# Assignment Zomato

*Saurav*

*April 10, 2017*

Loading the dataset

```
library(readxl)

data<-read_excel("Case_Study_Model.xlsx")

data<-as.data.frame(data)
```

Converting into catogorical variables

```
for(i in c(2:10,12:34)){
  data[,i]<-as.factor(data[,i])
}
```

Structure of Dataset

```
str(data)
```

```
## 'data.frame':    103961 obs. of  34 variables:
##  $ Key                    : num  1 2 3 4 5 6 7 8 9 10 ...
##  $ tier1_MOQ_             : Factor w/ 6 levels "0","100 to 200",..: 1 1 6 4 6 4 4 1 6 4 ...
##  $ Z_Ratings_             : Factor w/ 10 levels "0","2 to 2.5",..: 9 8 3 6 7 7 8 6 1 4 ...
##  $ chain_counter          : Factor w/ 10 levels "1","11 to 15",..: 1 4 1 9 4 9 9 4 1 1 ...
##  $ chain_counter_nat      : Factor w/ 9 levels "0 to 1","11 to 30",..: 1 4 1 7 4 7 2 4 4 1
 ...
##  $ meal_time              : Factor w/ 5 levels "Breakfast","Dinner",..: 5 5 2 5 3 5 2 5 2 5
 ...
##  $ payment_method_type    : Factor w/ 4 levels "card","cash",..: 1 4 2 1 4 2 3 4 1 1 ...
##  $ COD_or_OP_Flag         : Factor w/ 2 levels "COD","OP": 2 2 1 2 2 1 2 2 2 2 ...
##  $ source                 : Factor w/ 5 levels "android","iphone",..: 4 4 3 4 1 4 2 2 1 4 ...
##  $ app_type               : Factor w/ 11 levels "0","1","2","3",..: 9 9 9 9 9 9 9 9 9 9 ...
##  $ repeat                 : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ total_user_band        : Factor w/ 10 levels "(0) 904.04 to 16609.2",..: 1 5 2 4 5 3 1 2 7
1 ...
##  $ res_DSZ_distance_band  : Factor w/ 10 levels "(0) 3.23 to 8752.99",..: 4 3 7 3 6 10 2 6 4
 9 ...
##  $ res_count_dsz_band     : Factor w/ 10 levels "(0) 415 to 530",..: 1 6 2 6 6 4 5 4 8 5 ...
##  $ jan_ulv_band           : Factor w/ 10 levels "(0) 2553 to 15734",..: 1 6 10 6 4 5 2 6 10 6
...
##  $ jan_uni_menu_open_band : Factor w/ 10 levels "(0) 2689 to 6521",..: 1 8 10 7 5 6 7 10 10 9
...
##  $ ULV_to_Orders_band     : Factor w/ 11 levels "(.) 0 to 2.06",..: 10 10 11 9 9 10 11 11 11
 11 ...
##  $ MenuOpen_to_Orders_band: Factor w/ 11 levels "(.) 0 to 0.53",..: 9 9 11 7 9 10 9 10 11 11
 ...
##  $ cost_for_two_band      : Factor w/ 9 levels "(0) 950 to 2600",..: 9 6 9 9 9 9 1 5 8 9 ...
##  $ Res_Jan_orders_band    : Factor w/ 10 levels "(0) 680 to 1470",..: 1 8 10 7 7 7 8 10 10 10
...
##  $ EDT_band               : Factor w/ 9 levels "(0) 66 to 3020",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ DDT_band               : Factor w/ 7 levels "(0) 65 to 300",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ Delay_Exp_band         : Factor w/ 8 levels "(0) 15 to 2930",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ Delay_Exp2_band        : Factor w/ 8 levels "(0) 0.33 to 32.56",..: 1 1 1 1 1 1 1 1 1 1
 ...
##  $ CD_Supported_order     : Factor w/ 2 levels "0","1": 1 1 1 1 2 1 1 1 2 2 ...
##  $ Salt_discounted_order  : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Z_discounted_order     : Factor w/ 2 levels "0","1": 1 2 1 1 1 1 1 1 2 1 ...
##  $ Z_referral_order       : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ deep_cuisines_gt5      : Factor w/ 21 levels "0","1","2","3",..: 16 7 17 7 7 9 9 14 5 9
 ...
##  $ is_synergy             : Factor w/ 2 levels "0","1": 2 1 1 2 1 2 1 1 1 1 ...
##  $ logistics_partner_id   : Factor w/ 3 levels "0","3","8": 1 1 1 1 1 1 1 1 1 1 ...
##  $ rating                 : Factor w/ 6 levels "0","1","2","3",..: 1 1 1 1 4 1 4 6 4 1 ...
##  $ user_reported_delay    : Factor w/ 2 levels "0","1": 1 1 1 1 2 1 1 2 2 1 ...
##  $ cuisines               : Factor w/ 47 levels "2","3","4","5",..: 38 21 39 28 22 26 32 33 1
7 25 ...
```

Any missing values?

```
sum(is.na(data))
```

```
## [1] 2
```

Where are the missing values

```
sort(sapply(data,function(x){sum(is.na(x))}),decreasing = T)
```

```
##     payment_method_type                     Key              tier1_MOQ_ 
##                       2                       0                       0 
##              Z_Ratings_           chain_counter       chain_counter_nat 
##                       0                       0                       0 
##               meal_time          COD_or_OP_Flag                  source 
##                       0                       0                       0 
##                app_type                  repeat         total_user_band 
##                       0                       0                       0 
##   res_DSZ_distance_band      res_count_dsz_band            jan_ulv_band 
##                       0                       0                       0 
##  jan_uni_menu_open_band      ULV_to_Orders_band MenuOpen_to_Orders_band 
##                       0                       0                       0 
##       cost_for_two_band     Res_Jan_orders_band               EDT_band 
##                       0                       0                       0 
##               DDT_band          Delay_Exp_band         Delay_Exp2_band 
##                       0                       0                       0 
##      CD_Supported_order   Salt_discounted_order      Z_discounted_order 
##                       0                       0                       0 
##        Z_referral_order       deep_cuisines_gt5              is_synergy 
##                       0                       0                       0 
##    logistics_partner_id                  rating     user_reported_delay 
##                       0                       0                       0 
##                cuisines 
##                       0
```

Filling in the missing values with mode

NOTE: From the business prespective it would be important to find what created an NA for the payment mode type variable. But I'm restricting myself as that's out of the scope of proble statement.

```
Mode <- function(x) {
    uniq <- unique(x)
    uniq[which.max(tabulate(match(x, uniq)))]
}

data$payment_method_type[is.na(data$payment_method_type)]<-as.factor(Mode(data$payment_method_ty
pe))

sum(is.na(data))
```

```
## [1] 0
```

Moving on to clustering. Because of the presence of Catagorical and ordinal variables, I'm going to use K-Modes clustering algorithm for Creating 3 clusters. Because of time issue, i'll be using a 10% sample of the dataset only but its necessaricly the same thing. The sample will have same distribution of the repeat variable

```
#Creating split
library('caret')

index <- createDataPartition(data[,"repeat"], p=0.1, list=FALSE)
data_sample <- data[index,]

library('klaR')

#Setting random seed for reproduciblity
set.seed(1)

clust<-kmodes(data_sample[,c(2:10,12:34)], 3, iter.max = 5)

data_sample$cluster_id<-as.factor(clust$cluster)
```

Checking the distribution of no. of observations assigned to each cluster

```
table(data_sample$cluster_id)
```

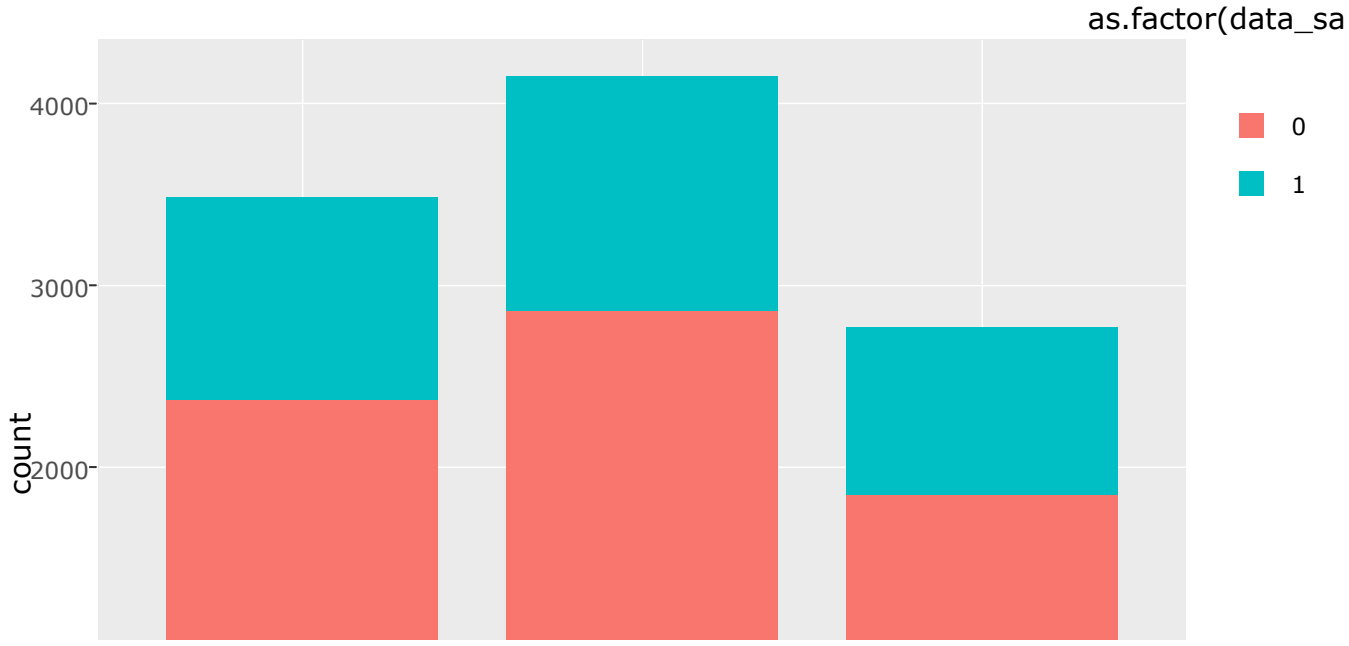
```
##
##    1    2    3
## 3482 4147 2768
```

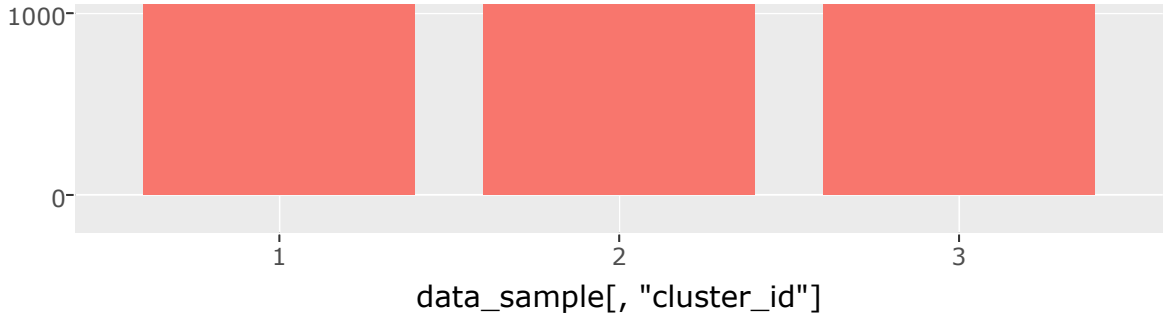
Creating an interactive plot for the distribution of Return variable in each clusters for data sample.

```
library('ggplot2')
library('plotly')

q<-qplot(data_sample[,'cluster_id'],
         fill = as.factor(data_sample[,'repeat'])
        )

ggplotly(q)
```




1000
0
1
2
3
data_sample[, "cluster_id"]