Simple Linear Regression

By Saba Akram

1/6/2021

cereal <- read.csv("C:/Users/sabaa/Downloads/cereals.CSV", stringsAsFactors=TRUE)  
# Save Rating and Sugar as new variables.  
sugars<-cereal$Sugars  
rating<-cereal$Rating  
which(is.na(sugars))

## [1] 58

sugars<-na.omit(sugars)  
length(sugars)

## [1] 76

rating<-rating[-58] # Deleting Record 58th from rating to match.

# Running Regression Analysis.  
lm1<-lm(rating~sugars)  
## Dsplaying Summaries  
summary(lm1)

##   
## Call:  
## lm(formula = rating ~ sugars)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -17.877 -5.612 -1.285 4.689 33.852   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 59.8530 1.9975 29.96 < 2e-16 \*\*\*  
## sugars -2.4614 0.2417 -10.18 1.01e-15 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 9.166 on 74 degrees of freedom  
## Multiple R-squared: 0.5835, Adjusted R-squared: 0.5779   
## F-statistic: 103.7 on 1 and 74 DF, p-value: 1.006e-15

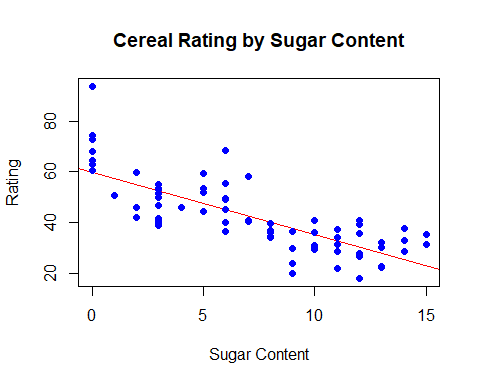
anova(lm1)

## Analysis of Variance Table  
##   
## Response: rating  
## Df Sum Sq Mean Sq F value Pr(>F)   
## sugars 1 8711.9 8711.9 103.69 1.006e-15 \*\*\*  
## Residuals 74 6217.4 84.0   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

lm1

##   
## Call:  
## lm(formula = rating ~ sugars)  
##   
## Coefficients:  
## (Intercept) sugars   
## 59.853 -2.461

# Plotting data with regression line   
plot(sugars,rating,main = "Cereal Rating by Sugar Content",xlab = "Sugar Content",ylab = "Rating",pch=16,col="blue",abline(lm1,col="red"))



#############  
   
##### Clearly from the above graph and the intercepts and the slope we came to know that as the content of the sugar increases the rating of the cereal decreases.  
   
   
############  
  
# Residuals, r^2, standardized residuals, leverage.  
lm1$residuals #### Residuals.

## 1 2 3 4 5 6   
## 23.3184774 -6.1779762 11.8795891 33.8518951 -5.7768122 -5.7292738   
## 7 8 9 10 11 12   
## 7.7809601 -3.1230932 4.0357574 5.7678971 -12.2731234 -6.6265977   
## 13 14 15 16 17 18   
## -17.8766620 -2.2228674 -5.1181082 -11.0237373 -9.0668525 5.4668166   
## 19 20 21 22 23 24   
## -5.4580412 -2.1743034 4.6807991 -5.5731123 0.9373812 -3.2150599   
## 25 26 27 28 29 30   
## 4.3530278 -1.3414216 15.7220656 5.6782322 10.6995176 -2.2902094   
## 31 32 33 34 35 36   
## 12.3207303 -13.8961920 4.5309811 0.9022507 -4.1956201 -10.9061026   
## 37 38 39 40 41 42   
## -4.1665978 -4.0349806 -8.5608126 -1.2287230 -13.2276423 0.2435784   
## 43 44 45 46 47 48   
## -3.5814594 2.3821607 4.3594684 1.3623704 2.4587968 -4.9785306   
## 49 50 51 52 53 54   
## -7.7759500 -1.9307554 4.7126605 -4.7879718 12.4474601 -10.9652163   
## 55 56 57 58 59 60   
## 0.9030951 3.1526281 4.4273784 8.9432226 -0.4582552 10.2486464   
## 61 62 63 64 65 66   
## -12.9312435 -11.9085973 8.3828681 14.6199321 12.9487701 8.2983403   
## 67 68 69 70 71 72   
## 0.6625677 11.8180771 -13.6290103 3.1996511 -5.8099123 -13.3625823   
## 73 74 75 76   
## -2.5626734 -2.6813113 -0.8765633 -3.9740962

lm1$residuals[12] ### Residual of cheeros, Record 12

## 12   
## -6.626598

a1<- anova(lm1)  
a1

## Analysis of Variance Table  
##   
## Response: rating  
## Df Sum Sq Mean Sq F value Pr(>F)   
## sugars 1 8711.9 8711.9 103.69 1.006e-15 \*\*\*  
## Residuals 74 6217.4 84.0   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

### Calculating r^2  
r\_sq<-a1$`Sum Sq`[1]/(a1$`Sum Sq`[1]+a1$`Sum Sq`[2])  
r\_sq

## [1] 0.5835462

std.res1<-rstandard(lm1) ### Standardized Error  
std.res1

## 1 2 3 4 5 6   
## 2.56183055 -0.67870387 1.30653068 3.78408802 -0.63463255 -0.63116942   
## 7 8 9 10 11 12   
## 0.86955016 -0.34309867 0.44337914 0.63435987 -1.35976207 -0.73724064   
## 13 14 15 16 17 18   
## -1.96594930 -0.24411959 -0.56928397 -1.21762296 -1.00472303 0.60567874   
## 19 20 21 22 23 24   
## -0.60709451 -0.23878620 0.52323676 -0.61557612 0.10326725 -0.35359593   
## 25 26 27 28 29 30   
## 0.48418457 -0.14814401 1.72662758 0.62554639 1.18541937 -0.25373654   
## 31 32 33 34 35 36   
## 1.38445683 -1.52820526 0.49832244 0.09965778 -0.46226516 -1.20444892   
## 37 38 39 40 41 42   
## -0.45901614 -0.44561547 -0.94051387 -0.13512630 -1.46105450 0.02676017   
## 43 44 45 46 47 48   
## -0.39679652 0.26312071 0.48145128 0.15045756 0.27349044 -0.54695475   
## 49 50 51 52 53 54   
## -0.85514418 -0.21203929 0.52222296 -0.52747023 1.39104826 -1.21115904   
## 55 56 57 58 59 60   
## 0.10095126 0.35241224 0.48640368 0.99083620 -0.05034328 1.12594381   
## 61 62 63 64 65 66   
## -1.43294689 -1.31535985 0.93706750 1.63426921 1.44746064 0.93246858   
## 67 68 69 70 71 72   
## 0.07318368 1.29976552 -1.50538746 0.35757247 -0.64173178 -1.47595925   
## 73 74 75 76   
## -0.28392333 -0.29616328 -0.09682048 -0.43658868

lev<- hatvalues(lm1) #### Leverage  
lev

## 1 2 3 4 5 6 7   
## 0.01389041 0.01381721 0.01601332 0.04749094 0.01381721 0.01930749 0.04697851   
## 8 9 10 11 12 13 14   
## 0.01381721 0.01389041 0.01601332 0.03036126 0.03841367 0.01586692 0.01315838   
## 15 16 17 18 19 20 21   
## 0.03797445 0.02443176 0.03072728 0.03036126 0.03797445 0.01315838 0.04749094   
## 22 23 24 25 26 27 28   
## 0.02443176 0.01930749 0.01601332 0.03797445 0.02413894 0.01315838 0.01930749   
## 29 30 31 32 33 34 35   
## 0.03036126 0.03036126 0.05737345 0.01586692 0.01601332 0.02443176 0.01952710   
## 36 37 38 39 40 41 42   
## 0.02413894 0.01930749 0.02413894 0.01389041 0.01586692 0.02443176 0.01389041   
## 43 44 45 46 47 48 49   
## 0.03036126 0.02443176 0.02413894 0.02413894 0.03797445 0.01389041 0.01586692   
## 50 51 52 53 54 55 56   
## 0.01315838 0.03072728 0.01930749 0.04697851 0.02443176 0.04749094 0.04749094   
## 57 58 59 60 61 62 63   
## 0.01389041 0.03036126 0.01381721 0.01389041 0.03072728 0.02443176 0.04749094   
## 64 65 66 67 68 69 70   
## 0.04749094 0.04749094 0.05737345 0.02443176 0.01601332 0.02443176 0.04697851   
## 71 72 73 74 75 76   
## 0.02443176 0.02443176 0.03036126 0.02443176 0.02443176 0.01381721

# Applying Regression in the orienteering Example.  
## Mentioning the data  
 x<-c(2,2,3,4,4,5,6,7,8,9)  
y<-c(10,11,12,13,14,15,20,18,22,25)  
o.data<-data.frame(cbind("Time"=x,"Distance"=y))  
lm2<-lm(Distance~Time,data = o.data) #### we can also use "lm2<-lm(y~x)"  
lm2

##   
## Call:  
## lm(formula = Distance ~ Time, data = o.data)  
##   
## Coefficients:  
## (Intercept) Time   
## 6 2

a2<-anova(lm2)  
a2

## Analysis of Variance Table  
##   
## Response: Distance  
## Df Sum Sq Mean Sq F value Pr(>F)   
## Time 1 216 216.0 144 2.144e-06 \*\*\*  
## Residuals 8 12 1.5   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

## Directly calculating r^2  
r\_sq2<-a2$`Sum Sq`[1]/(a2$`Sum Sq`[1]+a2$`Sum Sq`[2])  
r\_sq2

## [1] 0.9473684

### Calculating MSE  
mse<-a2$`Sum Sq`[2]/a2$Df[2]  
mse

## [1] 1.5

s<-sqrt(mse)  
s

## [1] 1.224745

###Standard deviation of Y  
sd(o.data$Distance)

## [1] 5.033223

lm2

##   
## Call:  
## lm(formula = Distance ~ Time, data = o.data)  
##   
## Coefficients:  
## (Intercept) Time   
## 6 2

r<-sign(lm2$coefficients[2])\*sqrt(r\_sq2) ###### Correlation Cofficients.  
r

## Time   
## 0.9733285

# Regression using hikers  
##Hard-core hiker.  
hardcore<-cbind("Time"=16,"Distance"=39)  
o.data<-rbind(o.data,hardcore)  
##### Creating a new table of 1x2 and then adding it in the o.data.   
##### Now we have 11 ovservations.  
lm3<-lm(Distance~Time,data=o.data)  
summary(lm3)

##   
## Call:  
## lm(formula = Distance ~ Time, data = o.data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.15188 -0.59091 0.09202 0.51275 1.90909   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 5.72506 0.65132 8.79 1.04e-05 \*\*\*  
## Time 2.06098 0.09128 22.58 3.11e-09 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.169 on 9 degrees of freedom  
## Multiple R-squared: 0.9827, Adjusted R-squared: 0.9807   
## F-statistic: 509.7 on 1 and 9 DF, p-value: 3.11e-09

anova(lm3)

## Analysis of Variance Table  
##   
## Response: Distance  
## Df Sum Sq Mean Sq F value Pr(>F)   
## Time 1 696.61 696.61 509.74 3.11e-09 \*\*\*  
## Residuals 9 12.30 1.37   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

hatvalues(lm3) ###### leverage of al the points.

## 1 2 3 4 5 6 7   
## 0.18847007 0.18847007 0.14578714 0.11529933 0.11529933 0.09700665 0.09090909   
## 8 9 10 11   
## 0.09700665 0.11529933 0.14578714 0.70066519

rstandard(lm3) ###### Standardized Residuals.

## 1 2 3 4 5 6   
## 0.14527829 1.09485088 0.08516653 -0.88122571 0.02823149 -0.92714432   
## 7 8 9 10 11   
## 1.71278715 -1.93712285 -0.19358734 0.67209730 0.46801423

######## here is the standard error of the residual which is equal to the difference of yi and y hat divided by the product of standard error of the estomate and sqrt of the (1-h) where h stands for the leverage point.  
## Cooks distance  
### It measures the influence of the variable in the regression. The variable is said to be influencial if its value is greater than 1. otherwise not.  
 ####It consists of two parts, one is the residual part and the other is the leverage part. The cooks distance is measured by the product of the residual part and the leverage part. ((yi-y^)/(m+1)s^2)\*[hi/(1-hi)^2]  
cooks.distance(lm3)

## 1 2 3 4 5 6   
## 2.450808e-03 1.391931e-01 6.189577e-04 5.060283e-02 5.193594e-05 4.617232e-02   
## 7 8 9 10 11   
## 1.466820e-01 2.015586e-01 2.442049e-03 3.854672e-02 2.563548e-01

### 5 hours, 20 km hiker  
##### Changing the 11th row.  
o.data[11,]<-cbind("Time"=5,"Distance"=20)  
lm4<-lm(Distance~Time,data = o.data)  
lm4

##   
## Call:  
## lm(formula = Distance ~ Time, data = o.data)  
##   
## Coefficients:  
## (Intercept) Time   
## 6.364 2.000

summary(lm4)

##   
## Call:  
## lm(formula = Distance ~ Time, data = o.data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.3636 -0.8636 -0.3636 0.6364 3.6364   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 6.3636 1.2781 4.979 0.000761 \*\*\*  
## Time 2.0000 0.2337 8.558 1.29e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.717 on 9 degrees of freedom  
## Multiple R-squared: 0.8906, Adjusted R-squared: 0.8784   
## F-statistic: 73.23 on 1 and 9 DF, p-value: 1.287e-05

anova(lm4)

## Analysis of Variance Table  
##   
## Response: Distance  
## Df Sum Sq Mean Sq F value Pr(>F)   
## Time 1 216.000 216.000 73.233 1.287e-05 \*\*\*  
## Residuals 9 26.545 2.949   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

rstandard(lm4)

## 1 2 3 4 5 6 7   
## -0.2457352 0.4300366 -0.2317106 -0.8413760 -0.2243669 -0.8327624 1.0096512   
## 8 9 10 11   
## -1.5061188 -0.2457352 0.4733409 2.2206996

hatvalues(lm4)

## 1 2 3 4 5 6 7   
## 0.25757576 0.25757576 0.16498316 0.10942761 0.10942761 0.09090909 0.10942761   
## 8 9 10 11   
## 0.16498316 0.25757576 0.38720539 0.09090909

cooks.distance(lm4)

## 1 2 3 4 5 6   
## 0.010475087 0.032079955 0.005304032 0.043491857 0.003092754 0.034674658   
## 7 8 9 10 11   
## 0.062628274 0.224095348 0.010475087 0.070785543 0.246575342

##### Here as we observed that the new value i.e. 11th row of 5 hours and 20 kms has the standardized residual of 2.2206996 which is greater than 2 hence by this, we know that the 11th position is the outlier position.  
#### Also its leverage value is 0.090909 which is lower than 2(m+1)/n or 3(m+1)/n=(0.36 or 0.545), hence it has low leverage value.  
#### Also the cooks distance of the 11th variable is 0.2465 which is less than 1, hence is is not influencial.  
#######The other measure of the influence is that if it lies in the 25th percentile of the F distribution the it is not influencial but if it lies beyond the 50th percentile of the F distribution then it is influencial  
   
  
o.data[11,]<-cbind("time"=10,"Distance"=23)  
lm5<-lm(Distance~Time,data = o.data)  
lm5

##   
## Call:  
## lm(formula = Distance ~ Time, data = o.data)  
##   
## Coefficients:  
## (Intercept) Time   
## 6.697 1.822

summary(lm5)

##   
## Call:  
## lm(formula = Distance ~ Time, data = o.data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.9194 -0.8969 -0.1635 0.6919 2.3697   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 6.6967 0.9718 6.891 7.14e-05 \*\*\*  
## Time 1.8223 0.1604 11.363 1.22e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.405 on 9 degrees of freedom  
## Multiple R-squared: 0.9348, Adjusted R-squared: 0.9276   
## F-statistic: 129.1 on 1 and 9 DF, p-value: 1.223e-06

anova(lm5)

## Analysis of Variance Table  
##   
## Response: Distance  
## Df Sum Sq Mean Sq F value Pr(>F)   
## Time 1 254.787 254.787 129.13 1.223e-06 \*\*\*  
## Residuals 9 17.758 1.973   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

hatvalues(lm5)

## 1 2 3 4 5 6 7   
## 0.24644550 0.24644550 0.16943128 0.11848341 0.11848341 0.09360190 0.09478673   
## 8 9 10 11   
## 0.12203791 0.17535545 0.25473934 0.36018957

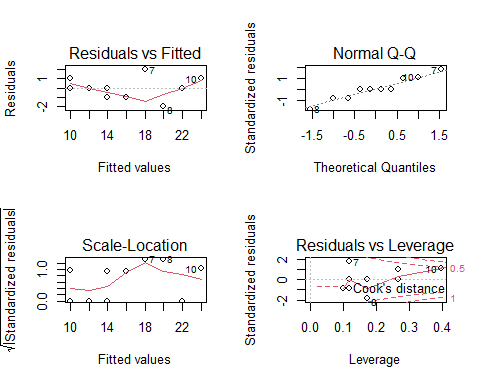
rstandard(lm5)

## 1 2 3 4 5 6   
## -0.27984229 0.54025108 -0.12772311 -0.74745710 0.01078063 -0.60423109   
## 7 8 9 10 11   
## 1.77309795 -1.10364850 0.56845516 1.56916868 -1.70831340

cooks.distance(lm5)

## 1 2 3 4 5 6   
## 1.280569e-02 4.772737e-02 1.663899e-03 3.754651e-02 7.810619e-06 1.885132e-02   
## 7 8 9 10 11   
## 1.646009e-01 8.465460e-02 3.435697e-02 4.208207e-01 8.214572e-01

##### Here we have leverage of the 11th observation (hatvalues) 0.3601 which is greater than 0.36 which we mentioned earlier. hence it haslow leverage value.  
#### we also have standardized residual of the 11th observation -1.708 which is less than 2 hence it not an outlier.  
#### We also have cooks distance of 0.82 which is less than 1 but it lies in the 62nd percentile of the F distribution. hence influencial.  
  
#### we also have standardized residual of the 11th observation -1.708 which is less than 2 hence it not an outlier.  
  
  
  
  
# Verifying the assumptions.  
par(mfrow=c(2,2))  
plot(lm2)



### Normal Probablity Plot : Top right   
### Residual vs fitted :top left  
### Square root of the abolute value of the standard variables: bottom left.  
### Resetting the plot space.  
  
# t-test  
summary(lm1)

##   
## Call:  
## lm(formula = rating ~ sugars)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -17.877 -5.612 -1.285 4.689 33.852   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 59.8530 1.9975 29.96 < 2e-16 \*\*\*  
## sugars -2.4614 0.2417 -10.18 1.01e-15 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 9.166 on 74 degrees of freedom  
## Multiple R-squared: 0.5835, Adjusted R-squared: 0.5779   
## F-statistic: 103.7 on 1 and 74 DF, p-value: 1.006e-15

### Here we get t value of -10.18 and p- value very close to zero which shows that there is no significant evidence in order to support the null hypothesis( sugar and ranking are not related i.e. b1=0), hence we reject the null hypothesis.  
### Hence we ranking and sugar are gighly related.  
  
  
# Confidence interval of th beta cofficient.  
confint(lm1,level = 0.95)

## 2.5 % 97.5 %  
## (Intercept) 55.872858 63.833176  
## sugars -2.943061 -1.979779

#Regression for carbohydrates and natural log of rating.  
carbs<-cereal$Carbo[-58]  
lrating<-log(rating)  
## Before testing the AD test we have to install the nortest package and then proceed. The low value of the AD statistics indicate that normal distribution fits for the data.  
library(nortest)  
ad.test(rating)

##   
## Anderson-Darling normality test  
##   
## data: rating  
## A = 0.96561, p-value = 0.01415

ad.test(lrating)

##   
## Anderson-Darling normality test  
##   
## data: lrating  
## A = 0.19108, p-value = 0.8945

ad.test(carbs)

##   
## Anderson-Darling normality test  
##   
## data: carbs  
## A = 0.66115, p-value = 0.08117

## Here after testing we observe that the p-value of rating is 0.01415, which is so low, hence we have solid evidence against null hypothesis that the distribution is normal.  
##Now after taking the log of the rating we observe that the p- value is 0.8945 which is so high indicating no evidence against the null hypothesis.  
## Similarly the p-value of the carbohydrate is 0.08117,which says that we have mild evidence against the null hypothesis that the distribution is normal.Hence it is barely acceptable.  
lm6<-lm(lrating~carbs)  
summary(lm6)

##   
## Call:  
## lm(formula = lrating ~ carbs)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.76915 -0.17770 -0.05079 0.21447 0.93079   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 3.504260 0.146539 23.913 <2e-16 \*\*\*  
## carbs 0.013137 0.009576 1.372 0.174   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.324 on 74 degrees of freedom  
## Multiple R-squared: 0.0248, Adjusted R-squared: 0.01162   
## F-statistic: 1.882 on 1 and 74 DF, p-value: 0.1742

a6<-anova(lm6)  
a6

## Analysis of Variance Table  
##   
## Response: lrating  
## Df Sum Sq Mean Sq F value Pr(>F)  
## carbs 1 0.1976 0.19761 1.8821 0.1742  
## Residuals 74 7.7697 0.10500

## After observing the data we find that the p-value of the slope during its estimation 0.174 wich is greater than 0.15from the p value table hence is has no evidence against the null hypothesis.  
##Also we see that the slope of the line is 0.013137 which means that the slope is close to zero indicating that lrating is independentof the carbohydrate.  
confint(lm6,level = 0.95)

## 2.5 % 97.5 %  
## (Intercept) 3.212274450 3.79624596  
## carbs -0.005943268 0.03221713

### Confidence Interval of r.  
alpha<-0.05  
n<-length(lrating)  
r\_sq6<-a6$`Sum Sq`[1]/(a6$`Sum Sq`[1]+a6$`Sum Sq`[2])  
r<-sign(lm6$coefficients[2])\*sqrt(r\_sq6)  
sr<-sqrt((1-r^2)/(n-2))  
lb<-r-qt(p=alpha/2,df=n-2,lower.tail = FALSE)\*sr  
ub<-r+qt(p=alpha/2,df=n-2,lower.tail = FALSE)\*sr  
lb

## carbs   
## -0.07124931

ub

## carbs   
## 0.3862266

#### Here by calclating the confidence interval of the r i.e. correlation cofficient we observe that the lower bound is -0.07124931 which is negative and the upper value is 0.3862266 which is psitive. It contans 0 in the confidence interval.  
#### If one end pon is nagative and the other end point is positive then we conclude with confidence level 100(1-alpha)% that x and y are not linearly correlated.

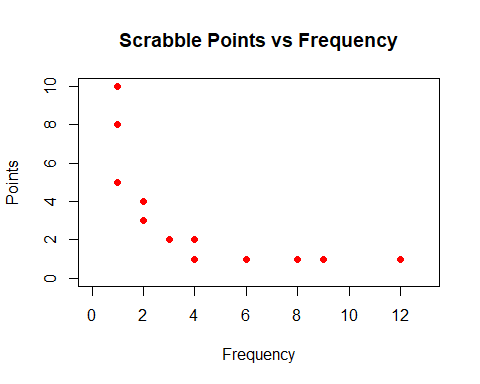
# Calculating the Confidence Interval and the Prediction Interval  
newdata<-data.frame(cbind(Distance=5,Time=5))  
conf.int<-predict(lm2,newdata,interval = "confidence")  
pred.int<-predict(lm2,newdata,interval = "prediction")  
conf.int

## fit lwr upr  
## 1 16 15.10689 16.89311

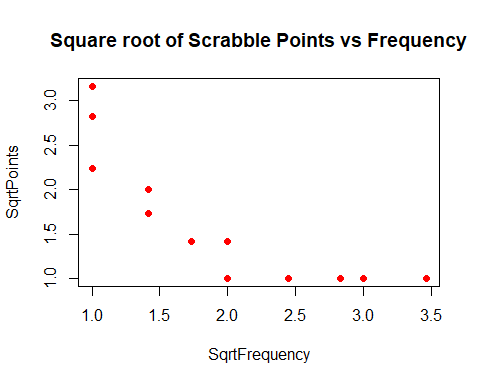
pred.int

## fit lwr upr  
## 1 16 13.03788 18.96212

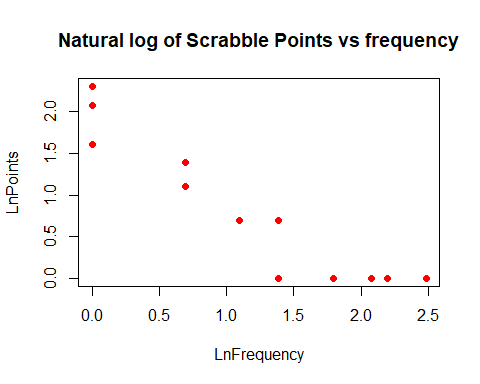
# Assessing Normality in Scrabble Example.  
## Scrabble Data  
s.freq<-c(9,2,2,4,12,2,3,2,9,1,1,4,2,6,8,2,1,6,4,6,4,2,2,1,2,1)  
s.point<-c(1,3,3,2,1,4,2,4,1,8,5,1,3,1,1,3,10,1,1,1,1,4,4,8,4,10)  
scrabble<-data.frame("Frequency"=s.freq,"Points"=s.point)  
plot(scrabble,main = "Scrabble Points vs Frequency",xlab = "Frequency",ylab = "Points",col="red",pch=16,xlim = c(0,13),ylim = c(0,10))



## The present position for all the untransfomed variable is t^1. Thus the bulging rule suggests that we apply either the square root transformation or the natural log transformation to both frequency and the points in order t achieve thelinear relationshp between the two variables.  
  
## The plot shows that it is not following a linear pttern. So in order to make it folw the inear pattern we will take the sqare root of the scrabble data.  
sq.scrabble<-sqrt(scrabble)  
plot(sq.scrabble,main = "Square root of Scrabble Points vs Frequency",xlab = "SqrtFrequency",ylab = "SqrtPoints",col="red",pch=16)



## Unfortunately, the graph indicates that the relationship between the sqrt frequency and the sqrt points is still not linear. So it will still be unappropriate to apply the linear regression. Hence we will apply the natural log transformation to each frequency and the point value.  
  
  
ln.scrabble<-log(scrabble)  
plot(ln.scrabble,main = "Natural log of Scrabble Points vs frequency",xlab = "LnFrequency",ylab = "LnPoints",col="red",pch=16)



## The above scatter plot exhibits the acceptable linearity, although, in the real world scatter plot, the lineaity is imperfect.Hence we may proceed with the regression analysis for lnpoints and lnfrequency.

# Run the regression on scrabble data, transformed and untransformed.  
lm7<-lm(Points~Frequency,data = ln.scrabble)  
summary(lm7)

##   
## Call:  
## lm(formula = Points ~ Frequency, data = ln.scrabble)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.5466 -0.1448 0.1391 0.1457 0.5579   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.94031 0.09916 19.57 2.94e-16 \*\*\*  
## Frequency -1.00537 0.07710 -13.04 2.20e-12 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2937 on 24 degrees of freedom  
## Multiple R-squared: 0.8763, Adjusted R-squared: 0.8712   
## F-statistic: 170 on 1 and 24 DF, p-value: 2.197e-12

anova(lm7)

## Analysis of Variance Table  
##   
## Response: Points  
## Df Sum Sq Mean Sq F value Pr(>F)   
## Frequency 1 14.6711 14.6711 170.03 2.197e-12 \*\*\*  
## Residuals 24 2.0709 0.0863   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

rstandard(lm7)

## 1 2 3 4 5 6 7   
## 0.9805705 -0.5050752 -0.5050752 0.5109923 2.0987808 0.4981830 -0.4952905   
## 8 9 10 11 12 13 14   
## 0.4981830 0.9805705 0.5031705 -1.1966450 -1.9054247 -0.5050752 -0.4921969   
## 15 16 17 18 19 20 21   
## 0.5429653 -0.5050752 1.3101916 -0.4921969 -1.9054247 -0.4921969 -1.9054247   
## 22 23 24 25 26   
## 0.4981830 0.4981830 0.5031705 0.4981830 1.3101916

## Here after observing the data of the lnfrequency and ln points, we get a very small P value which shows a strong evidence against the null hypothesis that frequency and Points are not linearly related.  
  
lm8<-lm(Points~Frequency,data = scrabble)  
summary(lm8)

##   
## Call:  
## lm(formula = Points ~ Frequency, data = scrabble)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.2001 -1.4661 -0.4661 0.8068 4.9008   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 5.7322 0.6743 8.502 1.06e-08 \*\*\*  
## Frequency -0.6330 0.1413 -4.480 0.000156 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.108 on 24 degrees of freedom  
## Multiple R-squared: 0.4554, Adjusted R-squared: 0.4327   
## F-statistic: 20.07 on 1 and 24 DF, p-value: 0.0001558

anova(lm8)

## Analysis of Variance Table  
##   
## Response: Points  
## Df Sum Sq Mean Sq F value Pr(>F)   
## Frequency 1 89.209 89.209 20.07 0.0001558 \*\*\*  
## Residuals 24 106.676 4.445   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

### Here after observing the raw data of the frequency and the points we get very low p value indicating a very strong evidence against the null hypothesis.

# Box\_Cox Transformation  
## Box cox Transformation requires MASS Package installation.  
library(MASS)  
bc<-boxcox(lm8)

