# DS 710 Homework 8

## 1

### a.

For me, I concentrated on five factors. I chose Bachelor's degrees, median household income, owner-occupied rate, and min/max temperature. I put a lot of weight on the Bachelor's degree factor. I also put slightly more weight on owner-occupied properties. In addition, for weather, I took a slightly unique approach. I enjoy all four seasons. This means a greater variance in max and min temperatures. As such, I "penalized" cities that did not have a great variance in their min and max temperatures.

### b.

cities <- read.csv('Best Cities.csv')  
pleasantness <- function(x)  
{  
 bachelor = as.numeric(x[1])   
 income = as.numeric(x[4])  
 housing = as.numeric(x[8])  
 maxTemp = as.numeric(x[17])  
 minTemp = as.numeric(x[18])  
 (bachelor^2 + income + housing\*2) / (maxTemp + minTemp)   
  
}  
#c.  
# I am skipping the first column intentionally  
apply(cities[,2:length(cities)], 2, pleasantness)

## Madison.city..Wisconsin Minneapolis.city..Minnesota   
## 752.7605 754.6651   
## San.Francisco.city..California Austin.city..Texas   
## 576.6394 445.3600   
## Philadelphia.city..Pennsylvania New.York.city..New.York   
## 382.5233 557.4339   
## Los.Angeles.city..California Seattle.city..Washington   
## 355.9113 591.9428   
## Portland.city..Oregon Miami.city..Florida   
## 463.4088 219.6752   
## Charlottesville.city..Virginia   
## 486.3094

### c.

Minneapolis is rated highest for me. This is where I currently live and I love it. Next highest is Madison. This is where I grew up and my wife and I often talk about going back there. I would say the model is accurate.

## 2

### a.

aMourner <- c(3, 7, 8, 3, 7, 3, 3, 6, 2, 3, 3, 2, 3, 4, 3, 8, 10, 2, 3, 3, 7, 4, 2, 10, 6, 3, 4, 9, 3, 6, 4, 2, 4, 2, 6, 4, 3, 7, 5, 2, 5, 4, 8, 11, 2, 6, 4, 4, 3, 3, 7, 2, 7, 3, 4, 2, 11, 2, 6, 5, 4, 8, 2, 3, 7, 2, 4, 6, 4, 3, 5, 6, 2, 3, 5, 10, 5, 6, 5, 4, 8, 8, 7, 2, 3, 8, 7, 2, 3, 6, 3, 6, 2, 3, 9, 3, 6, 4, 3, 3, 7, 3, 5, 2, 9, 3, 8, 8, 2, 6, 4, 3, 4, 5, 2, 3, 3, 4, 2, 7, 5, 6, 8, 4, 3, 7, 6, 6, 5, 2, 3, 6, 12, 6, 6, 2, 5, 5, 5, 6, 2, 5, 2, 3, 1, 7, 6, 3, 5, 4, 4, 1, 6, 3, 1, 7)  
  
t.test(aMourner, mu=4.69, alternative='two.sided')

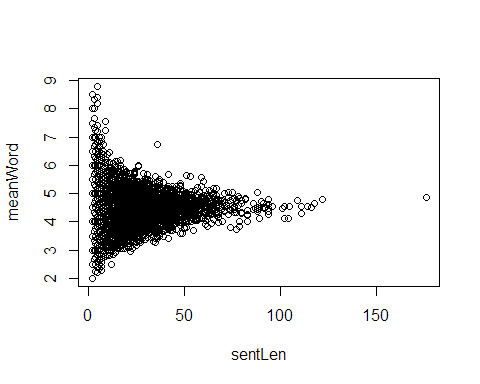
##   
## One Sample t-test  
##   
## data: aMourner  
## t = -0.195, df = 155, p-value = 0.8457  
## alternative hypothesis: true mean is not equal to 4.69  
## 95 percent confidence interval:  
## 4.287573 5.020119  
## sample estimates:  
## mean of x   
## 4.653846

### b.

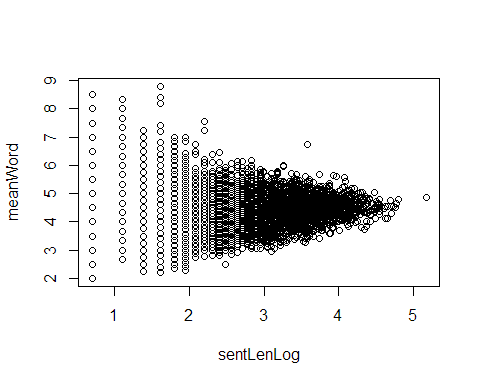
I have chosen to make the assumption that the average word length for all of John Hancock's writings was 4.69. With this assumption, I have failed to reject the null hypothesis. This means that it is *possible* that "A Mourner" has come from John Hancock.

## 3

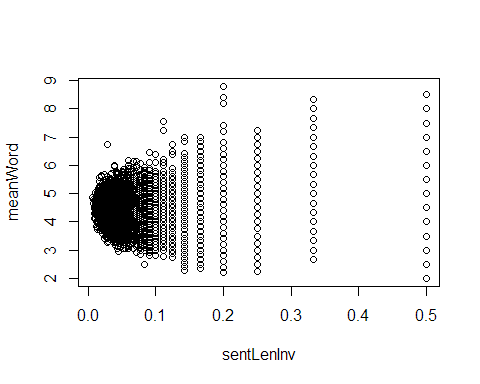
#a  
pp = read.csv('pride\_and\_prejudice.csv')  
sentLen = pp[[1]]  
meanWord = pp[[2]]  
plot(meanWord ~ sentLen)



#b  
sentLenLog = log(sentLen)  
plot(meanWord ~ sentLenLog)



##Trying inverse as another option  
sentLenInv = 1 / sentLen  
plot(meanWord ~ sentLenInv)



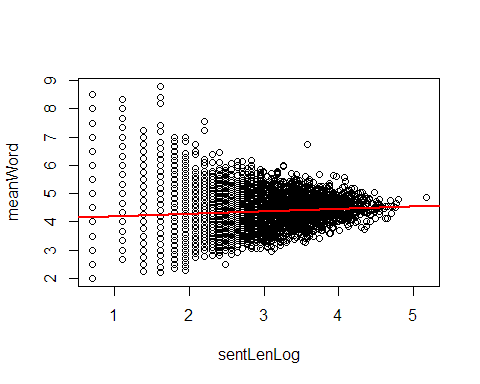
#c  
model = lm(meanWord ~ sentLenLog)  
model

##   
## Call:  
## lm(formula = meanWord ~ sentLenLog)  
##   
## Coefficients:  
## (Intercept) sentLenLog   
## 4.11513 0.08932

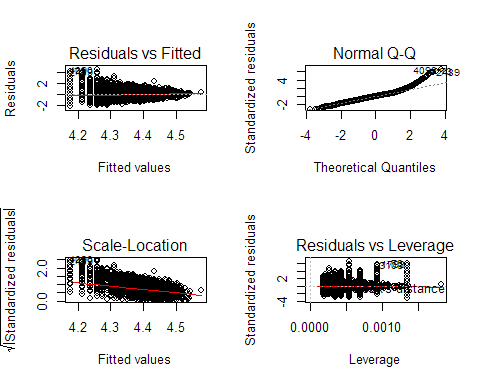
plot(meanWord ~ sentLenLog)  
summary(model)

##   
## Call:  
## lm(formula = meanWord ~ sentLenLog)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.1770 -0.4347 -0.0208 0.3587 4.5411   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 4.11513 0.03352 122.749 < 2e-16 \*\*\*  
## sentLenLog 0.08932 0.01173 7.612 3.13e-14 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.702 on 5756 degrees of freedom  
## Multiple R-squared: 0.009967, Adjusted R-squared: 0.009795   
## F-statistic: 57.95 on 1 and 5756 DF, p-value: 3.131e-14

#d  
abline(model, col='red', lwd=2)



#e  
par(mfrow = c(2,2))  
plot(model)



### a.

It does not appear that linear regression is appropriate here.

### c.

On average, for every extra word in the sentence there will be .005093 characters in the mean word length.

### d.

The slope is very nearly flat.

### e.

# TODO

The residual diagnostic plots shows...