**Laptop price prediction**

**Code**

data <- read.csv("C:/dataset/lp.csv")

print(data)

summary(data)

names(data)

table1<-table(data$Cpu)

barplot(table1,col=c(1,2,3,4,5),las=2,xlab = "Company",ylab="",main="barplot visualization of product")

#levels(data$Cpu)

CN<- table(data$Cpu)

pie(CN,main="Pie chart Visualization of company")

barplot(data$Cpu,main = "Cpu",ylab="Price")

#data<- table(data$Cpu)

data

#predicting result

Input<-data[c("Cpu")]

Output<-data$Price

regresser1<-lm(data$Price ~ data$Cpu,data1 = data)

regresser1

new <- data.frame('Price')

new

Cpu<-"Intel Core i5 3.1GHz"

predict(regresser1,new)

pred<-predict(regresser1,data.frame (Cpu = (5)))

print(pred)

#multiple

dataset <- read.csv("C:/dataset/lp.csv")

print(dataset)

summary(dataset)

multi.fit = lm(data$Price~data$Cpu, data=dataset)

multi.fit

**ui.r**

library(shiny)

source("data.R")

lstCategorie <-

getlstCategorie(getGroupedData(getTweetTraining(openDBSentiment())))

shinyUI (pageWithSidebar(

headerPanel("Tweet Sentiment"),

sidebarPanel(

selectInput("dataset", "Selecteer bank:"

, choices = lstCategorie

, selected = "Rabobank"

)

),

mainPanel(

htmlOutput("view")

)

)

Server.r

# **server.R**

library(googleVis)

library(shiny)

shinyServer(function(Cpu, Price) {

datasetInput <- reactive({

getFilteredBarFrame(data$Cpu)

})

output$view <- renderGvis({

gvisColumnChart(datasetInput()

)

})

})