**Lab 8: Logistic Regression**

**Problem statement:**

Refer attached bank.csv dataset. Develop a logistic regression classification model as:

* Class variable: repaid
* Independent variable: age and salary
* With summary command observe the results
* Display the probability of each data record
* Calculate and display the assigned class with respect to cut off value 0.5.
* Calculate and display the confusion matrix
* Calculate the accuracy of model
* Calculate the error rate of model
* Calculate the recall of model
* Calculate the precision of model

**Source Code:**

#Author: Ashish Upadhyay

#Branch: Computer Science and Engineering

#Semester: 6th

#Dr. SP Mukherjee International Institute of Information Technology, Naya Raipur

#Subject: Machine Learning Lab 8

#Task: Logistic Regression Implementation - Part I

setwd("C:/Users/Ashish Upadhyay/Documents/Semester6/MachineLearning/Lab Programs")

getwd()

train <- read.csv("bank.csv")

nrow(train)

head(train)

#install.packages('caTools')

library(caTools)

set.seed(88)

split <- sample.split(train$repaid, SplitRatio = 0.75)

#get training and test data

dresstrain <- subset(train, split == TRUE)

dresstest <- subset(train, split == FALSE)

#Logistic Regression Model

model <- glm (repaid ~ ., data = dresstrain, family = binomial)

#Summary

summary(model)

#Probability

probability <- predict(model, type = 'response')

probability

#Confusion Matrix (Cut-off value = 0.5)

con\_mat <- table(dresstrain$repaid, probability > 0.5)

con\_mat

#Accuracy

accuracy <- ((con\_mat[1, 1] + con\_mat[2, 2])/(con\_mat[1, 1] + con\_mat[1, 2] + con\_mat[2, 1] + con\_mat[2, 2])) \* 100

accuracy

#Precision

precision <- ((con\_mat[2, 2]) / (con\_mat[1, 2] + con\_mat[2, 2])) \* 100

precision

#Recall

recall <- (con\_mat[2, 2] / (con\_mat[2, 1] + con\_mat[2, 2])) \* 100

recall

#Error Rate

error\_rate <- (con\_mat[1, 1] / (con\_mat[1, 1] + con\_mat[1, 2] + con\_mat[2, 1] + con\_mat[2, 2])) \* 100

error\_rate

#F1 score

f1 <- (2 \* precision \* recall) / (precision + recall)

f1

**Output:**

> #Author: Ashish Upadhyay

> #Branch: Computer Science and Engineering

> #Semester: 6th

> #Dr. SP Mukherjee International Institute of Information Technology, Naya Raipur

> #Subject: Machine Learning Lab 8

> #Task: Logistic Regression Implementation - Part I

>

>

> setwd("C:/Users/Ashish Upadhyay/Documents/Semester6/MachineLearning/Lab Programs")

> getwd()

[1] "C:/Users/Ashish Upadhyay/Documents/Semester6/MachineLearning/Lab Programs"

>

> train <- read.csv("bank.csv")

> nrow(train)

[1] 2952

> head(train)

age salary repaid

1 70 55 1

2 96 51 1

3 86 71 1

4 87 67 1

5 77 87 1

6 74 87 1

>

> #install.packages('caTools')

> library(caTools)

>

> set.seed(88)

> split <- sample.split(train$repaid, SplitRatio = 0.75)

>

> #get training and test data

> dresstrain <- subset(train, split == TRUE)

> dresstest <- subset(train, split == FALSE)

> #Logistic Regression Model

> model <- glm (repaid ~ ., data = dresstrain, family = binomial)

>

> #Summary

> summary(model)

Call:

glm(formula = repaid ~ ., family = binomial, data = dresstrain)

Deviance Residuals:

Min 1Q Median 3Q Max

-2.63965 -0.14935 0.08644 0.34941 3.14588

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) -16.044781 0.790730 -20.29 <2e-16 \*\*\*

age 0.160963 0.007738 20.80 <2e-16 \*\*\*

salary 0.117573 0.006353 18.50 <2e-16 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 2756.6 on 2213 degrees of freedom

Residual deviance: 1130.5 on 2211 degrees of freedom

AIC: 1136.5

Number of Fisher Scoring iterations: 7

>

> #Probability

> probability <- predict(model, type = 'response')

> #Confusion Matrix (Cut-off value = 0.5)

> con\_mat <- table(dresstrain$repaid, probability > 0.5)

> con\_mat

FALSE TRUE

0 547 149

1 102 1416

>

> #Accuracy

> accuracy <- ((con\_mat[1, 1] + con\_mat[2, 2])/(con\_mat[1, 1] + con\_mat[1, 2] + con\_mat[2, 1] + con\_mat[2, 2])) \* 100

> accuracy

[1] 88.66305

>

> #Precision

> precision <- ((con\_mat[2, 2]) / (con\_mat[1, 2] + con\_mat[2, 2])) \* 100

> precision

[1] 90.47923

>

> #Recall

> recall <- (con\_mat[2, 2] / (con\_mat[2, 1] + con\_mat[2, 2])) \* 100

> recall

[1] 93.28063

>

> #Error Rate

> error\_rate <- (con\_mat[1, 1] / (con\_mat[1, 1] + con\_mat[1, 2] + con\_mat[2, 1] + con\_mat[2, 2])) \* 100

> error\_rate

[1] 24.70641

>

> #F1 score

> f1 <- (2 \* precision \* recall) / (precision + recall)

> f1

[1] 91.85858