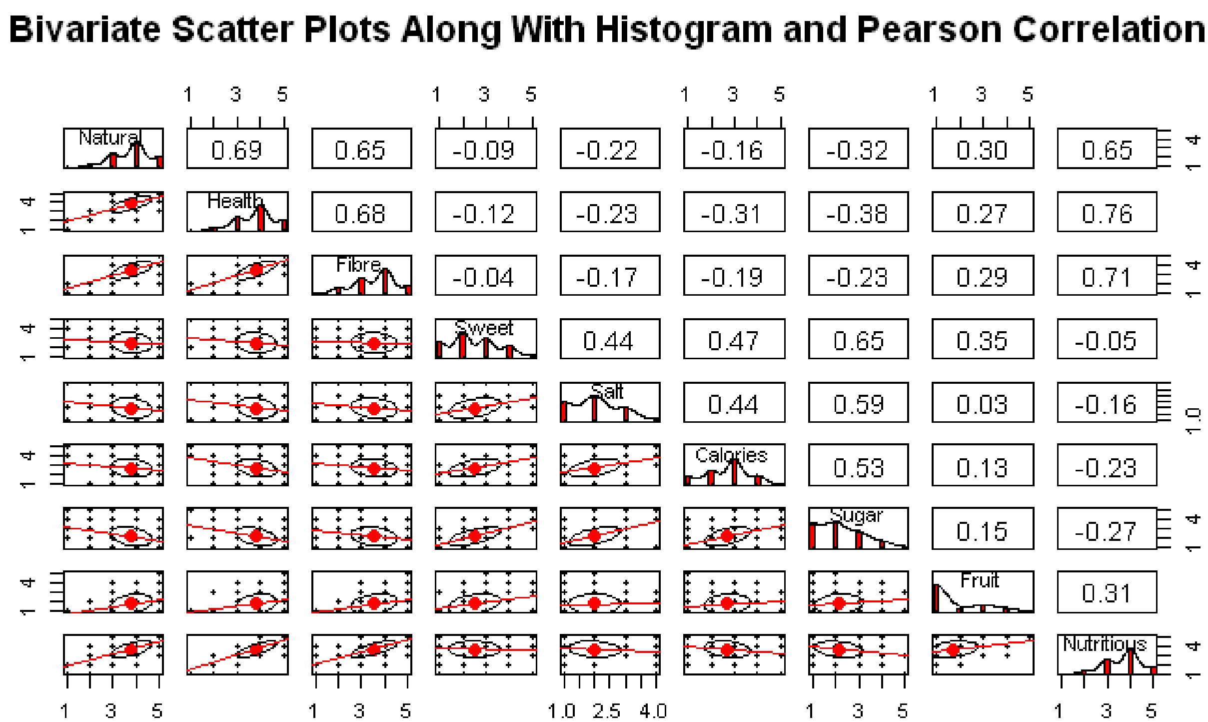


- By observation, we can group the likely variables that looks highly correlated to each other such as –
  - Those relating to Nutritional Values in Cereal – Natural, Health, Fibre, Sweet, Salt, Calories, Sugar, Fruit, Nutritious
  - Those relating to Taste Values in Cereal - Fun, Soggy, Crisp, Boring, Plain, Regular
  - Those relating to Family - Kids, Family, Treat, Easy, Process
  - Those relating to Satisfaction - Filling, Satisfying, Energy, Quality, Economical
- Use library(psych). Perform bivariate analysis using Scatter Plots and Pearson Correlation methods. The function “pairs.panels” [in psych package] can be used to create a scatter plot of matrices, with bivariate scatter plots below the diagonal, histograms on the diagonal, and the Pearson correlation above the diagonal. pairs.panels is most useful when the number of variables to plot is less than about 6-10. It is particularly useful for an initial overview of the data.
- The direction of the correlation is determined by whether the correlation plot is positive or negative. The closer a positive correlation lies to +1, the stronger it is.

Example 1:

High correlation can be observed between Natural<->Health<->Fibre<-Calories<->Nutritious

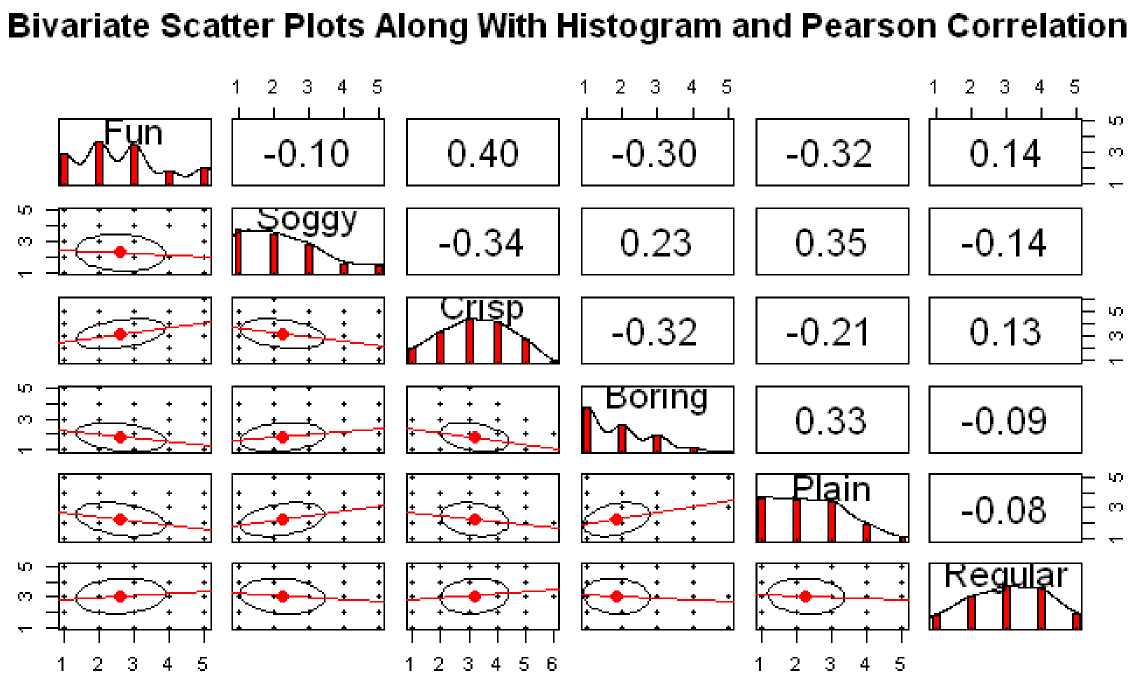
And between Sweet<-> Sugar<->Salt<->Calories



Correlation Plot 1

Example 2:

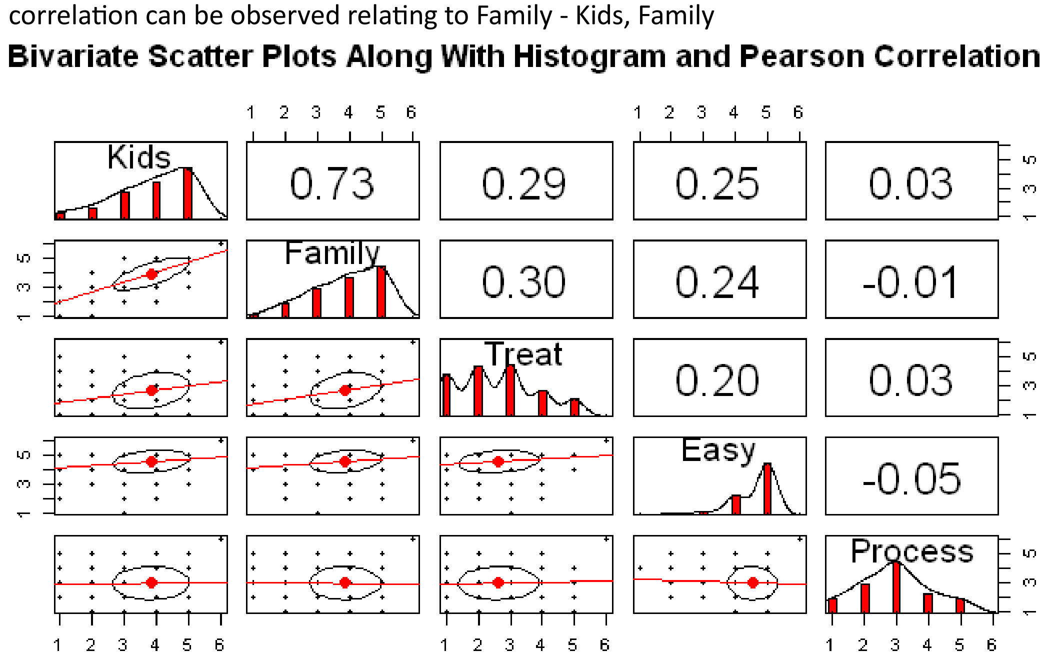
No significant correlation can be observed between Taste Values in Cereal - Fun, Soggy, Crisp, Boring, Plain, Regular



Correlation Plot 2

Example 2:

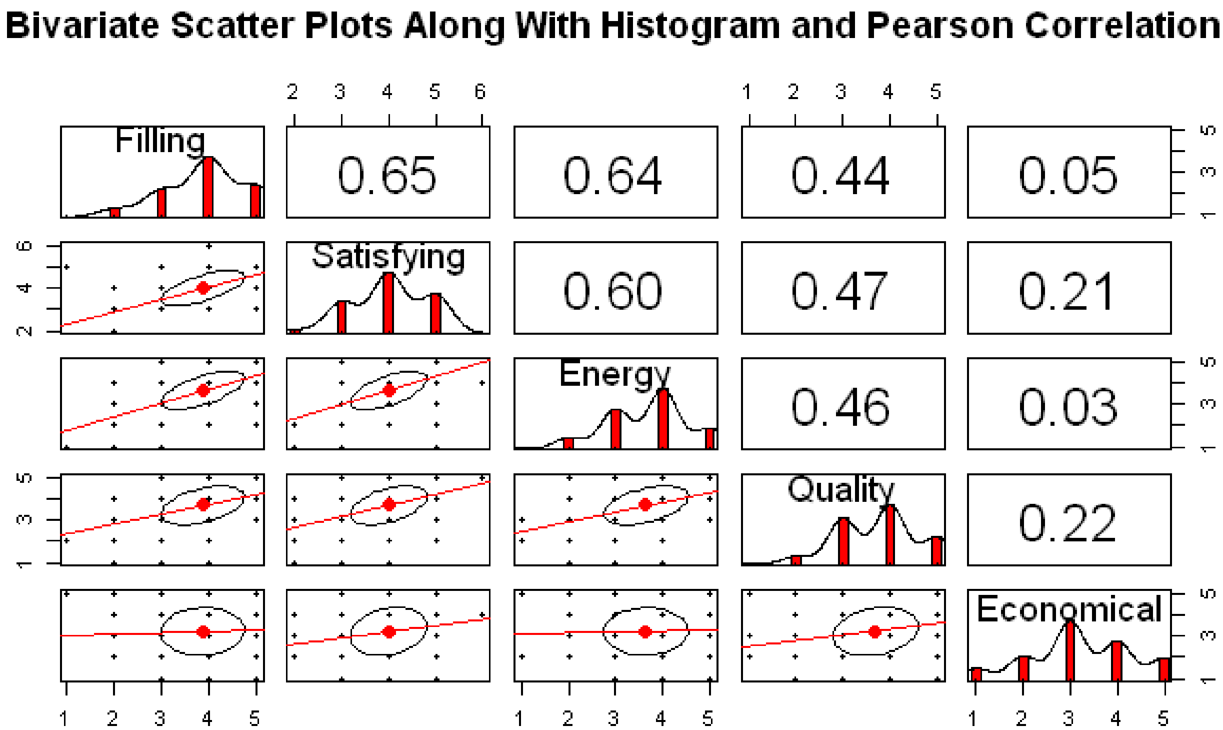
High correlation can be observed relating to Family - Kids, Family



Correlation Plot 3

Example 4:

High correlation can be observed relating to Satisfaction<-> Filling<-> Energy<-> Quality



Correlation Plot 4

PCA AND FA

PCA

Running PCA to identify the number of factors

##Apply Kaiser Rule to the cereal dataset (after removing the brands column)

cereal.pca <- princomp(cereal_new,scores = TRUE, cor = TRUE)

summary(cereal.pca)

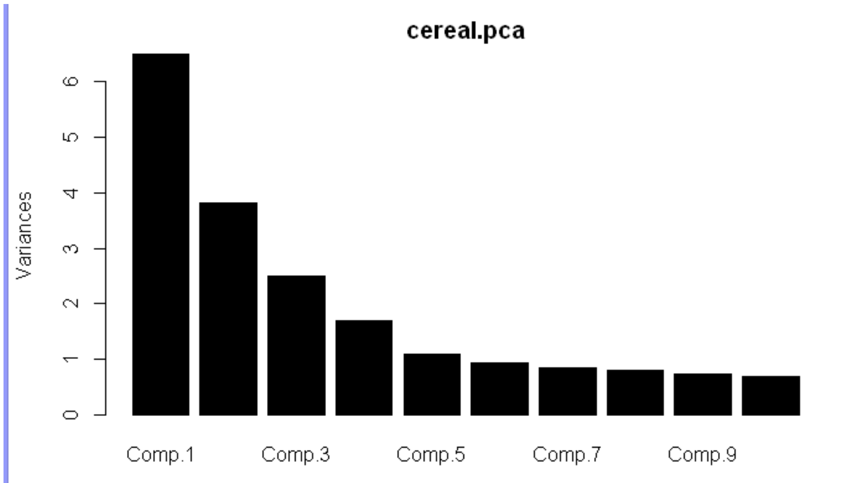
RESULTS:

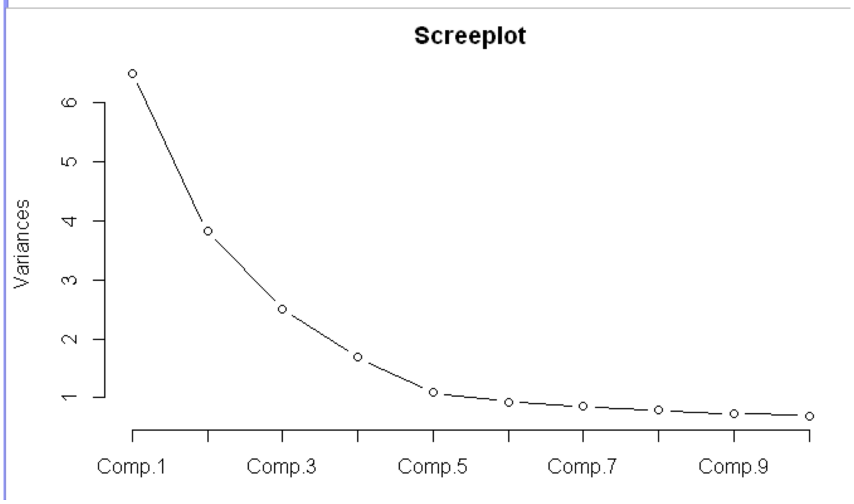
Comp1, Comp 2, and Comp 3 constitute 50% approx.(Proportion of variance)

Comp4, Comp 5, constitute 10% approx. .(Proportion of variance)

Comp 1 to Comp 5 - all component values have an Eigen value of greater than 1,

and hence, these 5 components will suffice and can be taken into consideration for the dimension reduction technique.





FA

From the above PCA, we now know that 5 factors to be sufficient to perform FA.

Use the commands in FA

factanal(cereal_new,factors = 5,rotation = "varimax")

RESULTS

```
Call:
factanal(x = cereal_new, factors = 5, rotation = "varimax")

Uniquenesses:
Filling   Natural    Fibre    Sweet     Easy     Salt Satisfying    Energy
0.283    0.389    0.311     0.361     0.838      0.513      0.373      0.432

Fun     Kids    Soggy Economical     Health      Family   Calories      Plain
 0.523    0.240    0.775    0.705     0.213      0.348      0.578      0.547

Crisp   Regular    Sugar    Fruit   Process    Quality      Treat     Boring
0.638     0.552     0.203     0.561    0.759     0.389      0.386      0.674

Nutritious
     0.242

Loadings:
           Factor1 Factor2 Factor3 Factor4 Factor5
Filling     0.647           0.190   0.144   0.487
Natural     0.731  -0.215                   0.153
Fibre       0.816
Sweet               0.696           0.351   0.166
Easy        0.230           0.307
Salt                0.689
Satisfying  0.570           0.387   0.199   0.333
Energy      0.611           0.168   0.225   0.339
Fun         0.125   0.155   0.377   0.538
Kids                        0.867
Soggy                       0.130  -0.454
Economical         -0.258   0.409  -0.197  -0.110
Health      0.840  -0.271
Family                      0.794   0.122
Calories   -0.155   0.592           0.122   0.179
Plain      -0.115                  -0.638  -0.150
Crisp               0.157   0.335   0.459
Regular     0.657
Sugar      -0.177   0.852           0.170
Fruit       0.341   0.161  -0.284   0.439   0.152
Process    -0.214   0.387          -0.101  -0.184
Quality     0.681  -0.222   0.200   0.218  -0.102
Treat       0.234   0.216   0.299   0.650
Boring     -0.150          -0.198  -0.508
Nutritious  0.849  -0.154
```

```
                Factor1 Factor2 Factor3 Factor4 Factor5
SS loadings       5.042   2.599   2.416   2.412   0.695
Proportion Var    0.202   0.104   0.097   0.096   0.028
Cumulative Var    0.202   0.306   0.402   0.499   0.527

Test of the hypothesis that 5 factors are sufficient.
The chi square statistic is 319.45 on 185 degrees of freedom.
The p-value is 3.09e-09
```

```
fact_cereal <- fa(r=cereal_new, nfactors=5, rotate="varimax", fm="pa")
fact_cereal
fa.diagram(fact_cereal)
```

```
                       PA1  PA3  PA2  PA4  PA5
SS loadings           5.10 2.66 2.62 1.73 1.16
Proportion Var        0.20 0.11 0.10 0.07 0.05
Cumulative Var        0.20 0.31 0.42 0.48 0.53
Proportion Explained  0.38 0.20 0.20 0.13 0.09
Cumulative Proportion 0.38 0.58 0.78 0.91 1.00

Mean item complexity =  1.8
Test of the hypothesis that 5 factors are sufficient.

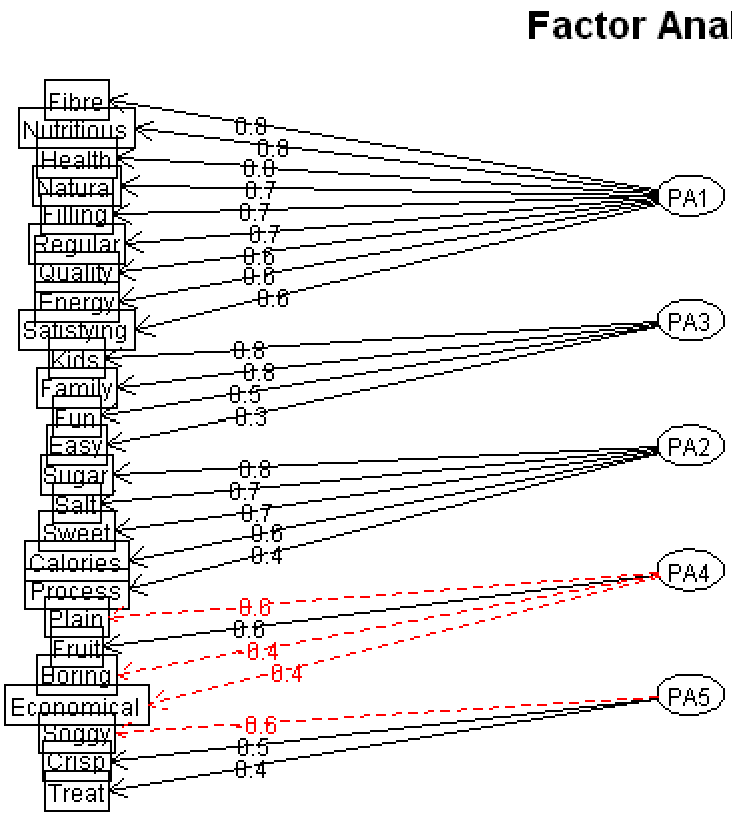
The degrees of freedom for the null model are  300  and the objective functio
n was  12.85 with Chi Square of  2888.04
The degrees of freedom for the model are 185  and the objective function was
 1.51

The root mean square of the residuals (RMSR) is  0.03
The df corrected root mean square of the residuals is  0.04

The harmonic number of observations is  235 with the empirical chi square  14
6.85  with prob <  0.98
The total number of observations was  235  with Likelihood Chi Square =  334.
62  with prob <  1e-10

Tucker Lewis Index of factoring reliability =  0.905
RMSEA index =  0.059  and the 90 % confidence intervals are  0.049 0.069
BIC =  -675.41
Fit based upon off diagonal values = 0.99
Measures of factor score adequacy
                                                   PA1  PA3  PA2  PA4  PA5
Correlation of (regression) scores with factors   0.96 0.93 0.92 0.83 0.79
Multiple R square of scores with factors          0.92 0.86 0.84 0.69 0.63
Minimum correlation of possible factor scores     0.84 0.72 0.68 0.38 0.26
```

Factor Analysis Diagram



ALL GREENS FRANCHISE

All Greens Franchise

Explain the importance of X2, X3, X4, X5, X6 on Annual Net Sales, X1.

The data (X1, X2, X3, X4, X5, X6) are for each franchise store.

X1 = annual net sales/$1000

X2 = number sq. ft./1000

X3 = inventory/$1000

X4 = amount spent on advertising/$1000

X5 = size of sales district/1000 families

X6 = number of competing stores in district

REQUIREMENTS

Problem 3- All Greens	**12**
a) Basic data summary, Univariate, Bivariate analysis, graphs	4.5
b) Correlation check , explanations of the relationships discovered, checking for linear relationship using Regression	7.5

DATA EXPLORATION

- The data shows that there are 27 observations and 6 variables
- Column Names of Dataset (allgreens) are as follows -

#X1 = Annual net sales/$1000 is numeric

#X2 = number sq. ft./1000 is numeric

#X3 = inventory/$1000 is numeric

#X4 = amount spent on advertising/$1000 is numeric

#X5 = size of sales district/1000 families is numeric

#X6 = number of competing stores in district is numeric

- change the col names to identify better

names(allgreens)[1] <- "AnnualSales"

names(allgreens)[2] <- "Area"

names(allgreens)[3] <- "Inventory"

names(allgreens)[4] <- "AdvAmout"

names(allgreens)[5] <- "Sales"

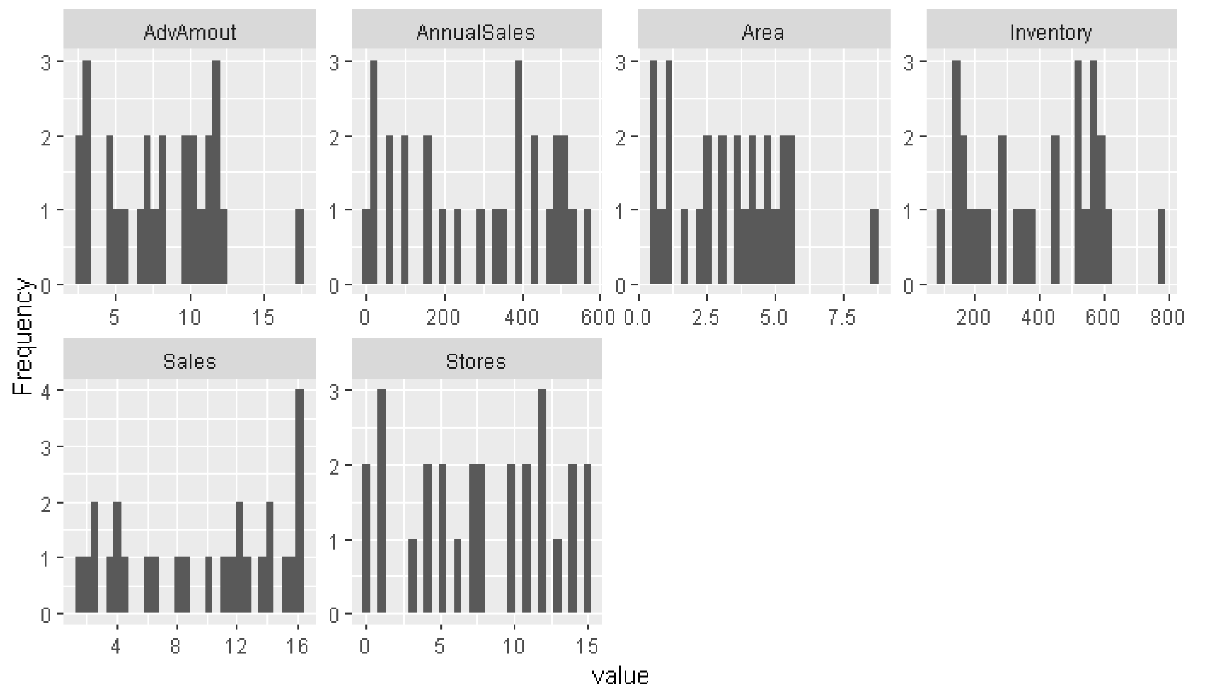
names(allgreens)[6] <- "Stores"

names(allgreens)

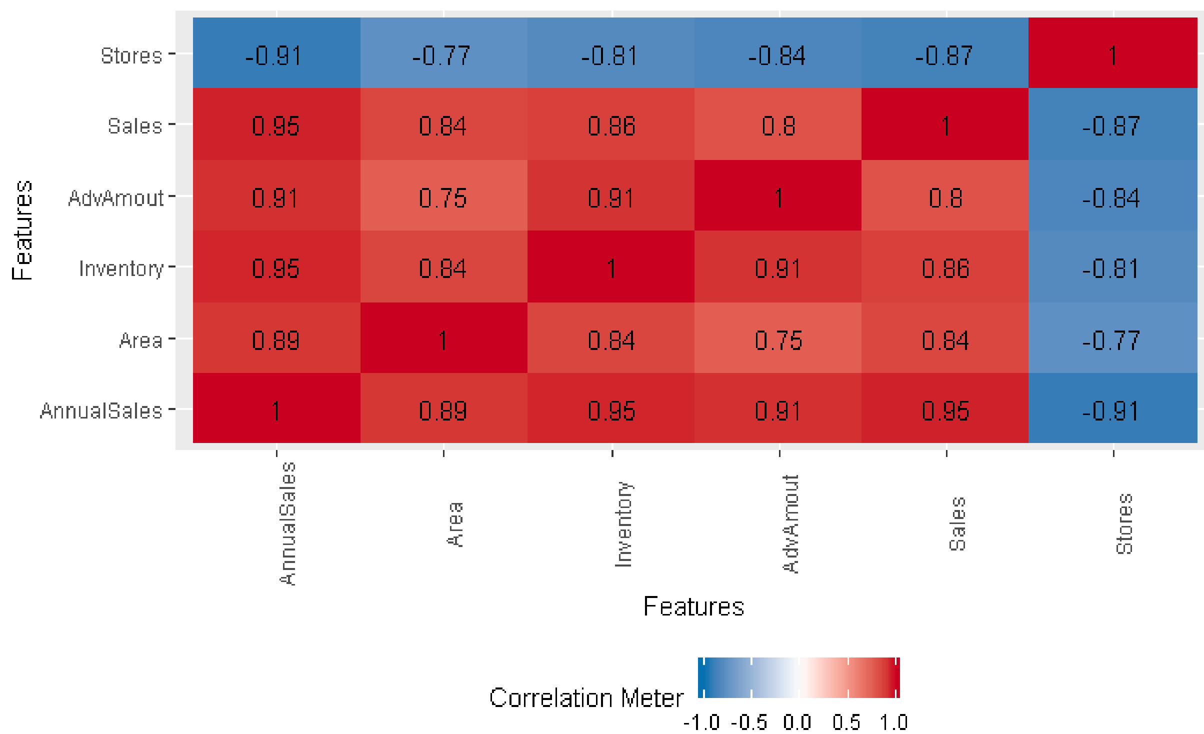
- Performed the str and summary function grouped by each brand to get an understanding of the data structure and average values (mean, median, min, and max)
- Used the “library(DataExplorer)” and use the function “plot_missing(cereal)” to ascertain that there are No missing data
- Used the “library(ggplot2)” and use the function ggplot function to check if there are outliers. The data shows outliers in many occasions.

UNIVARIATE AND BIVARIATE ANALYSIS

- We have one dependent variable and five independent variables
- The number of data points is only 27
- Use plot_histogram(allgreens)



- Use plot_correlation(allgreens)



- Use corrplot(cor(allgreens), method = "number")

- Also perform linear model between individual variables –
  Example: SLM2=lm(AnnualSales~Area), SLM3=lm(AnnualSales~Inventory), and so on.
- Inference - correlation matrix as well as linear model implies Annual sales is highly correlated with other 4 variables except Number of Stores
- Bivariate analysis to analyse two or more variables and examine their underlying relationships.

Example:

SLMb1=lm(AnnualSales~(Area+Inventory+AdvAmout+Sales+Stores))

- Use Variance Infation factor to check for multi-colinerity

  library(car)

  vif(SLMb1)

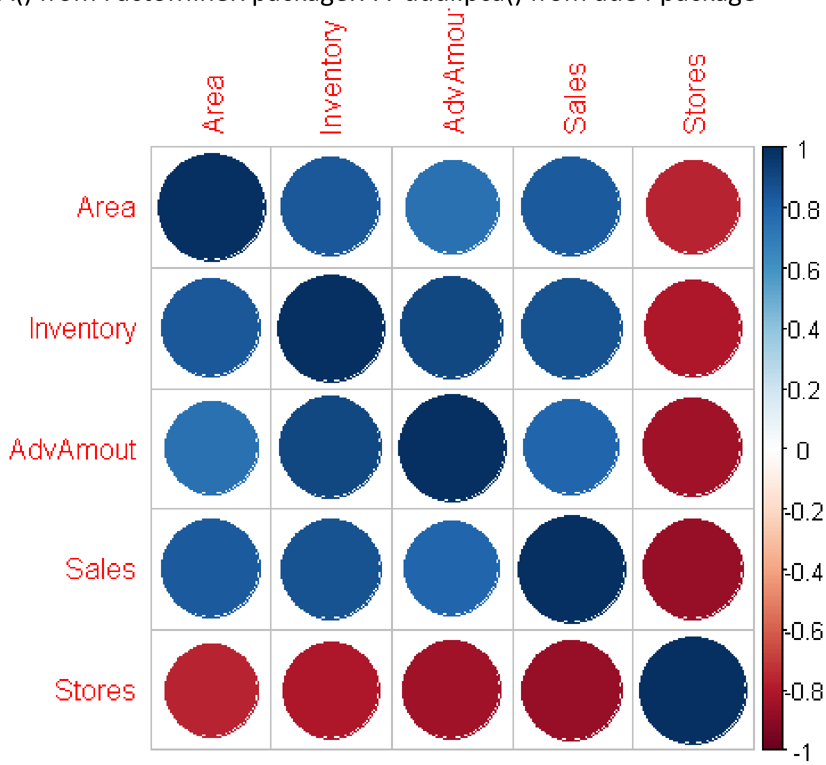
- Results
- Area Inventory AdvAmout Sales Stores
- 4.240914 10.122480 7.624391 6.912318 5.818768
- The variables with very high VIF (typically >4) means that we could drop that variable and and build a new model; So here we can remove the Inventory, which has very high VIF and re-build new model
  Example:

SLMb2=lm(AnnualSales~(Area+AdvAmout+Sales+Stores))

- This model is better, but there is still room for filtering out Sales variable
- Wwe can still drop the Sales variable as its VIF > 4

SLMb3=lm(AnnualSales~(Area+AdvAmout+Stores))
Results
Area   AdvAmout   Stores
2.657032   3.760743   3.996868

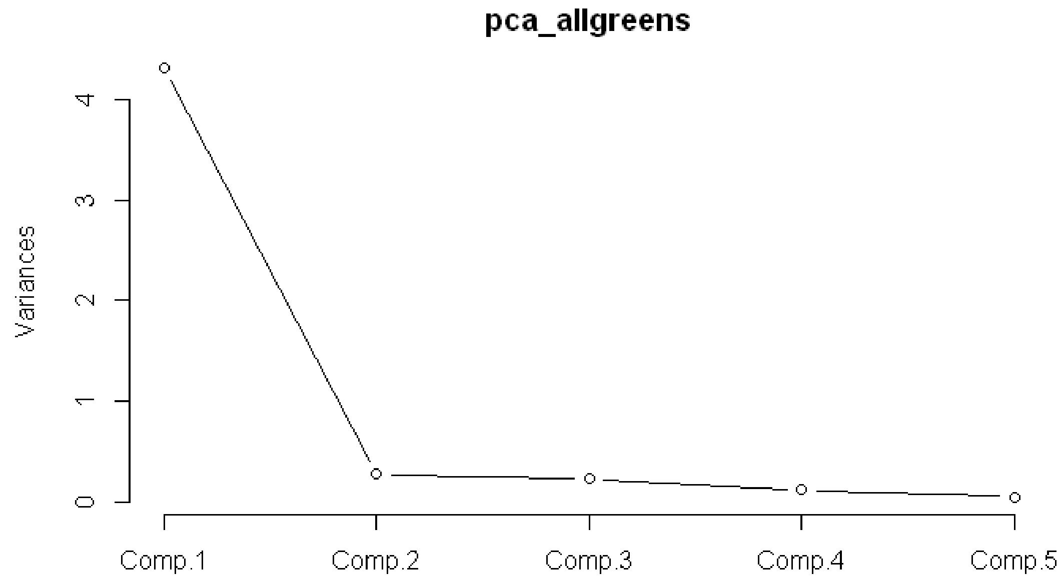
- Therefore the Problem of Multi-collinearity exists
- So we have to perform PCA and FA to resove multicollinearity and build better model;
  - ## we wil need to remove the dependent variable first;
  - ## There are several functions from different packages for performing PCA :
  - ## ??? The functions prcomp() and princomp() from the built-in R stats package;
  - ## PCA() from FactoMineR package.??? dudi.pca() from ade4 package




```
Importance of components:
                          Comp.1     Comp.2     Comp.3     Comp.4     Comp.5
Standard deviation      2.076857 0.52771292 0.47971777 0.35206709 0.23260188
Proportion of Variance 0.862667 0.05569619 0.04602583 0.02479025 0.01082073
Cumulative Proportion  0.862667 0.91836320 0.96438903 0.98917927 1.00000000
```

- From the output we can see that 86.2%, of the variation in the dataset is explained by the first component alone,
- Also only Comp1 has Eigen value of more than 1

- Use Kaiser method
- Any component with Eigen value greater than 1 is significant; rest can be dropped
  plot(pca_allgreens, type="line")



- Now perform FACTOR ANALYSIS
  library(GPArotation)
  library(psych)

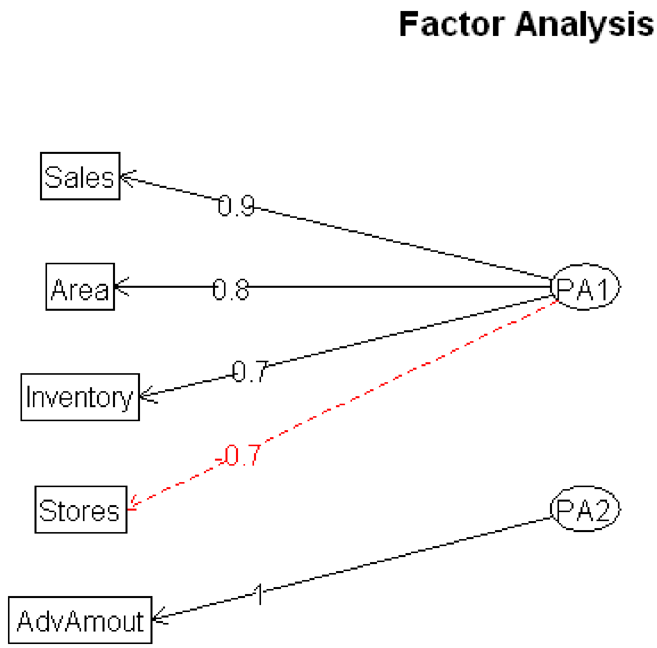
  pca_load_allgreens <- loadings(pca_allgreens)
  print(pca_load_allgreens, digits = 3, cutoff = 0.4, sort=TRUE)
  fact_allgreens <- fa(r=mat_allgreens3, nfactors=2, rotate="varimax", fm="pa")
  fact_allgreens
  fa.diagram(fact_allgreens)

  fact_allgreens <- fa(r=mat_allgreens3, nfactors=1, rotate="varimax", fm="pa")
  fact_allgreens
  fa.diagram(fact_allgreens)
  dim(fact_allgreens)
  biplot(fact_allgreens, scale=0)

```
Loadings:
                        Comp.1 Comp.2 Comp.3 Comp.4 Comp.5
mat_allgreens3Area       0.435  0.724         0.473       
mat_allgreens3AdvAmout   0.445 -0.585                0.548
mat_allgreens3Stores    -0.444         0.660 -0.412       
mat_allgreens3Sales      0.453               -0.679       
mat_allgreens3Inventory  0.459         0.477        -0.663

               Comp.1 Comp.2 Comp.3 Comp.4 Comp.5
SS loadings       1.0    1.0    1.0    1.0    1.0
```

```
Proportion Var     0.2     0.2     0.2     0.2     0.2
Cumulative Var     0.2     0.4     0.6     0.8     1.0
```



APPENDIX – PROBLEM 1 -CEREAL DATA ANALYSIS -R CODE

```
```{r}
setwd("C:/DATA/R-prog/wd/Datasets")
library(readr)
cereal=read_csv("cereal.csv")


#To view your dataset in R window
## View(allgreens)
names(cereal)
tail(cereal, 10)
View(cereal)

## > names(cereal)
## >  [1] "Cereals"    "Filling"    "Natural"    "Fibre"      "Sweet"      "Easy"       "Salt"
## >  [8] "Satisfying" "Energy"     "Fun"        "Kids"       "Soggy"      "Economical" "Health"
## > [15] "Family"     "Calories"   "Plain"      "Crisp"      "Regular"    "Sugar"      "Fruit"
## > [22] "Process"    "Quality"    "Treat"      "Boring"     "Nutritious"



dim(cereal) ## [1] 235 obs 26 variables

#Summary of data
summary(cereal)# 5 point summary

#by attaching you can call variables directly (you could avoid using $)
attach(cereal)

#datatype for each variable
str(cereal)

 ## FUN is Funcion - this command gives sumary for each product

by(cereal, INDICES = cereal$Cereals, FUN = summary)
##  check for missing Data
#install.packages("DataExplorer")
library(DataExplorer)

 ##plot_missing(cereal) ##No missing data


```
```

```
```{r}

##  check for Outliers in Data

## shows that Median values for all brands
## shows that there are outliers in NutriGain, PMuesli, Sustain,Viabit, and Weetbiax brands

# one box per Cereal
## "boxplot_cereal-quality.png"
p1 <-ggplot(cereal, aes(x=Cereals, y=Quality, fill=Quality, color=Cereals)) + geom_boxplot() +
facet_wrap(~Cereals, scale="free")
p1
# Save the file.

## "boxplot_cereal-Sugar.png"
## shows that Median values for all brands
## shows that there are outliers in SpecialK, Sustain, Viabit, and Allbran brands

p3<-ggplot(cereal, aes(x=Cereals, y=Sugar, fill=Sugar, color=Cereals)) + geom_boxplot() +
facet_wrap(~Cereals, scale="free")
p3

library(corrplot)
cereal_new<- cereal[,-1]
cereal_cor <- cor(cereal_new)
round(cereal_cor,2)
corrplot(cereal_cor, method = "circle",bg = "grey")
```

```{r}
#Take the original variables to vector c
cp<- subset(cereal,select = c(Cereals,Filling,Natural,Fibre,Sweet,Salt,Easy,Satisfying,
Energy,Fun,Kids,Soggy,Economical,Health,Family,Calories,Plain,Crisp,Regular,Sugar,
                      Fruit,Process,Quality,Treat,Boring,Nutritious))


#1) How Health, Fibre, Sweet, Salt, Calories, Sugar, Fruit, Nutritious are related?
library(psych)
corrplot1<-subset(cp,select=c(Natural, Health, Fibre, Sweet, Salt, Calories, Sugar, Fruit, Nutritious))

pairs.panels(corrplot1[,1:9],
         method = "pearson", #coorelation method
         hist.col = "red",
         main="Bivariate Scatter Plots Along With Histogram and Pearson Correlation",
         density = TRUE, # show density plots
         ellipses = TRUE, # show correlation ellipses
```
```

```
        lm=TRUE #linear regression fits
        )
#2) How  Fun, Soggy, Crisp, Boring, Plain, Regular related?
corrplot2<-subset(cp,select=c(Fun, Soggy, Crisp, Boring, Plain, Regular))
pairs.panels(corrplot2[,1:6],
        method = "pearson", #coorelation method
        hist.col = "red",
        main="Bivariate Scatter Plots Along With Histogram and Pearson Correlation",
        density = TRUE, # show density plots
        ellipses = TRUE, # show correlation ellipses
        lm=TRUE #linear regression fits
        )

#2) How are  Kids, Family, Treat, Easy, Process related?

corrplot3<-subset(cp,select=c(Kids, Family, Treat, Easy, Process))
pairs.panels(corrplot3[,1:5],
        method = "pearson", #coorelation method
        hist.col = "red",
        main="Bivariate Scatter Plots Along With Histogram and Pearson Correlation",
        density = TRUE, # show density plots
        ellipses = TRUE, # show correlation ellipses
        lm=TRUE #linear regression fits
        )
#3) How Filling, Satisfying, Energy, Quality, Economical related?

corrplot4<-subset(cp,select=c(Filling, Satisfying, Energy, Quality, Economical))
pairs.panels(corrplot4[,1:5],
        method = "pearson", #coorelation method
        hist.col = "red",
        main="Bivariate Scatter Plots Along With Histogram and Pearson Correlation",
        density = TRUE, # show density plots
        ellipses = TRUE, # show correlation ellipses
        lm=TRUE #linear regression fits
        )
```

Running PCA to identify the number of factors

```{r}

corrplot(cor(cereal_new), method = "circle",bg = "grey")

##Kaiser Rule

cereal_new<- cereal[,-1]

cereal.pca <- princomp(cereal_new,scores = TRUE, cor = TRUE)

summary(cereal.pca)
```

```
## Comp1, Comp 2, and Comp 3 constitute 50%  approx
## Comp4, Comp 5, constitute 10% approx
## Comp 1 to Comp 5 - all component values have an Eigen value of greater than 1,
## and hence can be taken into consideration for the dimension reduction technique.
plot(cereal.pca,col = "black")
screeplot(cereal.pca,type = "lines",main = "Screeplot")

## frm PCA, we now know that 5 factors to be sufficient to perform FA
factanal(cereal_new,factors = 5,rotation = "varimax")
fact_cereal <- fa(r=cereal_new, nfactors=5, rotate="varimax", fm="pa")
fact_cereal
fa.diagram(fact_cereal)
```
```
```

APPENDIX – PROBLEM 3 -ALL GREENS -R CODE

~~~
```{r}
setwd("C:/DATA/R-prog/wd/Datasets")
library(readr)
library(readxl)
allgreens=read_excel("Dataset_All Greens Franchise.xls")
#To view your dataset in R window
## View(allgreens)
head(allgreens, 10)


#How much is the data? Dimensions of the data
nrow(allgreens)# Number of Samples
ncol(allgreens)# Number of independent variables
dim(allgreens)
#total no of records :[1] 27 obs. of  6 variables:


#by attaching you can call variables directly (you could avoid using $)
attach(allgreens)

#datatype for each variable
str(allgreens)
class(X1) #X1 = Annual net sales/$1000  is numeric
class(X2) #X2 = number sq. ft./1000 is numeric
class(X3) #X3 = inventory/$1000  is numeric
class(X4) #X4 = amount spent on advertising/$1000  is numeric
class(X5) #X5 = size of sales district/1000 families  is numeric
class(X6) #X6 = number of competing stores in district is numeric

#change the col names
names(allgreens)[1] <- "AnnualSales"
names(allgreens)[2] <- "Area"
names(allgreens)[3] <- "Inventory"
names(allgreens)[4] <- "AdvAmout"
names(allgreens)[5] <- "Sales"
names(allgreens)[6] <- "Stores"
names(allgreens)

#Summary of data
summary(allgreens)# 5 point summary

#by attaching you can call variables directly (you could avoid using $)
attach(allgreens)

##  check for missing Data
#install.packages("DataExplorer")
~~~

```
library(DataExplorer)
library(corrplot)
 plot_missing(allgreens) ##No missing data



```

```{r}
## EDA Exploratory Data Analysis
## Univariate methods to analyse one variable at a time

## we have one dependent variable and five independent variables
## The number of data points is only 27
  plot_histogram(allgreens)
  plot_correlation(allgreens)
  corrplot(cor(allgreens), method = "number")
##Inference - correlation matrix implies Annual sales is highly correlated with other 4 variables except Number of Stores
## Therefore the Problem of Multi-colinearity exists
## So we have to perform FA to extract the principal component
```

```{r}
SLM2=lm(AnnualSales~Area)
summary(SLM2)
anova(SLM2)
## Multiple R-squared:  0.7994, F-statistic: 99.63 on 1 and 25 DF; p-value: 3.33e-10
## Inference - No of Sales is significantly dependent on the area of stores
# meaning the linear model of Sales depending on Area is robust and statistically valid.

SLM3=lm(AnnualSales~Inventory)
summary(SLM3)
anova(SLM3)
## Multiple R-squared:  0.894, F-statistic: 210.8 on 1 and 25 DF; p-value: 1.093e-13
## Inference - No of Sales is significantly dependent on the Inventory in stores
# meaning the linear model of Sales depending on Inventory is robust and statistically valid.

SLM4=lm(AnnualSales~AdvAmout)
summary(SLM4)
anova(SLM4)


## Multiple R-squared:  0.8354; F-statistic: 126.9 on 1 and 25 DF; p-value: 2.745e-11
## Inference - No of Sales is significantly dependent, but not as much as other variables,  on the AdvAmout in stores
# meaning the linear model of Sales depending on AdvAmout is robust and statistically valid.
```

````
SLM5=lm(AnnualSales~Sales)
summary(SLM5)
anova(SLM5)

## Multiple R-squared:  0.9095, F-statistic: 251.3 on 1 and 25 DF, p-value: 1.496e-14;
## Signif. codes: □***□
## Inference -  No of Sales is significantly dependent on the number of stores in a district
# meaning the linear model of Sales depending on Stores is robust and statistically valid.


SLM6=lm(AnnualSales~Stores)
summary(SLM6)
anova(SLM6)

## Multiple R-squared:  0.8322, F-statistic:   124 on 1 and 25 DF; p-value: 3.516e-11;
## Signif. codes: □***□
## Inference -  No of Sales is significantly dependent on the number of stores in a district
# meaning the linear model of Sales depending on Stores is robust and statistically valid.


#histogram plots..shape of the histogram is an important observation
hist(AnnualSales, main="AnnualSales in $1000", col = "grey")
boxplot(AnnualSales, main="AnnualSales", sub=paste("Outlier rows: ",
boxplot.stats(AnnualSales)$out))  # box plot for 'AnnualSales'
hist(Area,col="blue")
boxplot(Area, main="Area", sub=paste("Outlier rows: ", boxplot.stats(Area)$out))  # box plot for
'AnnualSales'

hist(AdvAmout,col="blue")
boxplot(AdvAmout, main="AdvAmout", sub=paste("Outlier rows: ", boxplot.stats(AdvAmout)$out))  #
box plot for 'AnnualSales'

hist(Inventory,col="blue")
boxplot(Inventory, main="Inventory", sub=paste("Outlier rows: ", boxplot.stats(Inventory)$out))  # box
plot for 'AnnualSales'

hist(Sales,col="blue")
boxplot(Sales, main="Sales", sub=paste("Outlier rows: ", boxplot.stats(Sales)$out))  # box plot for
'AnnualSales'

hist(Stores,col="blue")
boxplot(Stores, main="Stores", sub=paste("Outlier rows: ", boxplot.stats(Stores)$out))  # box plot for
'AnnualSales'


```
````
````

```{r}
## Bivariate analysis to analyse two or more variables and examine their underlying relationships.


SLMb1=lm(AnnualSales~(Area+Inventory+AdvAmout+Sales+Stores))
summary(SLMb1)
anova(SLMb1)

## Multiple R-squared:  0.9932, F-statistic: 611.6 on 5 and 21 DF, p-value: < 2.2e-16
## Inference - No of Sales is significantly dependent on all variables
## we need to check for multi-colinerity problem

## Use Variance Infation factor to check for multi-colinerity
library(car)
vif(SLMb1)
## Results
##      Area Inventory  AdvAmout     Sales    Stores
##  4.240914 10.122480  7.624391  6.912318  5.818768
## the variables with  very high VIF (typically >4) means that we could drop that variable and and build
## a new model; So here we can remove the Inventory, which has vry high VIF and re-build new model

SLMb2=lm(AnnualSales~(Area+AdvAmout+Sales+Stores))
summary(SLMb2)
anova(SLMb2)
## Multiple R-squared:  0.9902, F-statistic: 555.4 on 4 and 22 DF, p-value: < 2.2e-16
vif(SLMb2)

## Results
## Area      AdvAmout    Sales     Stores
## 3.579850   3.795323    5.861520 5.468943

## This model is better, but there is still room for filtering out Sales variable
## we can still drop the Sales variable as its VIF > 4

SLMb3=lm(AnnualSales~(Area+AdvAmout+Stores))
summary(SLMb3)
anova(SLMb3)
## Multiple R-squared:  0.9602, 184.9 on 3 and 23 DF, p-value: < 3.088e-16
vif(SLMb3)
## Results
##     Area  AdvAmout   Stores
## 2.657032  3.760743   3.996868

## Dropping the Outliers

cooks.distance(SLMb3)
## provides how far the the obs are from the mean values
## eliminate data points that are 4 times far from the mean
```

~~~
cd <- cooks.distance(SLMb3)
which(cd > 4*mean(cd))
## Results - 27 ; only one outlier ; we can drop 27 as outlier

allgreens2<- allgreens[-c(27),]
dim(allgreens2)
## result - [1] 26  6 (26 obs 6 col)

##build model again and check
SLMb4=lm(AnnualSales~(Area+AdvAmout+Stores), data=allgreens2)
summary(SLMb4)
anova(SLMb4)
##Multiple R-squared:  0.977,F-statistic: 311.4 on 3 and 22 DF,  p-value: < 2.2e-16


```


```{r}
## Now lets use PCA and FA to resove multicollinearity and build better model;
## we wil need to remove the dependent variable first;
##  There are several functions from different packages for performing PCA :
## ??? The functions prcomp() and princomp() from the built-in R stats package;
##  PCA() from FactoMineR package.??? dudi.pca() from ade4 package

library("factoextra")
allgreens3 <- allgreens[,-1]
dim(allgreens3)
View(allgreens3)
head(allgreens3)

##Cor matrix of allgreens3
mat_allgreens3  <- as.matrix(allgreens3)
corrplot(cor(mat_allgreens3))

##PCA on the new matrix
pca_allgreens <- princomp(~mat_allgreens3, scores = TRUE, cor = TRUE)
pca_allgreens
summary(pca_allgreens)

## From the output we can see that 86.2%, of the variation in the dataset is explained by the first ##
component alone,
## Also only Comp1 has  Eigen value of more than 1

## Use Kaiser method
## any component with Eigen value greater than 1 is significant; rest can be dropped
plot(pca_allgreens, type="line")
## By plotting we can see only 1 component is significant
~~~

````
screeplot(pca_allgreens)
From the scree plot we can see that the amount of variation explained drops dramatically after the
first component. This suggests that just one component may be sufficient to summarise the data.

Now perform FACTOR ANALYSIS
library(GPArotation)
library(psych)

pca_load_allgreens <- loadings(pca_allgreens)
print(pca_load_allgreens, digits = 3, cutoff = 0.4, sort=TRUE)
fact_allgreens <- fa(r=mat_allgreens3, nfactors=2, rotate="varimax", fm="pa")
fact_allgreens
fa.diagram(fact_allgreens)

fact_allgreens <- fa(r=mat_allgreens3, nfactors=1, rotate="varimax", fm="pa")
fact_allgreens
fa.diagram(fact_allgreens)
dim(fact_allgreens)
biplot(fact_allgreens, scale=0)
#

```
````
````