|  |
| --- |
|  |
| Applied Statistics And Machine Learning For Business Analytics |
|  |
|  |
|  |
| **3/20/2020** |

|  |
| --- |
|  |

**###QUESTION 1**

####a) **Describe the dataset using appropriate plots/curves/charts**

code = read.csv('insurance.csv')#reading the file

code1=table(code$region)

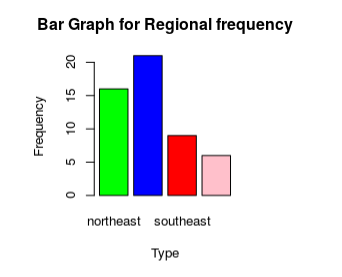
code1

# bar graph

barplot(code1, main = "Bar Graph for Regional frequency",

xlab = "Type", ylab = "Frequency", col = c("Green","Blue", "Red","Pink"),

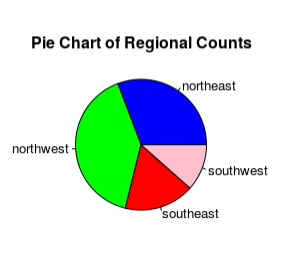
border = "Black")



# pie chart

pie(code1, radius = 1, main = "Pie Chart of Regional Counts",

col = c("Blue", "Green", "Red","Pink"))



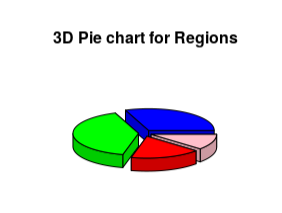
# 3D pie chart

###install.packages('plotrix')

library(plotrix)

pie3D(code1, labels = myClass, explode = .1, main = "3D Pie chart for Regions",

col = c("Blue", "Green", "Red","Pink"))



#Quantitative data

#histogram

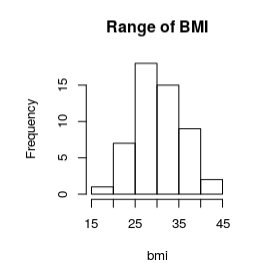
head(code)

layout(matrix(c(1,2,3,4,5,6), 2, 3, byrow = TRUE))

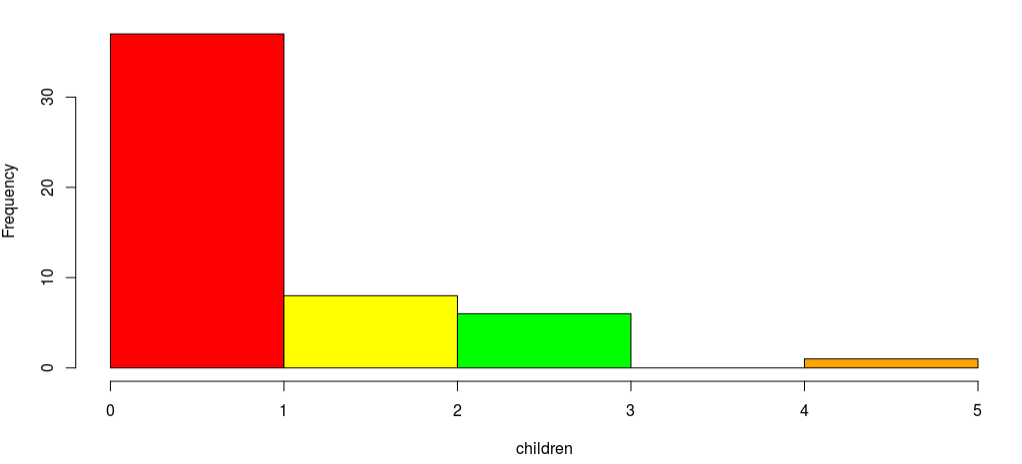
colors = c("red", "yellow", "green", "violet", "orange", "blue", "pink", "cyan",

"purple", "grey", "white","brown",'azure')

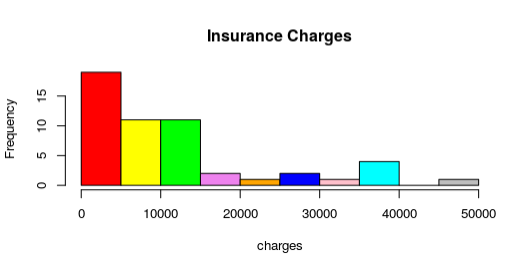
hist(bmi, main = "Range of bmi")



hist(children, col=colors, main = "Number of Children")



hist(charges,col=colors, main="Insurance Charges")



# Stem and Leaf display

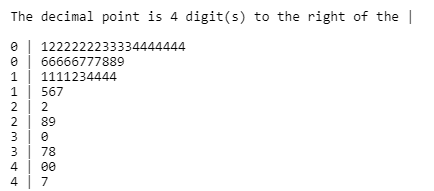
# weight of students

Insurance.charges = code$charges

stem(Insurance.charges)

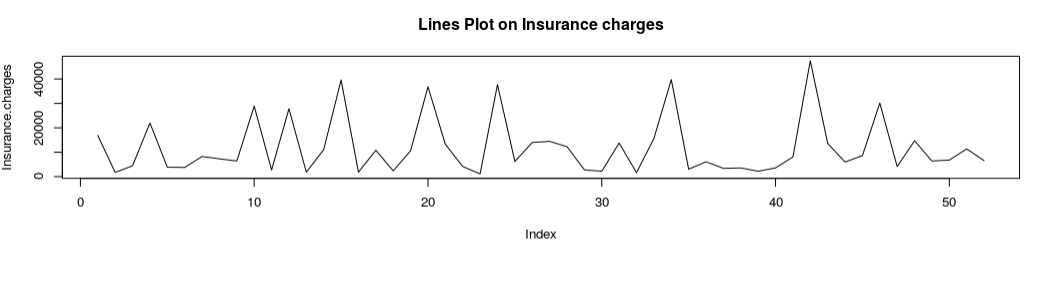
# different plots matrix

layout(matrix(c(1,1,2,3), 2, 2, byrow = TRUE))



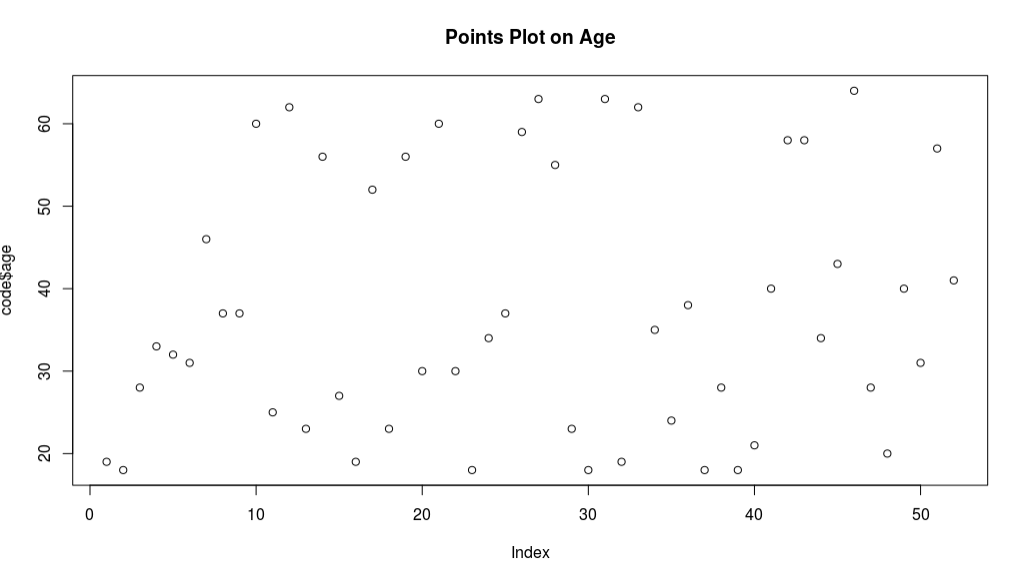
# line plot

plot(Insurance.charges,type = "l", main = "Lines Plot on Insurance charges")



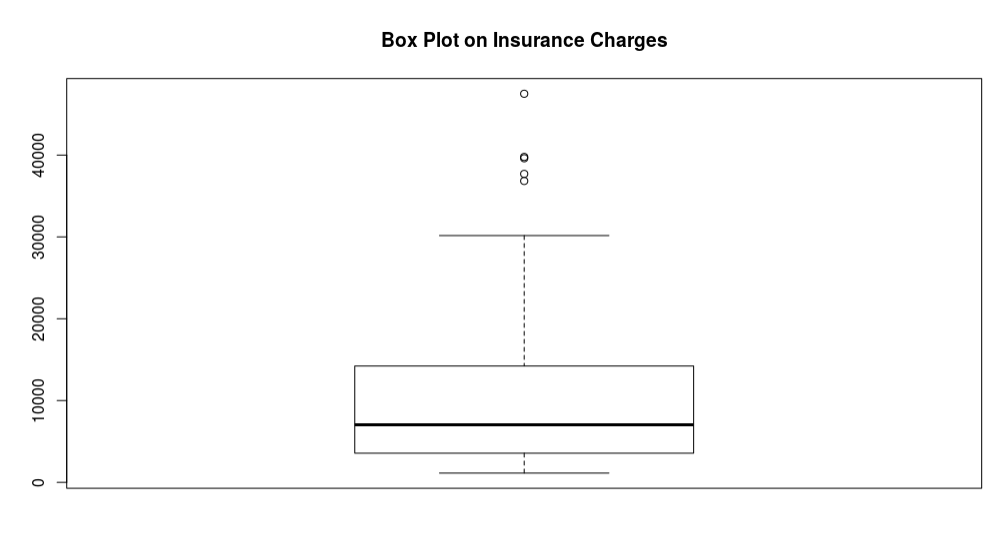
# points plot

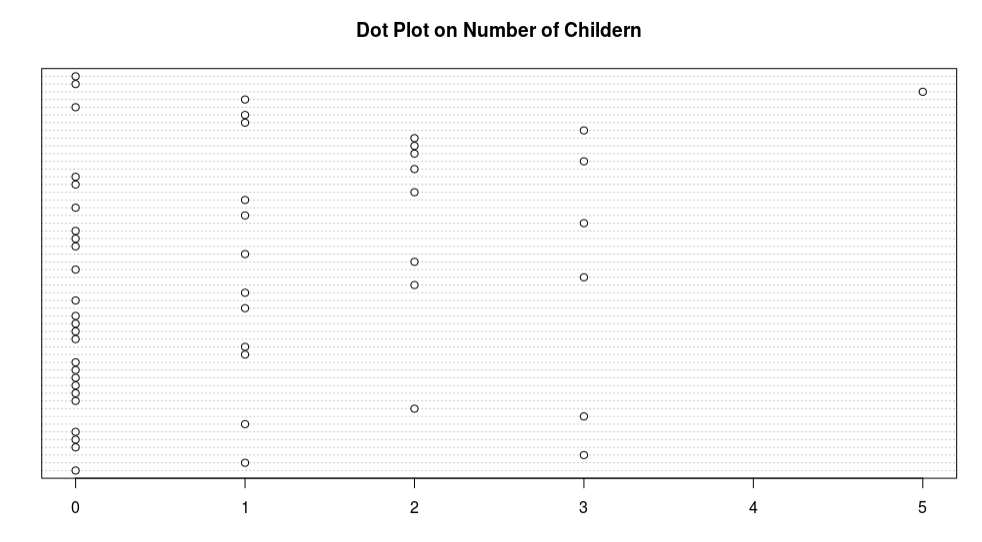
plot(code$age,type = "p", main = "Points Plot on Age")



# Box Plot

boxplot(Insurance.charges,main = "Box Plot on Insurance Charges")

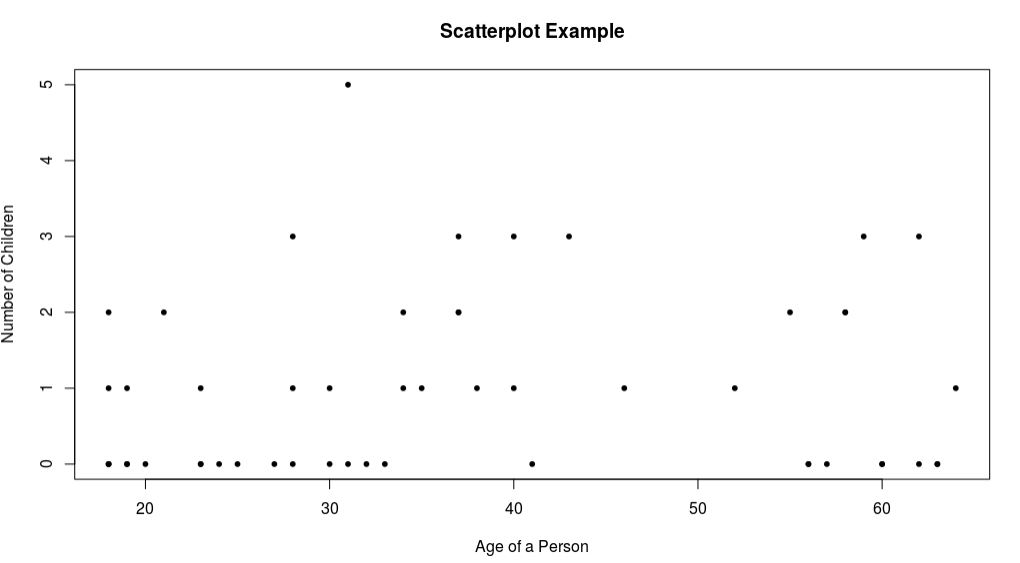


dotchart(code$children,main = "Dot Plot on Number of Childern")

# Simple Scatterplot

plot(age,children, main="Scatterplot Example",

xlab="Age of a Person", ylab="Number of Children", pch=20)



##### b) **Consider one of continuous attributes, and compute central**

data<-read.csv('insurance.csv')

Mean=mean(data$charges)

Mean

Xbar=median(data$charges)

Xbar

Max=max(data$charges)

Max

Min=min(data$charges)

Min

Range<-range(data$charges)

Range

S=sd(data$charges)

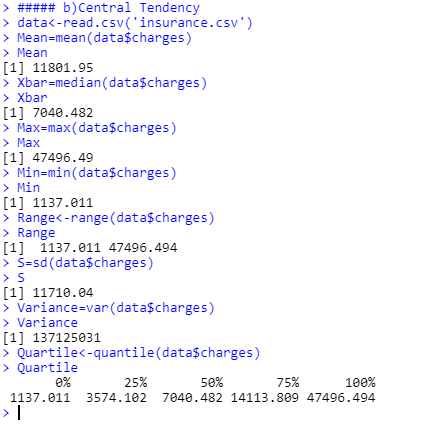
S

Variance=var(data$charges)

Variance

Quartile<-quantile(data$charges)

Quartile



#####(c)**For a particular variable of the dataset, use Chebyshev's rule, #and propose one-sigma interval. Based on your proposed interval, #specify the outliers if any.**

######### one sigma and outlier

data<-read.csv('insurance.csv',header=TRUE)

xbar= mean(data$age)

xbar

s= sd(data$age)

s

xL=xbar-1\*s

xL

xU=xbar+1\*s

xU

int=c(xL,xU)

int

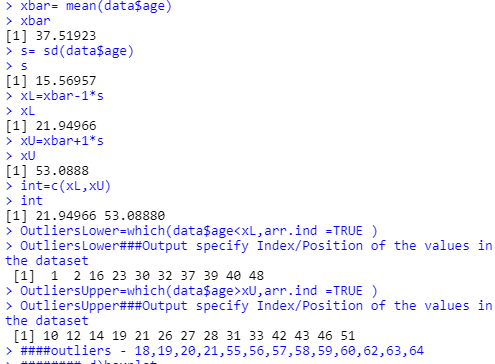
OutliersLower=which(data$age<xL,arr.ind =TRUE )

OutliersLower###Output specify Index/Position of the values in the dataset

OutliersUpper=which(data$age>xU,arr.ind =TRUE )

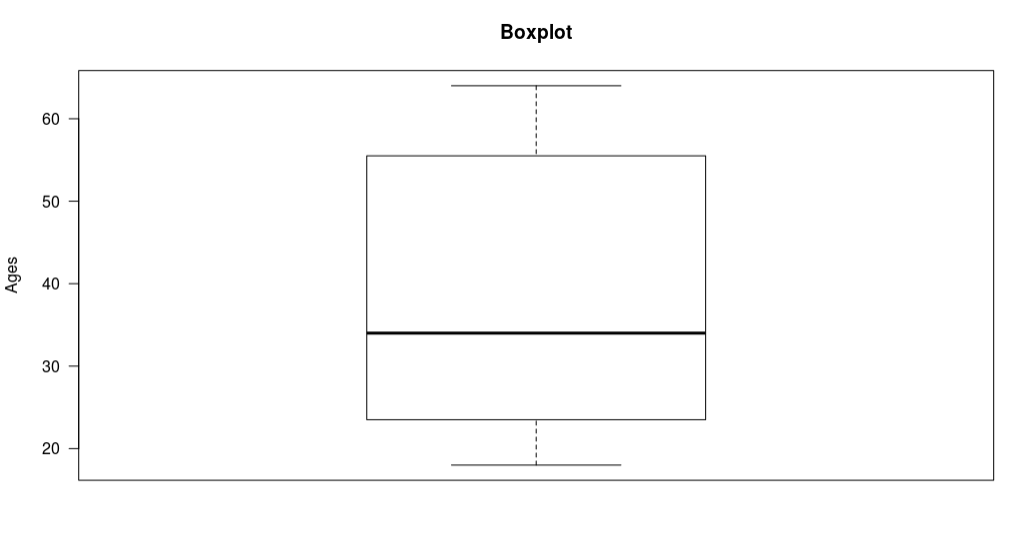
OutliersUpper###Output specify Index/Position of the values in the dataset

####outliers - 18,19,20,21,55,56,57,58,59,60,62,63,64



######## d) **Explain how the box-plot technique can be used to detect #outliers. Apply this technique for one attribute of the dataset**

boxplot(data$age, main="Boxplot",ylab="Ages",ylim=c(18,64),las=1)



**Question 2**

**# ###a)** **Select four variables of the dataset, and propose an appropriate #probability model to quantify uncertainty of each #variable.**

**data= read.csv('insurance.csv')#reading the file**

**data**

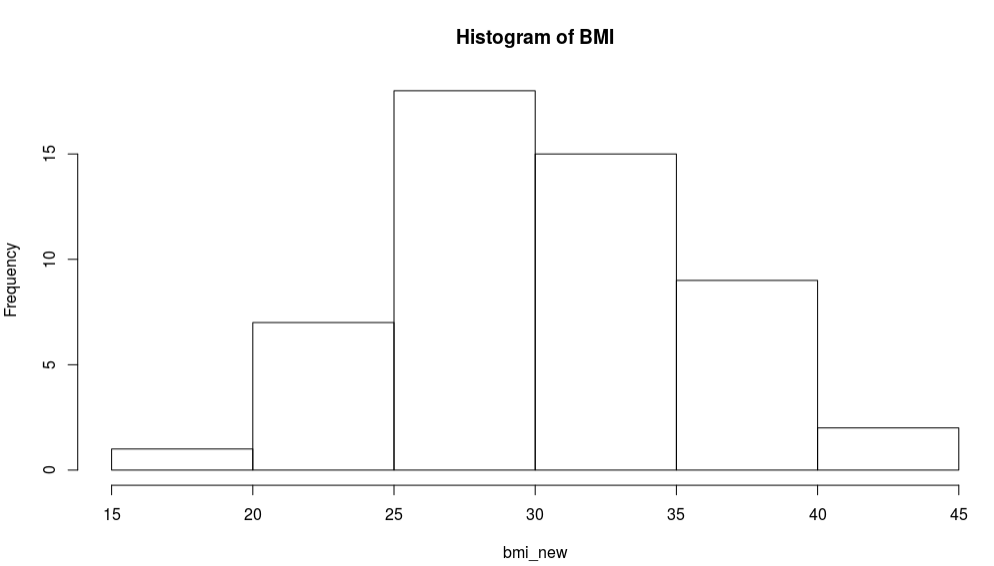
**# First Variable selected -> BMI**

**head(data$bmi) # output of height attribute is continuous**

**# due to continuous values using histogram to predict probability model**

**bmi\_new<-na.omit(data$bmi)**

**hist(bmi\_new, main = "Histogram of BMI")**



**# the output of histogram looks like a normal distribution**

**####b)** **For each model in part (a), estimate the parameters of model.**

**m1=mean(bmi\_new)**

**s1=sd(bmi\_new)**

**m1**

**s1**

**print(c('Miyu for this model is: ',m1))**

**print(c('Sigma for this model is: ',s1))**

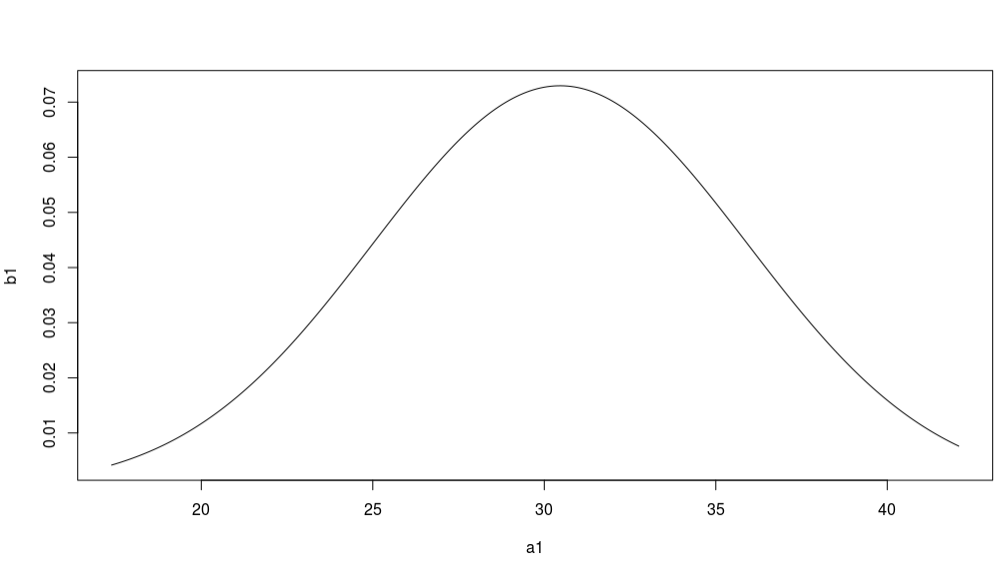
**####c)** **find the prediction for each attribute**

**##probability when x=30**

**a1=seq(min(bmi\_new),max(bmi\_new),0.1)**

**b1=dnorm(a1,m1,s1)**

**plot(a1,b1,'l')**



**p1<-pnorm(30,m1,s1,lower.tail = T)**

**p1**

**#hence we can say that 46.63 % of people have a bmi of 30**

**# ###Second Variable selected -> Age**

**head(data$age) # output of height attribute is Discrete**

**# due to random numbers we will use Poisson Model**

**age\_new<-na.omit(data$age)**

**####b)** **For each model in part (a), estimate the parameters of model.**

**l2=mean(age\_new)**

**print(c('lambda for this model is l2',l2))**

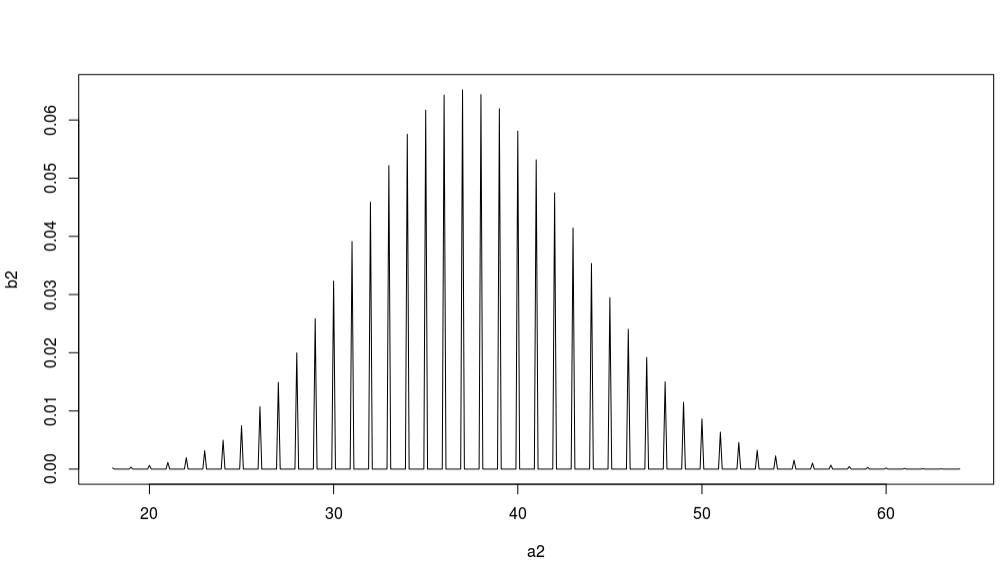
**#find probablity when X=38**

**####c)** **find the prediction for each attribute**

**a2=seq(min(age\_new),max(age\_new),0.1)**

**b2=dpois(a2,l2)**

**plot(a2,b2,'l')**



**p2<-ppois(38,l2,lower.tail = T)**

**p2**

**#hence we can say that 57 % of prople have a age of 38**

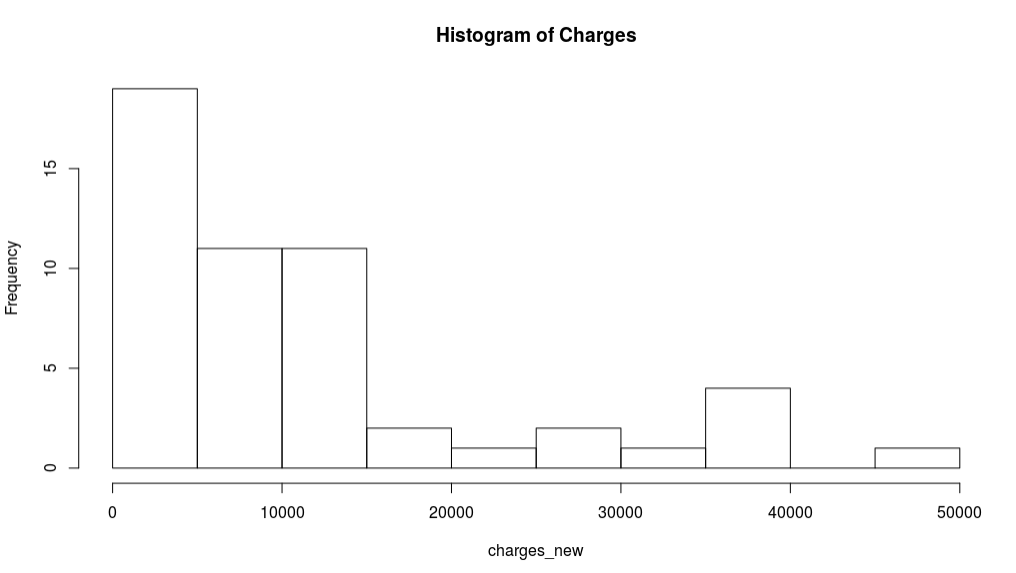
**#### Third Variable selected -> Charges**

**head(data$charges) # output of Charges attribute is continuous**

**# due to continuous values using histogram to predict probability model**

**charges\_new<-na.omit(data$charges)**

**hist(charges\_new, main = "Histogram of Charges")**



**# the output of histogram looks like exponential distribution**

**####b)for each variable estimate parameter of model**

**m3=mean(charges\_new)**

**m3**

**l3=(1/m3)**

**print(c('Lambda for this model is: ',l3))**

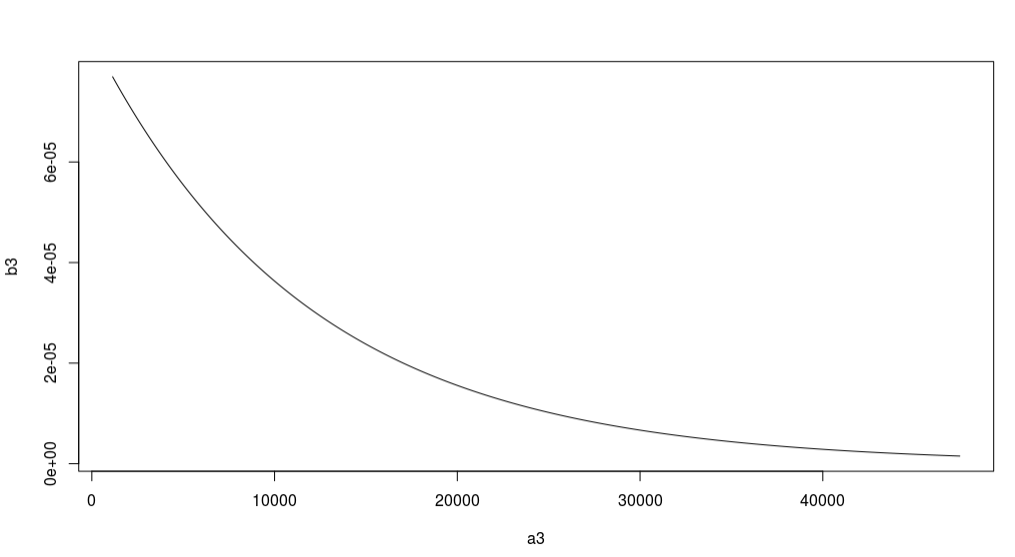
**###c) find the prediction for each attribute**

**#plot the pdf of expotential model with lamda=l3 and alter find probability when x=12000**

**a3=seq(min(charges\_new),max(charges\_new),0.1)**

**b3=dexp(a3,l3)**

**plot(a3,b3,'l')**



**p3=pexp(12000,l3,lower.tail = T)**

**p3**

**#hence we can say that 64% of people have a charges of 12000**

**#### Fourth Variable of the Dataset is Children**

**ch\_new= na.omit(data$children)**

**print(ch\_new) # multiple discrete value (0,1,2,3,5)**

**# due to four different integers, we will use multinomial distribution here**

**ch = length(ch\_new) # total count**

**ch1 = length(ch\_new[ch\_new==0])**

**ch2 = length(ch\_new[ch\_new==1])**

**ch3 = length(ch\_new[ch\_new==2])**

**ch4 = length(ch\_new[ch\_new==3])**

**ch5 = length(ch\_new[ch\_new==5])**

**# e.g. 12 trails where rank 1 noticed 6 times, 2 noticed 1 time and**

**# 3 noticed 3 times 4 noticed 2 times**

**multi.n = 52**

**multi.x = c(25,12,8,6,1)**

**multi.p = c(ch1/ch,ch2/ch,ch3/ch,ch4/ch,ch5/ch)**

**multi.pmf = dmultinom(multi.x,multi.n,multi.p)**

**# generate 300 random samples from multinomial dist**

**# where size=5, Then use the dataset to estimate multi.pmf1**

**set.seed(123) # for same random number**

**multi.rmf=rmultinom(300,5,multi.p)**

**multi.rmf1=rmultinom(300,0:5,multi.p)**

**multi.rmf.matrix=t(multi.rmf) # Matrix Transpose**

**dim(multi.rmf.matrix)**

**p\_hat2=mean(data$children)**

**p\_hat2**

**Question 3**

**data<-read.csv('insurance.csv',header=TRUE)#reading the file**

**x1=data$charges**

**y=data$smoker**

**dataset <- na.omit(data.frame( x1,y)) # remove missing values**

**####a)** **Suggest an appropriate GLM to model ouput to input variables**

**####Modeling the relationship between insurance charges and smoker using logistic regression**

**#######input variables charges**

**########Output variable smoker**

**#######Predicting the Smoker based on independent variables charges**

**library(caTools) # useful to split data to training and test datasets**

**acc=0**

**mc=1000**

**for(i in 1:mc){**

**set.seed(1234)**

**####b)** **Split the dataset into 80% as a trainset and 20% testset, then model #the trainset by fitting your proposed GLM.**

**#spliting the dataset into 80% and 20%**

**split=sample.split(y,SplitRatio=0.8)**

**trainset=subset(dataset, split=="TRUE") # training dataset**

**testset=subset(dataset,split=="FALSE") # test dataset**

**trainset**

**testset**

**dim(testset)**

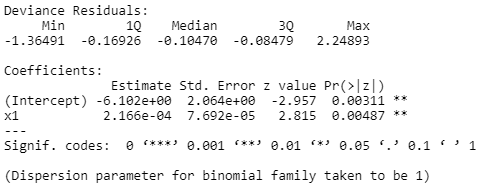
**####Fitting logistic regression to the traing dataset**

**trainset.glm <- glm(formula=y ~x1, family="binomial",trainset)**

**summary(trainset.glm)**

**####c) Specify the significant variables on the output variable**

**###both intercept and charges values are significant**



**xnew=c(1,21984.47)**

**hyptest=sum(coef(trainset.glm)\*xnew)**

**hyptest**

**###d)** **Predict the output of the test dataset using the trained model. #And #provide the functional form of the optimal predictive model.**

**phati=1/(1+exp(-hyptest))#predicted value of the probability of**

**phati**

**###if phati>0.5 then y=1(smoker) else y=0(Non Smoker)**

**####d)########predictng testdata set using traing datset model**

**Prob\_smok\_pred=predict(trainset.glm,type='response',newdata=testset[-3])**

**Prob\_smok\_pred**

**#independent variable charges is significant variable(alpha=0.05) on the output variable smoker**

**ypredict=ifelse(Prob\_smok\_pred>0.5,'yes','no')#converting probability values into Yes or No**

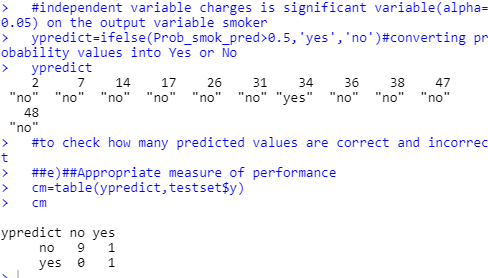
**ypredict**

**#to check how many predicted values are correct and incorrect**

**####e) Propose the appropriate measure of performance to evaluate #the #model and compute it for your derived model.**

**cm=table(ypredict,testset$y)**

**cm**



**#accuracy**

**accuracy=mean(ypredict == testset$y) # correctness of prediction**

**acc=acc+(1/mc)\*accuracy**

**}**

**Acc**

