Report Assignment 1

Data Preparation and Data Warehousing & OLAP

DT15126 [EG] Fundamentals /Applied Data Sci 20219

Name: Abdallah Medhat Mohamed Rashed

Email: arash015@uOttawa.ca

ID: 300273110

Objectives

The purpose of this assignment to apply prepare data on the file csv and data warehousing using R.

Part A: Data Preparation

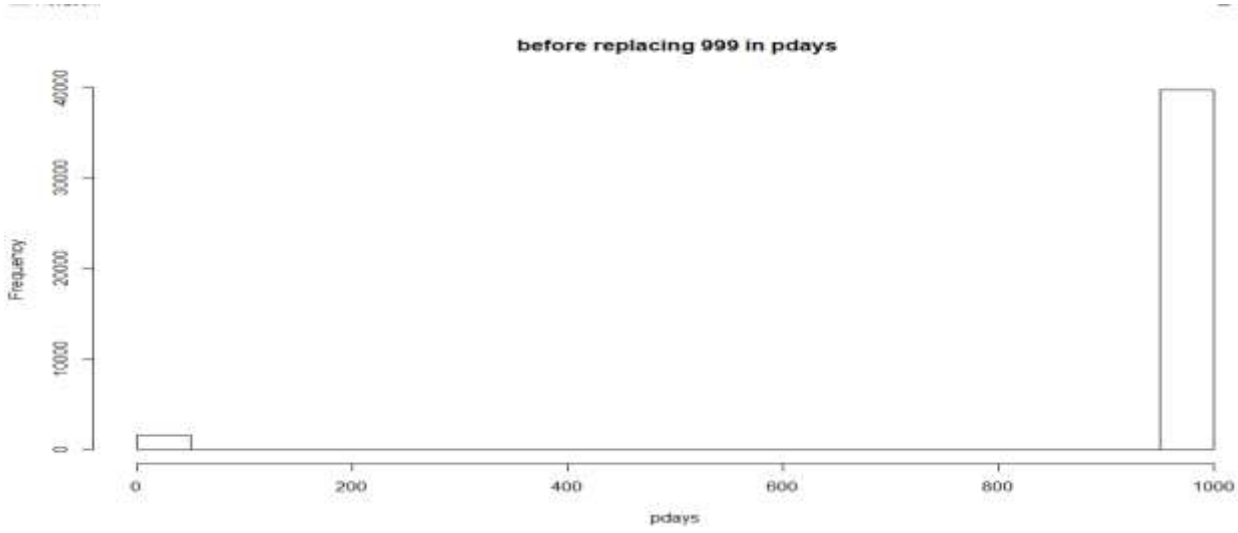
1) **Import the data (bank-additional-full.csv) by using read.csv and reduce the dataset to four predictors (age, education, previous, and pdays), and the target column, response (y) by using %>% select from library dplyr .**

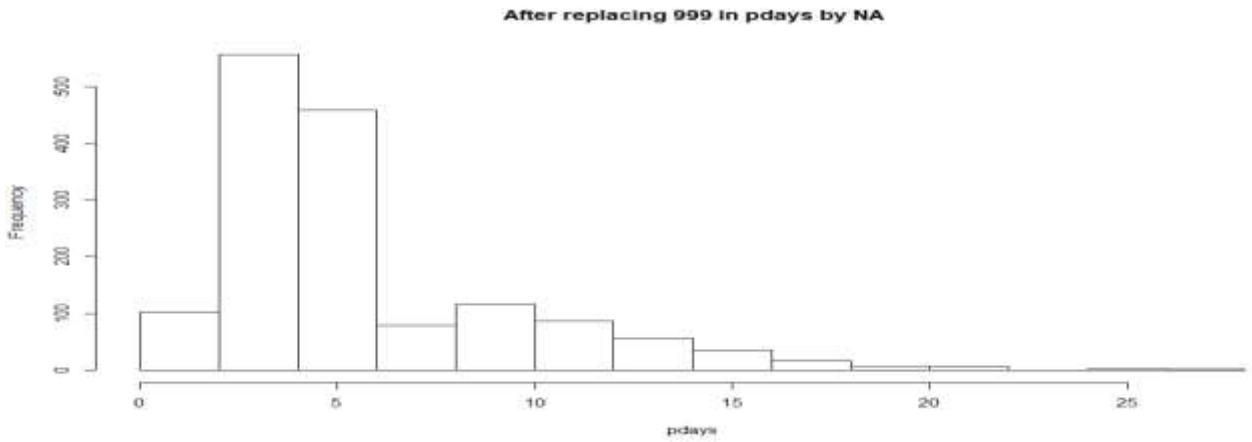
```
data_set<-read.csv("bank-additional-full.csv",header =TRUE,sep = ";")
library(dplyr)
data_set1<-data_set %>% select("age","education","previous","pdays","y")
```

2) **Change the field value 999 to "NA" to represent missing values.**

```
data_set1$pdays[data_set1$pdays ==999]<- NA
```

3) **Explain why the field pdays is essentially useless until you handle the 999 code.**
When using the NA in data, we can apply the mathematical operation like mean, median, mode by ignoring the NA but if we use the value 999, we can't approve the mathematical operation. When the use of NA instead of the value 999 data distribution was very different.





4) **Visualize pdays after replacing value 999 with NA by histogram.(histogram after replacing value 999 in the question 3).**

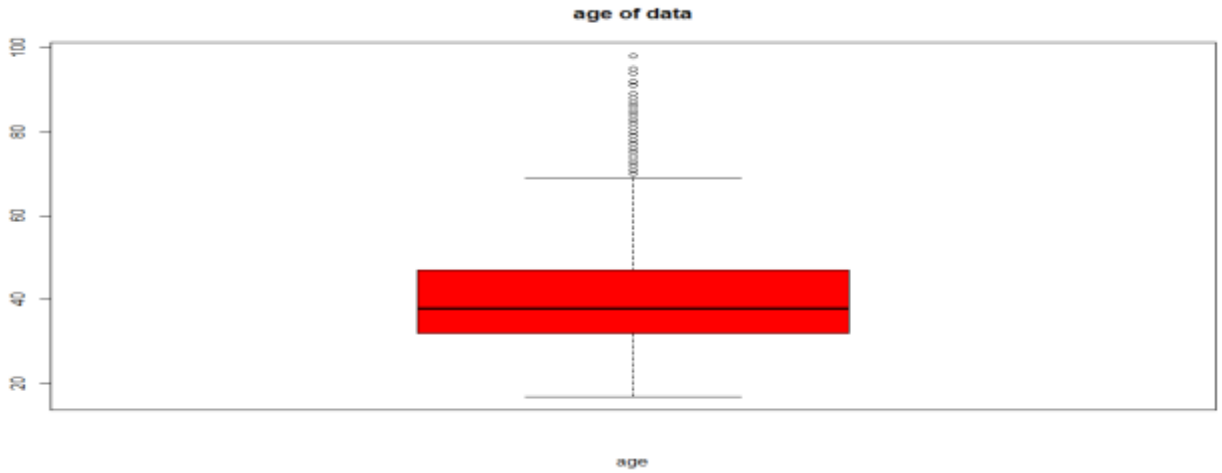
```
summary(data_set1)
hist(as.numeric(data_set1$pdays), main = " After replacing 999 in pdays by NA",
     xlab = "pdays")
```

5) **Transform the data values of the education field into numeric values by using function (revalue) from library (plyr).**

```
library(plyr)
replace_edu<- revalue(data_set1$education,c("illiterate" = 0, "basic.4y" = 4,
    "basic.6y" = 6,"basic.9y"=9,"high.school"=12,
    "professional.course"=14,"university.degree"=16,"unknown"=NA)
    ,warn_missing = TRUE)
data_set1$education=replace_edu
```

6) **Compute the mean, median & mode of the age variable by functions (mean, median and mfv from library (statip)) and using a boxplot, give the five number summary of the data by function (boxplot).**

```
age_mean <- mean(data_set1$age,na.rm = FALSE)
age_median <- median(data_set1$age, na.rm = FALSE)
#install.packages("statip")
library(statip)
age_mode <- mfv(data_set1$age)
#another way to find mode to age
age__mode <- function(x) {
  uniqx<- unique(x)
  uniqx[which.max(tabulate(match(x, uniqx)))]
}
result <- age__mode(data_set1$age)
summary(data_set1$age)
boxplot(data_set1$age,data=data_set1, main="age of data", xlab="age",col = "red")
```

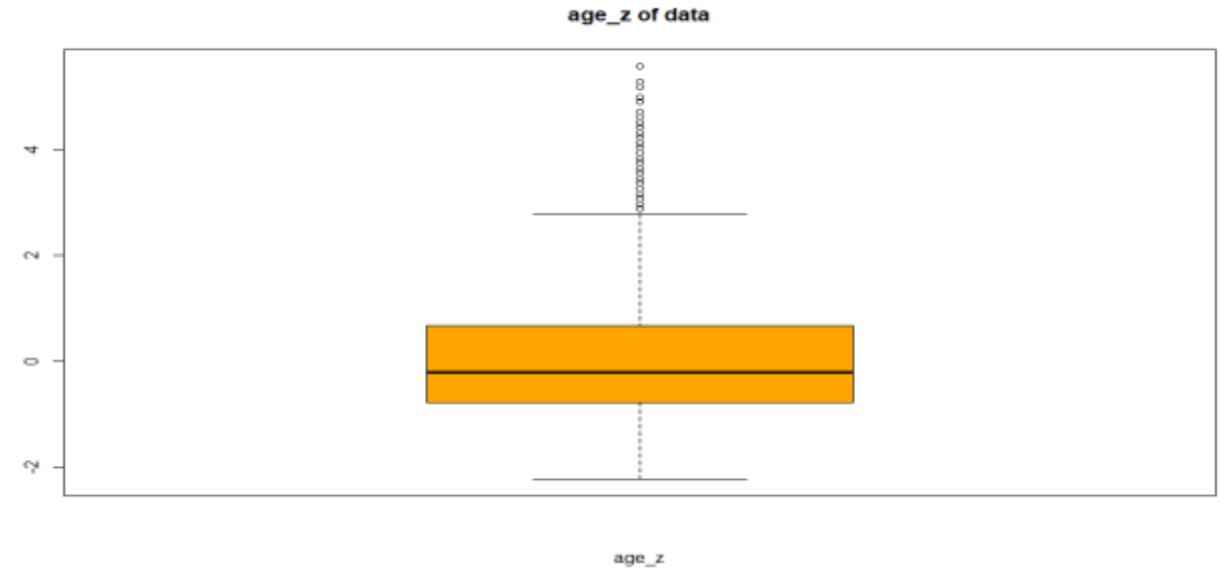


7) **Standardize the age variable by using function (scale) to prepare data and use data in any model or any analysis.**

```
age_z <- scale(x = data_set1$age)
data_set1$age_z<-age_z
```

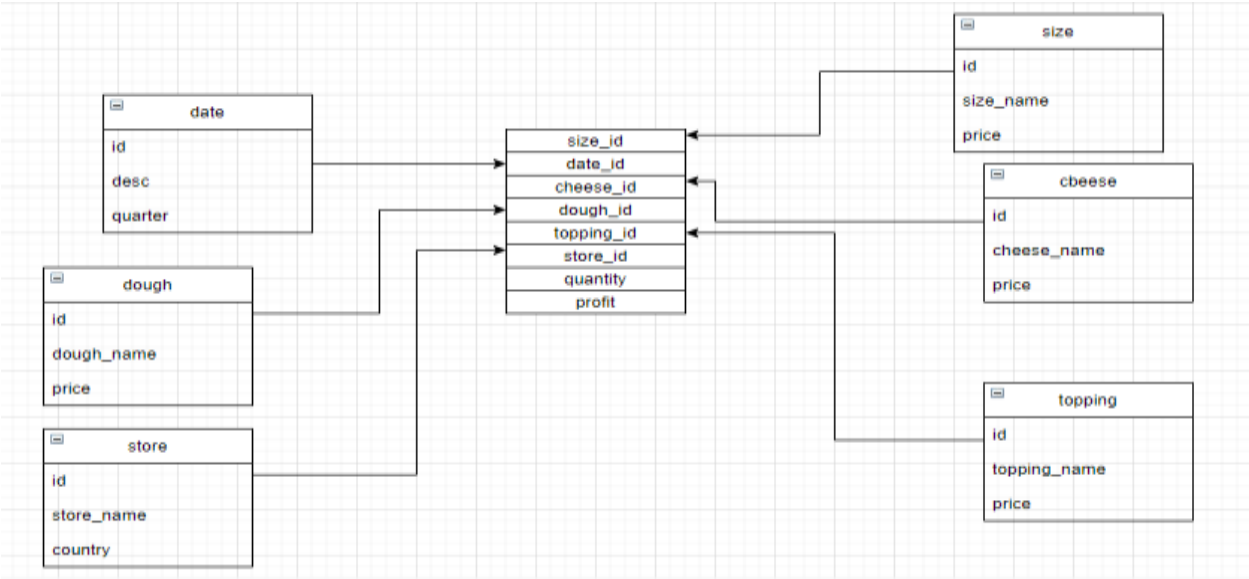
8) **Obtain outliers from age_z by using function (boxplot).**

```
boxplot(data_set1$age_z,data=data_set1, main="age_z of data", xlab="age_z",
        col = "orange")
age_z_outliers <-  data_set1[ which( data_set1$agez  > 2), ]
age_z_outliers
```

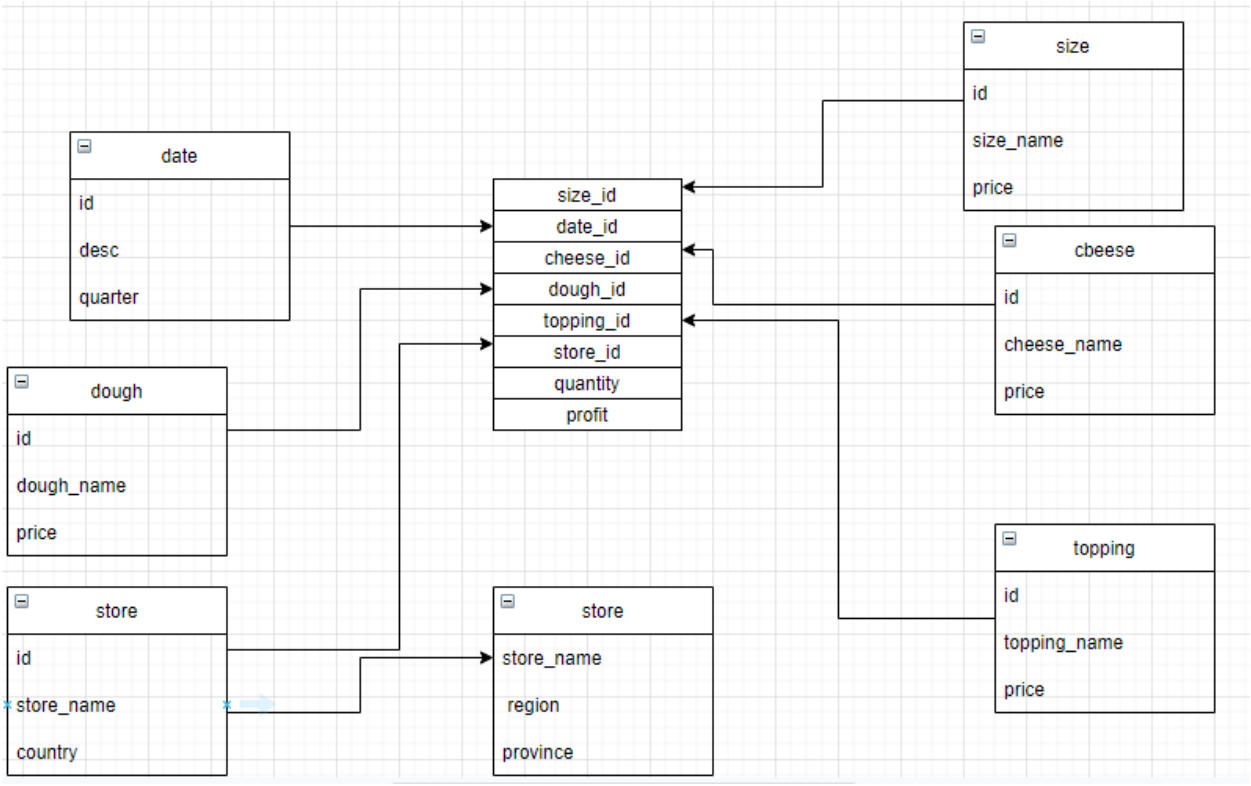


Part B: Data Warehousing & OLAP

1) Star schema represents the problem of pizza wishes.



2) Snowflake schema represents the problem of pizza wishes.



3) Generate a set of sample data stored in csv files for the dimensions and fact table and using R to read csv files and make the fact table then save fact table in file csv.
   1) Generate five files csv (cheese, dough, size, store, tooping) and make data frame for date (month).

```
cheese_table <- read.csv("cheese.csv",header =TRUE,sep = ";")
dough_table <- read.csv("dough.csv",header =TRUE,sep = ";")
sizes_table <- read.csv("size.csv",header =TRUE,sep = ";")
state_table <- read.csv("store.csv",header =TRUE,sep = ";")
topping_table <- read.csv("tooping.csv",header =TRUE,sep = ";")
date_table <-
  data.frame(id=1:12,
             desc=c("Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul",
                    "Aug", "Sep", "Oct", "Nov", "Dec"),
             quarter=c("Q1","Q1","Q1","Q2","Q2","Q2","Q3",
                       "Q3","Q3","Q4","Q4","Q4"))
```

2) Generate the fact table and save it in file csv (sales_pizza). It consist of 700 record and was ordered by month and year.

```
gen_sales <- function(no_of_recs) {

  # Generate transaction data randomly
  loc <- sample(state_table$store_name, no_of_recs,
                replace=T, prob=c(2,2,1,1,1))
  time_month <- sample(date_table$id, no_of_recs, replace=T)
  time_year <- sample(c(2018, 2019), no_of_recs, replace=T)
  cheese <- sample(cheese_table$cheese_name, no_of_recs, replace=T)
  size <- sample(sizes_table$size_name, no_of_recs, replace=T,prob=c(1,1,2,2,2))
  dough <- sample(dough_table$Dough_name, no_of_recs, replace=T)
  topping <- sample(topping_table$topping_name, no_of_recs, replace=T)
  Quantity <- sample(c(1,2), no_of_recs, replace=T, prob=c(10, 3))
  profit <- Quantity*(cheese_table$price+sizes_table$price+topping_table$price
                      +dough_table$price)

  sales <- data.frame(month=time_month,
                      year=time_year,
                      loc=loc,
                      cheese=cheese,
                      size=size,
                      dough=dough,
                      topping=topping,
                      Quantity=Quantity,
                      profit=profit
                      )

  # Sort the records by time order
  sales <- sales[order(sales$year, sales$month),]
  row.names(sales) <- NULL
  return(sales)
}
```

```
# Now create the sales fact table
sales_fact <- gen_sales(700)
write.csv(sales_fact,"sales_pizza.csv")
sales=read.csv('sales_pizza.csv',header =TRUE,sep = ";")
# Look at a few records
head(sales)
```

4) Build an OLAP cube for your revenue and show the cells of a subset of the cells by applying the mathematical operation sum (revenue cube) on profit in the fact table then apply Slice and Dice on the revenue cube.

```
# Build up a cube
revenue_cube <-
  tapply(sales_fact$profit,
         sales_fact[,c("size","cheese","dough","topping", "month", "year","loc")],
         FUN=function(x){return(sum(x))})

# Showing the cells of the cude
revenue_cube
dimnames(revenue_cube)
#SLICE
revenue_cube[,,,, "2", "2019",]
revenue_cube[,,,, "2", "2018",]
revenue_cube["xlarge","Mozzarella","stuffed crust","onions",
             "1","2018" ,"California"]
#DICE
revenue_cube[c("large","small"),,,,
             c("1","2","3"),
             ,
             c("California","new York")]
```

5) To know customers are beginning to prefer bigger pizzas by using drill-down and roll-up.

```
# drilldown and roll-up
apply(revenue_cube, c("year", "size"),
      FUN=function(x) {return(max(x, na.rm=TRUE))})
apply(revenue_cube, c("year", "size","month"),
      FUN=function(x) {return(sum(x, na.rm=TRUE))})
```

After using drill-down and roll-up, the result shows customers prefer pizza large or x-large (bigger pizza).
The big pizza is trending.

```
      size
year    medium   xlarge large personal small
  2018       58       58    58       78    58
  2019       78       67    78       58    58
```

Other result in March with 2018 and 2019.

```
, , month = 3

      size
year    medium   xlarge large personal small
  2018      149      285   126      133    26
  2019       74      239   262      133   110
```

Conclusion

In conclusion, this report could be summarized by doing prepare data such as select, transform, scale, visualize some columns and replace some values in dataset and data warehousing use the star schema, snowflake schema, cube, drill-down and roll-up. Applying all operations by using R. .