# ImplementLinearandLogisticRegression

**AIM:**

ToimplementLinearandLogisticRegressionusingRprogramminginRStudio.**a)Linear**

# regression

#Sampledata

heights<-c(150,160,165,170,175,180,185)weights

<-c(55,60,62,68,70,75, 80)

#Createadataframedata<-data.frame(heights,weights)

#Fitalinearregressionmodellinear\_model

<-lm(weights~heights,data=data)

#Printthesummaryofthemodelprint(summary(linear\_model))

#Plottingthedataandregressionlineplot(data$heights,data$weights,

main = "Linear Regression: Weight vs.Height", xlab="Height (cm)", ylab

="Weight(kg)", pch

=19,col="blue")

# Add regression lineabline(linear\_model,col="red",lwd=2)

# OUTPUT:



**b)Logisticregression**

#Loadthedatasetdata(mtcars)

#Convert'am'toafactor(categoricalvariable)mtcars$am <-factor(mtcars$am,levels= c(0, 1), labels = c("Automatic", "Manual"))

#Fitalogisticregressionmodellogistic\_model<-glm(am

~mpg,data=mtcars,family=binomial)

#Printthesummaryofthemodelprint(summary(logistic\_model))

# Predict probabilities for the logistic modelpredicted\_probs<-predict(logistic\_model,type=

"response")

#Displaythepredictedprobabilitiesprint(predicted\_probs)

#Plottingthedataandlogisticregressioncurveplot(mtcars$mpg,as.numeric(mtcars$am)-1,

main="LogisticRegression:Transmissionvs.MPG",xlab="MilesPerGallon(mpg)", ylab

="ProbabilityofManualTransmission", pch =19,col = "blue")

# Add the logistic regression curvecurve(predict(logistic\_model,data.frame(mpg=x),type="response"), add=TRUE, col = "red", lwd =2)

# OUTPUT:





**RESULT:**

ThustheimplementationLinearandLogisticRegressionusingRprogramminginRStudiohavebeen successfully executed.