**week-9a**

> height<-c(150,160,170,180,190)

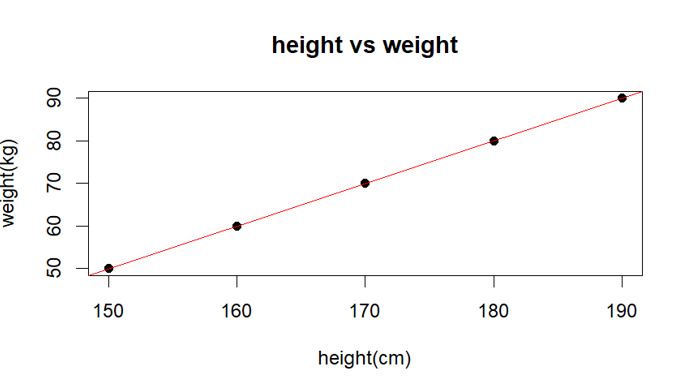
> weight<-c(50,60,70,80,90)

> model<-lm(weight~height)

> plot(height,weight,main="height vs weight",xlab="height(cm)",ylab="weight(kg)",pch=16)

> abline(model,col="red")

**Output:**



**Week-9b:**

> data<-mtcars[,c("mpg","wt","disp","hp")]

> model<-lm(mpg~wt+disp+hp,data=data)

> summary(model)

**Output:**

Call:

lm(formula = mpg ~ wt + disp + hp, data = data)

Residuals:

Min 1Q Median 3Q Max

-3.891 -1.640 -0.172 1.061 5.861

Coefficients:

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

(Intercept) 37.105505 2.110815 17.579 < 2e-16 \*\*\*

wt -3.800891 1.066191 -3.565 0.00133 \*\*

disp -0.000937 0.010350 -0.091 0.92851

hp -0.031157 0.011436 -2.724 0.01097 \*

---

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

Residual standard error: 2.639 on 28 degrees of freedom

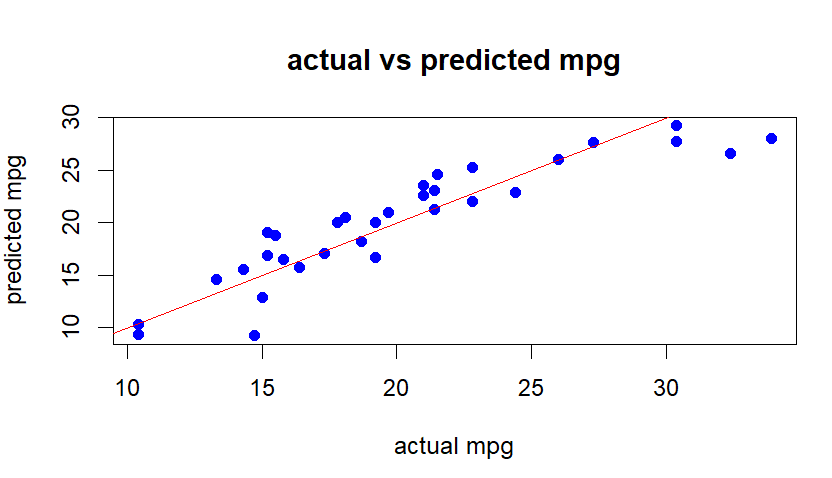
Multiple R-squared: 0.8268, Adjusted R-squared: 0.8083

F-statistic: 44.57 on 3 and 28 DF, p-value: 8.65e-11

**code for graph:**

> plot(data$mpg,predict(model),xlab="actual mpg",ylab="predicted mpg",main="actual vs predicted mpg",pch=16,col="blue")

> abline(a=0,b=1,col="red")



**Week-9c:**

> age<-c(20,30,40,50,60)

> buy<-c(0,1,0,1,1)

> model<-glm(buy~age,family=binomial)

> summary(model)

**Output:**

Call:

glm(formula = buy ~ age, family = binomial)

Coefficients:

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

(Intercept) -3.73901 3.71955 -1.005 0.315

age 0.10904 0.09748 1.119 0.263

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 6.7301 on 4 degrees of freedom

Residual deviance: 4.8439 on 3 degrees of freedom

AIC: 8.8439

Number of Fisher Scoring iterations: 4

**For graph:**

> predicted\_probs<-predict(model,newdata=data.frame(age=c(25,35)),type="response")

> print(predicted\_probs)

1 2

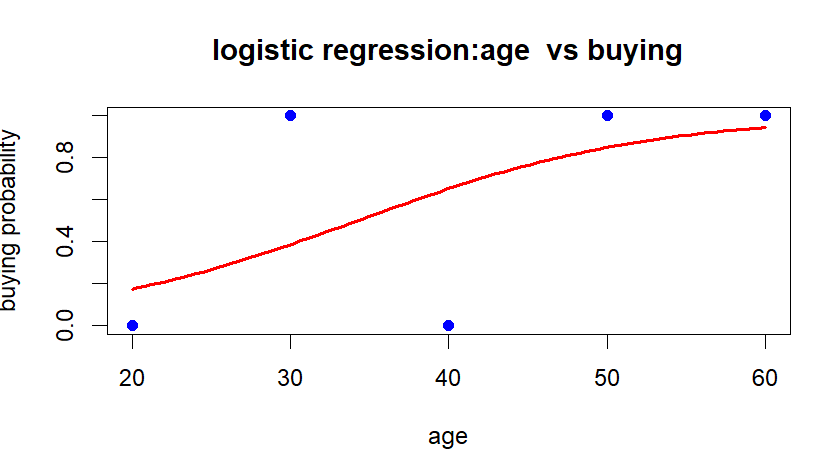
0.2664033 0.5193596

> age\_seq<-seq(min(age),max(age),length.out=100)

> fitted\_probs<-predict(model,newdata=data.frame(age=age\_seq),type="response")

> plot(age,buy,main="logistic regression:age vs buying",xlab="age",ylab="buying probability",pch=16,col="blue",ylim=c(0,1))

> lines(age\_seq,fitted\_probs,col="red",lwd=2)



**Week-10a:**

> sales<-c(100,120,140,160,180,200)

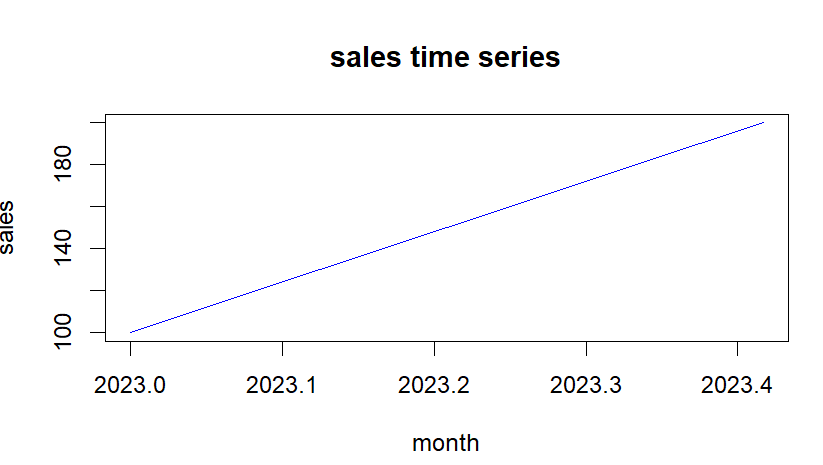
> time\_series<-ts(sales,frequency=12,start=c(2023,1))

> plot(time\_series,main="sales time series",xlab="month",ylab="sales",col="blue")

> linear\_model<-lm(sales~c(1:6))

> abline(linear\_model,col="red")

**Output:**

****

**Week-10b:**

|  |
| --- |
| > xvalues<-c(1.6,2.1,2,2.23,3.71,3.25,3.4,3.86,1.19,2.21)  > yvalues<-c(5.19,7.43,6.94,8.11,18.75,14.88,16.06,19.12,3.21,7.58)  > png(file="nls3.png")  > plot(xvalues,yvalues)  > model<-nls(yvalues~b1\*xvalues^2+b2,start=list(b1=1,b2=3))  > new.data<-data.frame(xvalues=seq(min(xvalues),max(xvalues),len=10))  > lines(new.data$xvalues,predict(model,newdata=new.data))  > dev.off()  RStudioGD  2  > print(sum(resid(model)^2))  [1] 1.081935  > print(confint(model))  Waiting for profiling to be done...  2.5% 97.5%  b1 1.137708 1.253135  b2 1.497364 2.496484 |
|  |
| |  | | --- | |  | |

**Week-10c:**

data("iris")

install.packages("caret")

install.packages("C50")

library(caret)

library(C50)

set.seed(7)

inTraininglocal<-createDataPartition(iris$Species,p=.70,list=F)

training<-iris[inTraininglocal,]

testing<-iris[-inTraininglocal,]

model<-C5.0(Species~.,data=training)

summary(model)

pred<-predict.C5.0(model,testing[,-5]) #type="prob"

a<-table(testing$Species,pred)

sum(diag(a))/sum(a)

plot(model)

**week-11a:**

**dnorm:**

> x<-seq(-10,10,by=.1)

> y<-dnorm(x,mean=2.5,sd=0.5)

> png(file="dnorm.png")

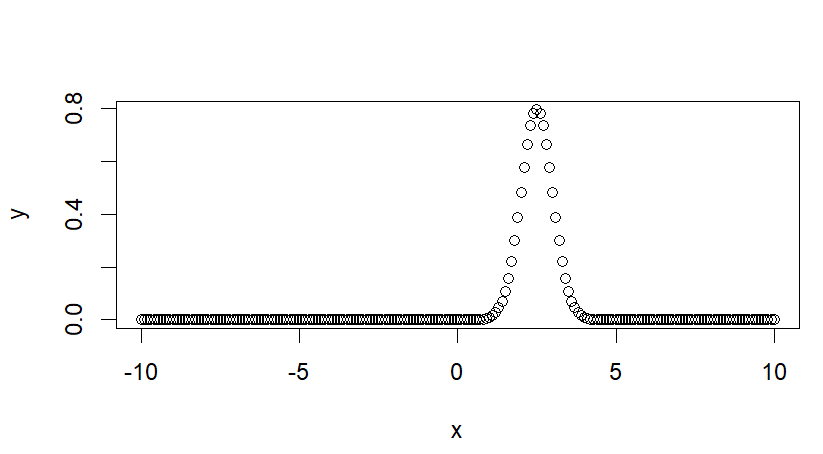
> plot(x,y)

> dev.off()

RStudioGD

2

> plot(x,y)



**Pnorm:**

> x<-seq(-10,10,by=.2)

> y<-pnorm(x,mean=2.5,sd=2)

> png(file="pnorm.png")

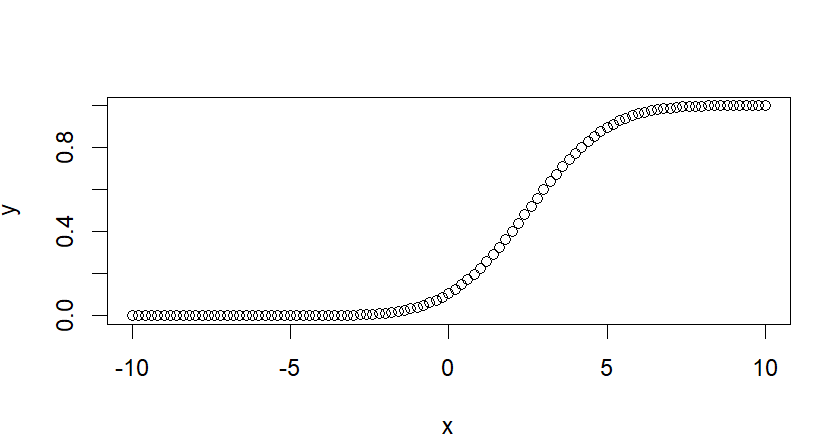
> plot(x,y)

> dev.off()

RStudioGD

2

>plot(x,y)

****

**Qnorm:**

> x<-seq(0.1,by=0.02)

> y<-qnorm(x,mean=2,sd=1)

> png(file="qnorm.png")

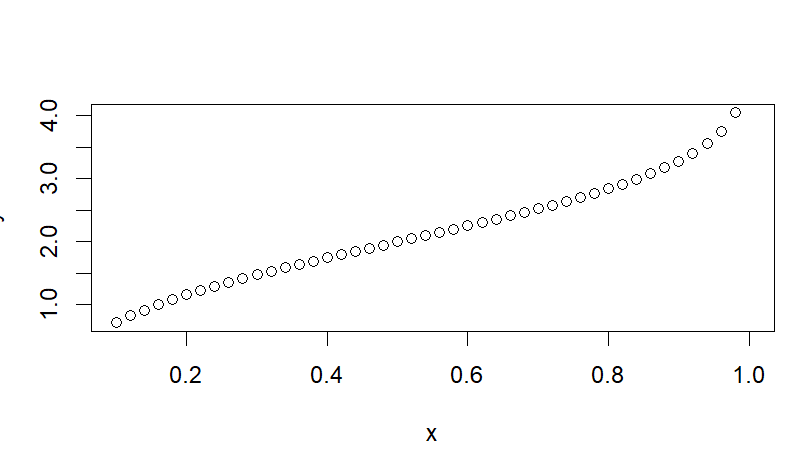
> plot(x,y)

> dev.off()

RStudioGD

2

> plot(x,y)



**Rnorm:**

> y<-rnorm(50)

> png(file="rnorm.png")

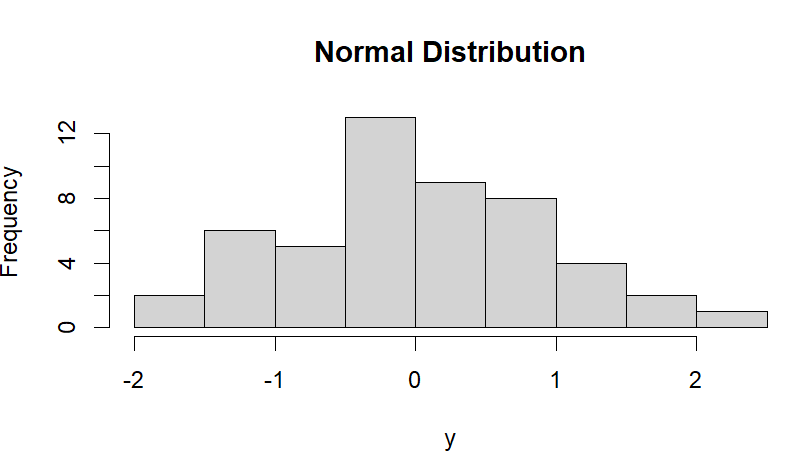
> hist(y,main="Normal Distribution")

> dev.off()

RStudioGD

2

> hist(y,main="Normal Distribution")

****

**Week-11b:**

**Dbinom:**

> x<-seq(0,50,by=1)

> y<-dbinom(x,50,0.5)

> png(file="dbinom.png")

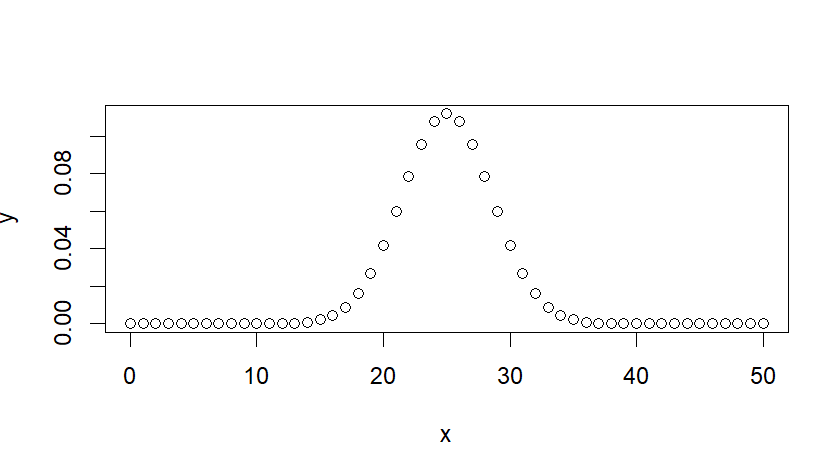
> plot(x,y)

> dev.off()

RStudioGD

2

> plot(x,y)



**Pbinom:**

> x<-pbinom(26,51,0.5)

> print(x)

[1] 0.610116

**Qbinom:**

> x<-qbinom(0.25,51,1/2)

> print(x)

[1] 23

**Rbinom:**

> x<-rbinom(8,150,.4)

> print(x)

[1] 67 75 51 62 55 64 58 62

**Week12a:**

> library("MASS")

> car\_data<-data.frame(Cars93$AirBags,Cars93$Type)

> car\_data=table(Cars93$AirBags,Cars93$Type)

> print(car\_data)

Compact Large Midsize Small Sporty Van

Driver & Passenger 2 4 7 0 3 0

Driver only 9 7 11 5 8 3

None 5 0 4 16 3 6

> print(chisq.test(car\_data))

Pearson's Chi-squared test

data: car\_data

X-squared = 33.001, df = 10, p-value = 0.0002723

**Week-12b:**

> x<-c(0.593,0.142,0.329,0.691,0.231,0.793,0.519,0.392,0.418)

> t.test(x,alternative="greater",mu=0.3)

One Sample t-test

data: x

t = 2.2051, df = 8, p-value = 0.02927

alternative hypothesis: true mean is greater than 0.3

95 percent confidence interval:

0.3245133 Inf

sample estimates:

mean of x

0.4564444

**Week-12c:**

> x<-c(18,19,22,25,27,28,41,45,51,55)

> y<-c(14,15,15,17,18,22,25,25,27,34)

> print(var.test(x,y))

F test to compare two variances

data: x and y

F = 4.3871, num df = 9, denom df = 9, p-value = 0.03825

alternative hypothesis: true ratio of variances is not equal to 1

95 percent confidence interval:

1.089699 17.662528

sample estimates:

ratio of variances

4.387122