This project is about “**Black Friday Practice Problem**”.

The data can be reached at <https://datahack.analyticsvidhya.com/contest/black-friday/>

Problem Statement

A retail company “ABC Private Limited” wants to understand the customer purchase behaviour (specifically, purchase amount) against various products of different categories. They have shared purchase summary of various customers for selected high volume products from last month.  
The data set also contains customer demographics (age, gender, marital status, city\_type, stay\_in\_current\_city), product details (product\_id and product category) and Total purchase\_amount from last month.

Now, they want to build a model to predict the purchase amount of customer against various products which will help them to create personalized offer for customers against different products.

**Data**

**Variable Definition**

**User\_ID User ID**

**Product\_ID Product ID**

**Gender Sex of User**

**Age Age in bins**

**Occupation Occupation (Masked)**

**City\_Category Category of the City (A,B,C)**

**Stay\_In\_Current\_City\_Years Number of years stay in current city**

**Marital\_Status Marital Status**

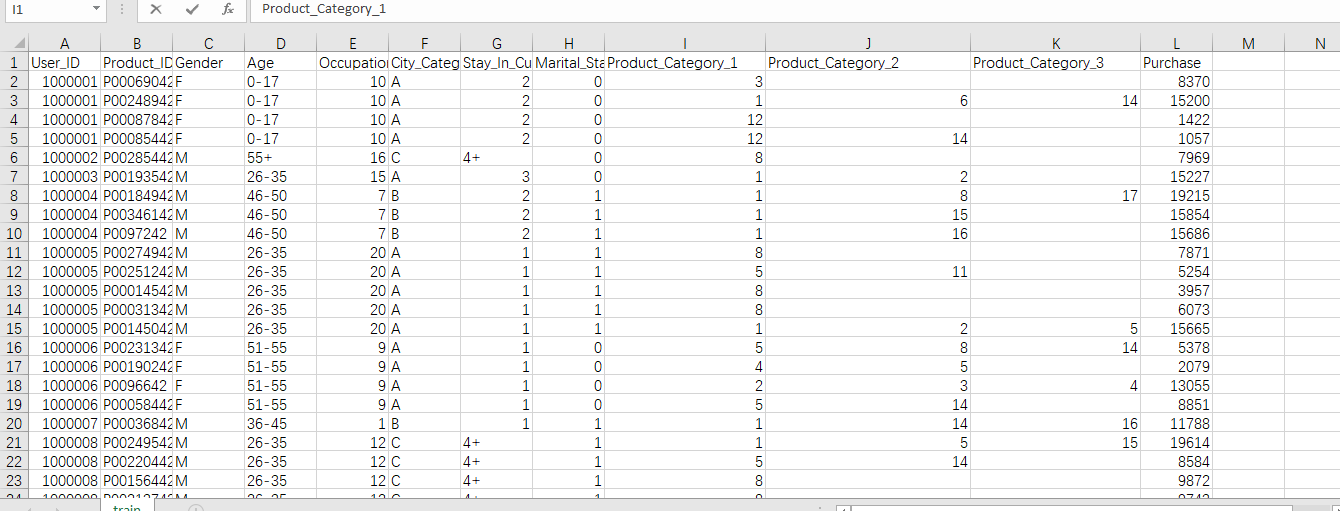
**Product\_Category\_1 Product Category (Masked)**

**Product\_Category\_2 Product may belongs to other category also (Masked)**

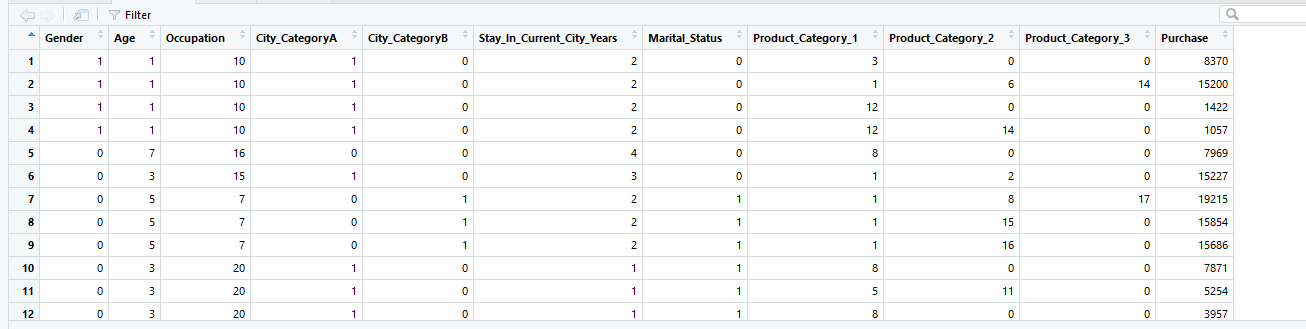
**Product\_Category\_3 Product may belongs to other category also (Masked)**

**Purchase Purchase Amount (Target Variable)**

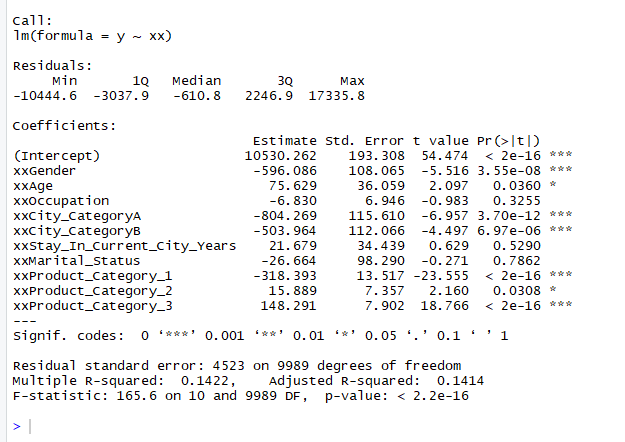
The picture blow is the head of the dataset. There are more than 55w+ observations and 11 variables ignoring the ID of users and products. **Due to the computational limitation of my hardware, I only select the top 10000 observations to fit my model** which should be enough.



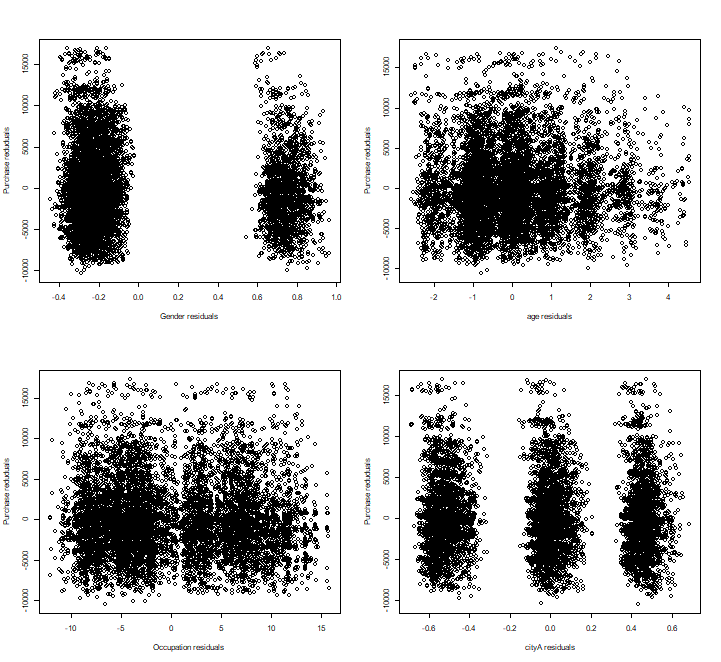
Since the variables “Gender”, ”Age”, “City\_Category”,

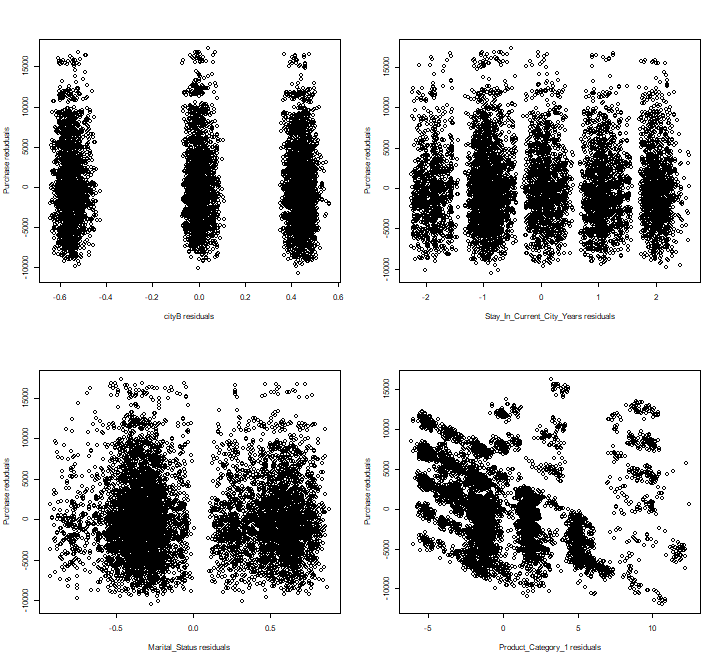
“Stay\_In\_Current\_City\_Years” are not numeric, so I need to do some pretreatment to the initial dataset. For the “City\_Category” whose values are ”A”,”B”,”C”, I use the method of **“dummy variable”** and transform “City\_Category” to two variable “City\_CategoryA”, “City\_CategoryB”. If an observation belongs to City A, then assign 1 to “City\_CategoryA”, and 0 to “City\_CategoryB”; If an observation belongs to City B, then assign 0 to “City\_CategoryA”, and 1 to “City\_CategoryB”. If an observation belongs to City C, then assign 0 to “City\_CategoryA”, and 0 to “City\_CategoryB”. I do the same thing to “Gender”, 1 for female and 0 for male. As for “Age” which is expressed by 7 intervals, **I scored the 7 intervals from 1 to 7** and I did the same thing to “Stay\_In\_Current\_City\_Years”. Finally, I assigned 0 to the miss values. Here is the head of data after the pretreatment.

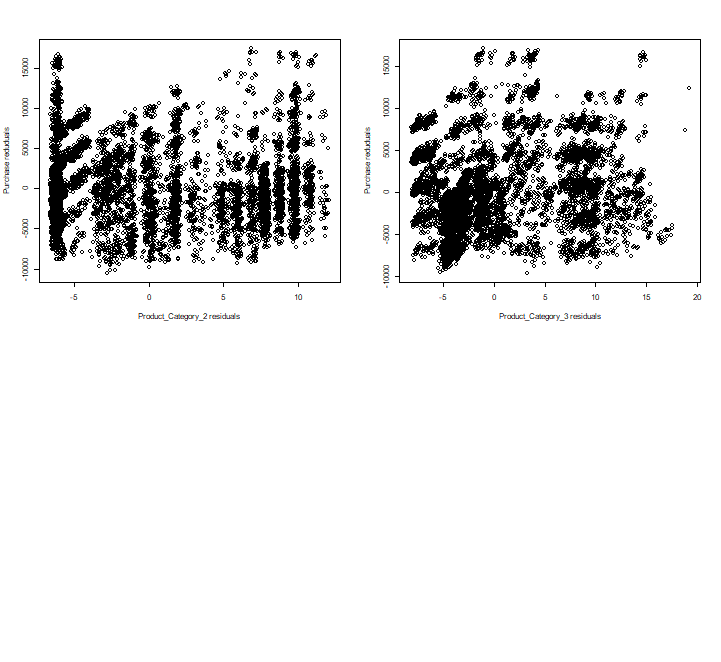
To starting modeling, I do the simple linear regression first. This is the summary of the model:



The small R2 of the model indicates poor fit. So I want to find out the pattern of the relationship between every predict variables and target variable. Here I drew the partial least squares plots:







And I will try to make some transformation to the 10 predict variables depending on the plots in the later time.

Code:

train <- read\_csv("C:/Users/shiru/Desktop/train.csv",

col\_types = cols(Marital\_Status = col\_number(),

Occupation = col\_number(), Product\_Category\_1 = col\_number(),

Product\_Category\_2 = col\_number(),

Product\_Category\_3 = col\_number(),

Purchase = col\_number()))

mm=10000

train[is.na(train)]=0

train=train[1:mm,]

Gender=rep(0,c(mm))

age=rep(0,c(mm))

City\_CategoryA=rep(0,c(mm))

City\_CategoryB=rep(0,c(mm))

for(i in 1:mm){

if(train$Gender[i]=="F") Gender[i]=1

}

for(i in 1:mm){

if(train$Age[i]=="0-17")

age[i]=1

else if(train$Age[i]=="18-25")

age[i]=2

else if(train$Age[i]=="26-35")

age[i]=3

else if(train$Age[i]=="36-45")

age[i]=4

else if(train$Age[i]=="46-50")

age[i]=5

else if(train$Age[i]=="51-55")

age[i]=6

else

age[i]=7

}

for(i in 1:mm){

if(train$City\_Category[i]=="A")

City\_CategoryA[i]=1

else if(train$City\_Category[i]=="B")

City\_CategoryB[i]=1

else 0

}

for(i in 1:mm){

if(train$Stay\_In\_Current\_City\_Years[i]=="4+")

train$Stay\_In\_Current\_City\_Years[i]="4"

}

train$Stay\_In\_Current\_City\_Years=as.numeric(train$Stay\_In\_Current\_City\_Years)

head(as.numeric(train$Gender))

train$Gender=Gender

head(as.numeric(train$Age))

train$Age=age

trainm=as.data.frame(cbind(as.matrix(train[,3:5]),as.matrix(City\_CategoryA),

as.matrix(City\_CategoryB),as.matrix(train[,7:12])))

names(trainm)[4:5]=c("City\_CategoryA","City\_CategoryB")

xx<-as.matrix(trainm[,1:10])

y<-as.matrix(trainm[,11])

out=lm(y~xx)

summary(out)

par(mfrow=c(2,2),cex=0.5)

d<-residuals(lm(y~xx[,-1]))

m<-residuals(lm(xx[,1]~xx[,-1]))

plot(m,d,xlab="Gender residuals",ylab="Purchase reduduals")

d<-residuals(lm(y~xx[,-2]))

m<-residuals(lm(xx[,2]~xx[,-2]))

plot(m,d,xlab="age residuals",ylab="Purchase reduduals")

d<-residuals(lm(y~xx[,-3]))

m<-residuals(lm(xx[,3]~xx[,-3]))

plot(m,d,xlab="Occupation residuals",ylab="Purchase reduduals")

d<-residuals(lm(y~xx[,-4]))

m<-residuals(lm(xx[,4]~xx[,-4]))

plot(m,d,xlab="cityA residuals",ylab="Purchase reduduals")

d<-residuals(lm(y~xx[,-5]))

m<-residuals(lm(xx[,5]~xx[,-5]))

plot(m,d,xlab="cityB residuals",ylab="Purchase reduduals")

d<-residuals(lm(y~xx[,-6]))

m<-residuals(lm(xx[,6]~xx[,-6]))

plot(m,d,xlab="Stay\_In\_Current\_City\_Years residuals",ylab="Purchase reduduals")

d<-residuals(lm(y~xx[,-7]))

m<-residuals(lm(xx[,7]~xx[,-7]))

plot(m,d,xlab="Marital\_Status residuals",ylab="Purchase reduduals")

d<-residuals(lm(y~xx[,-8]))

m<-residuals(lm(xx[,8]~xx[,-8]))

plot(m,d,xlab="Product\_Category\_1 residuals",ylab="Purchase reduduals")

d<-residuals(lm(y~xx[,-9]))

m<-residuals(lm(xx[,9]~xx[,-9]))

plot(m,d,xlab="Product\_Category\_2 residuals",ylab="Purchase reduduals")

d<-residuals(lm(y~xx[,-10]))

m<-residuals(lm(xx[,10]~xx[,-10]))

plot(m,d,xlab="Product\_Category\_3 residuals",ylab="Purchase reduduals")