2) Delivery\_time -> Predict delivery time using sorting time

**Ans :**

**R Code :**

## Simple Linear Regression

########## Delivery Time Data Set #########

delivery <- read.csv('D:\\Data Science\\Excelr\\Assignments\\Assignment\\Simple Linear Regression\\delivery\_time.csv')

DT <- delivery$Delivery.Time

ST <- delivery$Sorting.Time

boxplot(DT, col="dodgerblue4")

boxplot(ST,col="dodgerblue4")

plot(ST,DT,main="Scatter Plot",

col="Dodgerblue4",

col.main="Dodgerblue4",

col.lab="Dodgerblue4",

xlab="Delivery Time",

ylab="Sorting Time", pch=20)

reg.model<-lm(DT~ST, data=delivery)

summary(reg.model)

plot(ST,DT,main="Line of Best Fit",col="Dodgerblue4", col.main="Dodgerblue4")

abline(reg.model, col="red")

res <- signif(residuals(reg.model))

pre <- predict(reg.model)

segments(ST, DT, ST, pre)

predict(reg.model,newdata= data.frame(ST=c(15,20)))

**Results :**

> reg.model<-lm(DT~ST, data=delivery)

> summary(reg.model)

Call:

lm(formula = DT ~ ST, data = delivery)

Residuals:

Min 1Q Median 3Q Max

-5.1729 -2.0298 -0.0298 0.8741 6.6722

Coefficients:

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

(Intercept) 6.5827 1.7217 3.823 0.00115 \*\*

ST 1.6490 0.2582 6.387 3.98e-06 \*\*\*

---

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

Residual standard error: 2.935 on 19 degrees of freedom

Multiple R-squared: 0.6823, Adjusted R-squared: 0.6655

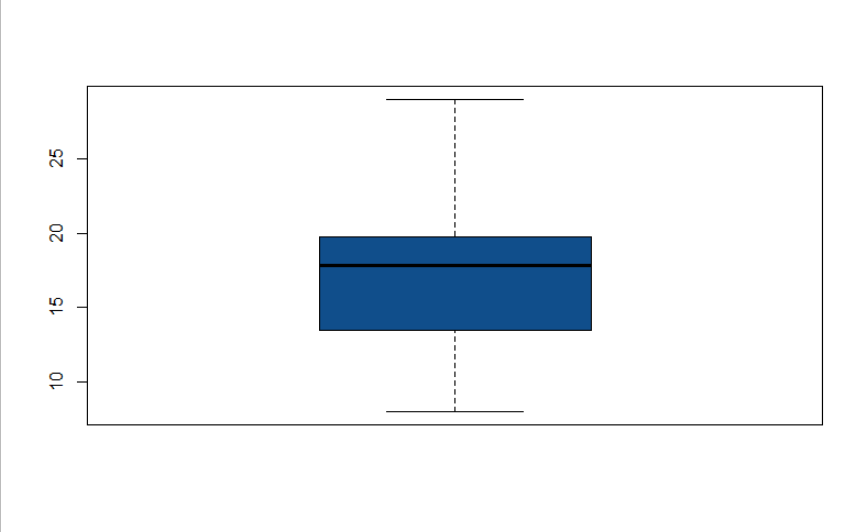
F-statistic: 40.8 on 1 and 19 DF, p-value: 3.983e-06

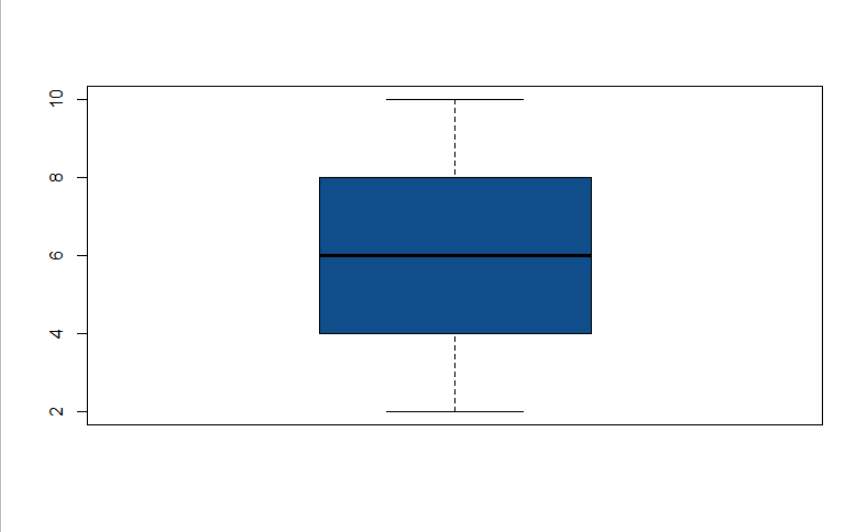
> predict(reg.model,newdata= data.frame(ST=c(15,20)))

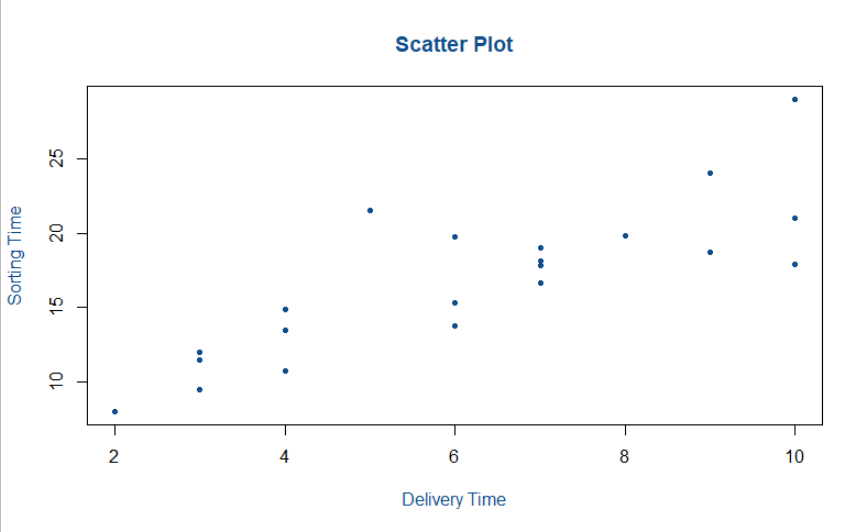
1 2

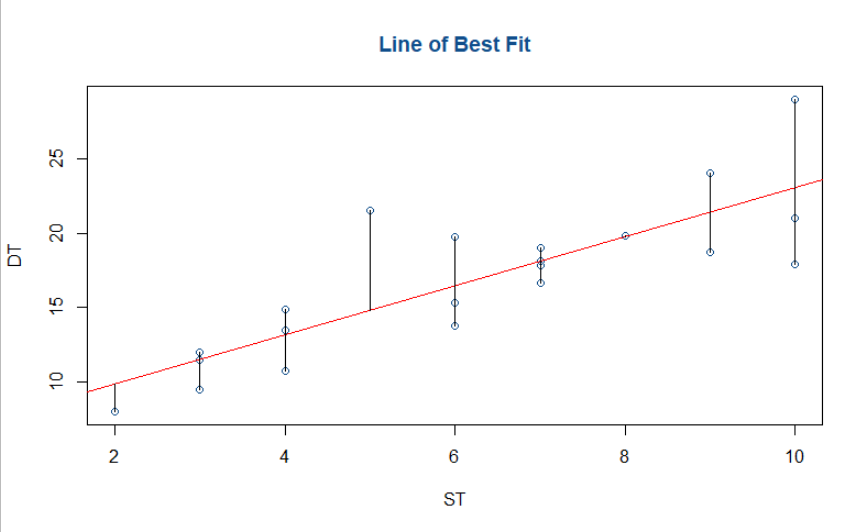
31.31803 39.56313

**Plots :**









**Inference :**

Getting R squared value of 0.6823.