>candy<-read.table("C:/Users/BFOUA/Data Science/candy.txt", header=TRUE)

> candy\_model<- lm(Candy ~., data = candy)

> step\_model <- stepAIC(candy\_model, direction = "both", trace = FALSE)

> install.packages("MASS")

package ‘MASS’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in

C:\Users\BFOUA\AppData\Local\Temp\RtmpGYikMe\downloaded\_packages

> library(MASS)

> step\_model <- stepAIC(candy\_model, direction = "both", trace = FALSE)

> summary(step\_model)

Call:

lm(formula = Candy ~ Costume + Decorations + Pumpkin, data = candy)

Residuals:

Min 1Q Median 3Q Max

-1540.52 -438.26 -37.37 502.13 1622.91

Coefficients:

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

(Intercept) 363.04612 420.31956 0.864 0.388699

Costume 0.35141 0.05026 6.992 3.41e-11 \*\*\*

Decorations 0.36881 0.10485 3.517 0.000532 \*\*\*

Pumpkin 0.36794 0.06403 5.747 3.10e-08 \*\*\*

---

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

Residual standard error: 624 on 214 degrees of freedom

Multiple R-squared: 0.3071, Adjusted R-squared: 0.2973

F-statistic: 31.61 on 3 and 214 DF, p-value: < 2.2e-16

> set.seed(19)

> R=runif(nrow(candy))

> candy$R=R

> candy$type<-ifelse(candy$R<=.6, "train", "test")

> train<-candy[candy$type=="train",]

> test<-candy[candy$type=="test",]

> regmodel<-lm(Candy~Costume+Decorations+Pumpkin,data=train)

> summary(regmodel)

Call:

lm(formula = Candy ~ Costume + Decorations + Pumpkin, data = train)

Residuals:

Min 1Q Median 3Q Max

-1338.71 -520.61 -88.33 496.25 1656.40

Coefficients:

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

(Intercept) -214.55788 561.66417 -0.382 0.703151

Costume 0.38089 0.06759 5.636 1.22e-07 \*\*\*

Decorations 0.53354 0.14680 3.635 0.000415 \*\*\*

Pumpkin 0.35609 0.07977 4.464 1.86e-05 \*\*\*

---

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

Residual standard error: 638 on 117 degrees of freedom

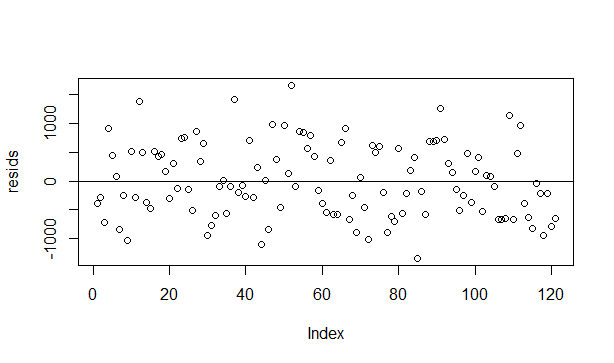
Multiple R-squared: 0.3673, Adjusted R-squared: 0.351

F-statistic: 22.64 on 3 and 117 DF, p-value: 1.256e-11

> resids<-resid(regmodel)

> plot(resids)

> abline(0,0)



> predicted <- predict.lm(regmodel,newdata=test)

> mape<-100\*mean(abs(test$Candy-predicted)/test$Candy)

> mape

[1] 13.01761

> newdata=data.frame(Costume=5000,Decorations=3000,Pumpkin=3300)

> predict(regmodel,newdata,interval="prediction",level=0.95)

fit lwr upr

1 4465.641 3189.146 5742.136

**Interpretation**

In this assignment we used R to determine the inputs (Costume, decorations, and Pumpkin) that are significantly correlated to the output (Candy). Then We predicted the output using those 3 inputs. The mean absolute percent error is 13.02 %. This mean absolute value suggests our model will be off the true value by 13.02 % on average. If we wanted to make a prediction for costume sales of 5000, decoration of 3000, and pumpkin of 3300 at the 95% confidence level, we will get a value between 3189.146 and 5742.136.