Unit 2: Simple Linear Regression

Contents

[STOR 455 - Class 2 – Review of Inference 1](#_Toc88766268)

[STOR 455 Class 3 Linear Models and assessing conditions 1](#_Toc88766269)

[STOR 455 Class 4 assessing conditions and transformations 1](#_Toc88766270)

[STOR 455 Class 5 Transformations 1](#_Toc88766271)

[STOR 455 Class 6 Outliers and Points of Influence 1](#_Toc88766272)

[STOR 455 Class 7 Outliers and Points of Influence Again 1](#_Toc88766273)

[STOR 455 Class 8 Inference for Regression Slope 1](#_Toc88766274)

[STOR 455 Class 8 Inference for Regression Slope 1](#_Toc88766275)

[STOR 455 Class 9 Partitioning Variability - ANOVA 1](#_Toc88766276)

[STOR 455 Quiz 1 Solutions 1](#_Toc88766277)

[**Quiz A** 1](#_Toc88766278)

# Unit 2: Simple Linear Regression

## STOR 455 - Class 2 – Review of Inference

# loads packages needed  
# install a package before first using it for the first time  
  
library(readr)  
library(mosaic)  
  
# loads the DistanceHome dataframe into the environment from github  
  
DistanceHome <- read\_csv("https://raw.githubusercontent.com/JA-McLean/STOR455/master/data/DistanceHome.csv")  
  
# Alternative way to load dataframe (remove # to use)  
# DistanceHome.csv must be saved in the same folder as this notebook!  
  
#DistanceHome <- read\_csv("DistanceHome.csv")  
  
# Shows the variables and first 6 cases (by default)  
head(DistanceHome)

## # A tibble: 6 x 3  
## Distance Hours Introvert   
## <dbl> <dbl> <chr>   
## 1 7606 21 Introversion  
## 2 7606 21 Introversion  
## 3 3800 20 Introversion  
## 4 7102 20 Introversion  
## 5 6000 20 Introversion  
## 6 7756 18 Introversion

**Example: Distance to Home** - *Question:* How can we predict the distance from campus to home for Carolina students? - *Data:* Estimated distance to home (in miles) for students taking STOR 455 in a previous semester. - *Predictor variables:* Start with none.

**Example: Constant Model** Y = c + Error

Where c = an unknown constant

**Terminology** - “The constant c” = **parameter** of the model - “Sample estimate” - use data to provide a sample estimate of c

*How should we estimate 𝑐 from data?*

*Below:* Summarize the Distance variable - Numerical: mean and median

# dataframe$variable\_name  
  
mean(DistanceHome$Distance)

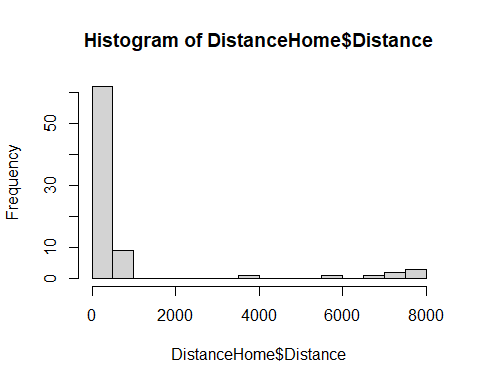
## [1] 844.6234

median(DistanceHome$Distance)

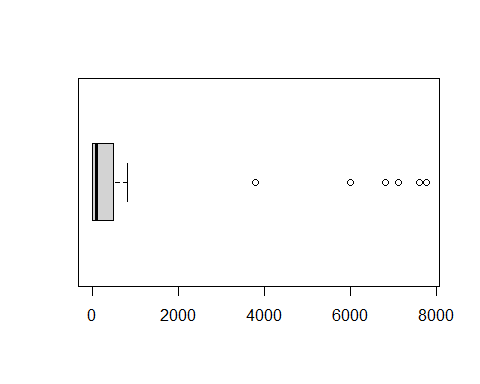
## [1] 113

*Below:* Summarize the Distance variable - Graphical: histogram, boxplot

hist(DistanceHome$Distance, breaks=20)



boxplot(DistanceHome$Distance, horizontal = TRUE)



**Predicted Value for Response** - Get an estimate for Y using the predictors and the model with estimated parameters.

*Notation:* The predicted y is denoted yhat

For the constant Model: yhat = chat

Examples: yhat = chat = ybar (\*sample mean) yhat = chat = m (sample median)

**Can we use a predictor to improve the model?**  X = Hours to travel home? X = Introvert? - Two sample t test for a difference in means

**Model with a Binary Predictor** Y = f(x) + Error where X = introversion, mu1 = mean distance for Extroverts mu2 = mean distance for Introverts

mean(Distance~Introvert, data=DistanceHome)

## Extroversion Introversion   
## 365.6026 1288.5939

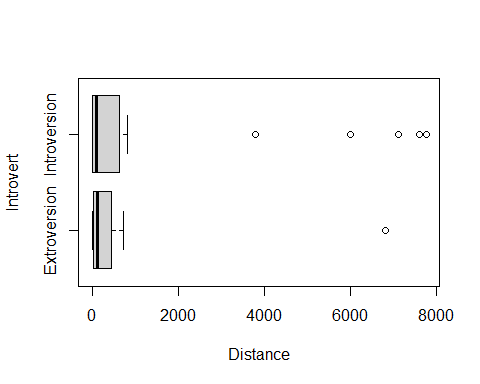
tally(~Introvert, data=DistanceHome)

## Introvert  
## Extroversion Introversion   
## 38 41

sd(Distance~Introvert, data=DistanceHome)

## Extroversion Introversion   
## 1094.443 2559.412

boxplot(Distance~Introvert, data=DistanceHome, horizontal=TRUE)



**Two-sample T-Test Difference in Means** *Hypothesis* Ho: Mu1 = Mu2 Ha: Mu1 != Mu2

*Compare to a t-dist*

**P-value** - The p-value is the proportion of samples, when the H0 is true, that would be as (or more) extreme as the observed sample.

*Below, Conclusion:* Decision: Reject H0 only when the p-value is small.

t.test(Distance~Introvert, data=DistanceHome)

##   
## Welch Two Sample t-test  
##   
## data: Distance by Introvert  
## t = -2.1103, df = 55.025, p-value = 0.03939  
## alternative hypothesis: true difference in means between group Extroversion and group Introversion is not equal to 0  
## 95 percent confidence interval:  
## -1799.48957 -46.49298  
## sample estimates:  
## mean in group Extroversion mean in group Introversion   
## 365.6026 1288.5939

**Normality?** - The two-sample t-test assumes both samples are from normal populations

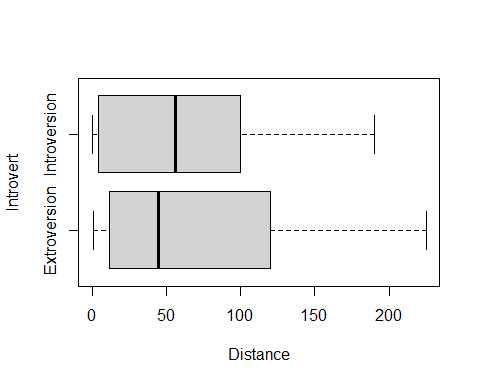
**Domestic Distance** - Suppose that we look only at distances that are really “drivable”? *See below*

Domestic = subset(DistanceHome, Distance<250)  
head(Domestic)

## # A tibble: 6 x 3  
## Distance Hours Introvert   
## <dbl> <dbl> <chr>   
## 1 225 3.5 Extroversion  
## 2 120 3 Extroversion  
## 3 190 3 Introversion  
## 4 172 3 Introversion  
## 5 167. 3 Extroversion  
## 6 190 3 Introversion

* For distance home (only including students less than 250 miles from home) the Introvert variable does not improve the model significantly.

boxplot(Distance~Introvert, data=Domestic, horizontal=TRUE)



t.test(Distance~Introvert, data=Domestic)

##   
## Welch Two Sample t-test  
##   
## data: Distance by Introvert  
## t = 0.09629, df = 51.968, p-value = 0.9237  
## alternative hypothesis: true difference in means between group Extroversion and group Introversion is not equal to 0  
## 95 percent confidence interval:  
## -33.17687 36.52132  
## sample estimates:  
## mean in group Extroversion mean in group Introversion   
## 67.16667 65.49444

**Inference Review: Hypothesis Testing** - Suppose that we look only at distances that are really “drivable”? *Test* Ho: mu1 = Mu2 Ha: mu1 != Mu2

There is a 93.3% chance that we would receive a samples with a difference as extreme as we did if the null hypothesis is true. p-value = 0.933

Since the p-value is greater than 0.05, we fail to reject the null hypothesis. There is not evidence to suggest that there is a difference in the number of miles from home Carolina students are (of those students 250 miles or less) based on if they are introverts or extroverts.

There is a 93.3% chance that we would receive a samples with a difference as extreme as we did if the null hypothesis is true. p-value = 0.933

Domestic=subset(DistanceHome,Distance<250)

**Domestic Distance** For distance home (only including students less than 250 miles from home) the Introvert variable does not improve the model significantly.

t.test(Distance~Introvert, data=Domestic)

##   
## Welch Two Sample t-test  
##   
## data: Distance by Introvert  
## t = 0.09629, df = 51.968, p-value = 0.9237  
## alternative hypothesis: true difference in means between group Extroversion and group Introversion is not equal to 0  
## 95 percent confidence interval:  
## -33.17687 36.52132  
## sample estimates:  
## mean in group Extroversion mean in group Introversion   
## 67.16667 65.49444

## STOR 455 Class 3 Linear Models and assessing conditions

# message=FALSE, warning=FALSE supress warnings and messages from appearing in knitted html  
  
library(readr)  
  
DistanceHome <- read\_csv("https://raw.githubusercontent.com/JA-McLean/STOR455/master/data/DistanceHome.csv")  
  
# If notebook and csv file are saved in the same folder  
# DistanceHome <- read\_csv("DistanceHome.csv")  
  
Domestic=subset(DistanceHome,Distance<250)

**Single Quantitative Predictor Model** - Notation:  
– Y = Response variable – X = Predictor variable

*Assume (for now) that both Y and X are quantitative variables.* Y = f(x) + Error

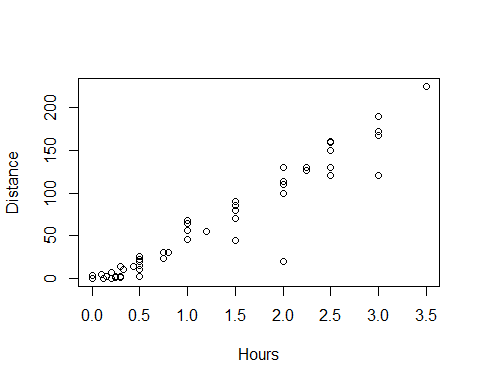
*Simple Linear Model* - X = Single quantitative predictor - Y = Quantitative response

*Goal*: Find a line that best summarizes the trend in the data.

Y = Bo + B1x + Error Response = Intercept + Slope*Predictor + Random Error* Assumptions:\* - Assume: Error ~ Follows a normal distribution and independent - There are 3 parameters to estimate: Bo, B1, and std error

**Scatterplot in R** *See below*

plot(Distance~Hours, data=Domestic)



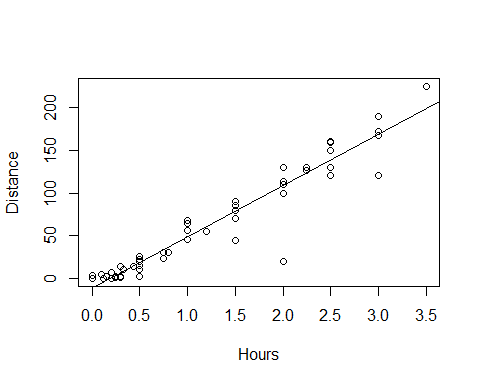
**Least Square Regression in R** *Syntax:* Syntax: lm(Response~Predictor,data= )

lm(Distance~Hours, data=Domestic)

##   
## Call:  
## lm(formula = Distance ~ Hours, data = Domestic)  
##   
## Coefficients:  
## (Intercept) Hours   
## -11.06 59.98

*R – Plot with Fitted Line*

mod1=lm(Distance~Hours, data=Domestic)  
plot(Distance~Hours, data=Domestic)  
abline(mod1)



*Simple Linear Model- Conditions* **Model:** 1. Linearity: The means for Y vary as a linear function of X. **Error:** 2.Zero Mean: The distribution of the errors is centered at zero. 3.Constant variance: The variance for Y is the same at each X. (Homoscedasticity) 4.Independence: No relationships among errors. 5.Normality: - Residuals are normally distributed - (sometimes) At each X, the Y’s follow a normal distribution.

*Linear* Look for consistent curvature or non-linear patterns

*Constant Variance* Look for “fan-shaped” pattern - Fan-shaped is **bad**

summary(mod1)

##   
## Call:  
## lm(formula = Distance ~ Hours, data = Domestic)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -88.892 -4.680 2.172 7.082 26.141   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -11.063 4.056 -2.727 0.00868 \*\*   
## Hours 59.977 2.484 24.144 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 18.26 on 52 degrees of freedom  
## Multiple R-squared: 0.9181, Adjusted R-squared: 0.9165   
## F-statistic: 582.9 on 1 and 52 DF, p-value: < 2.2e-16

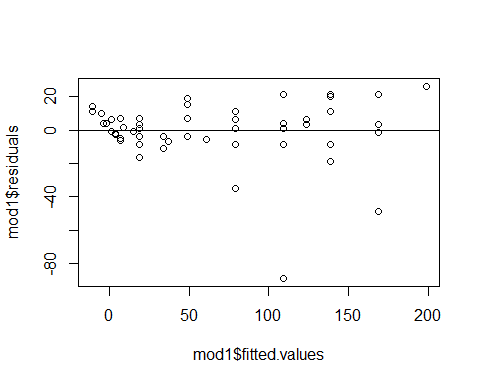
mod1$residuals

## 1 2 3 4 5 6   
## 26.1414099 -48.8698461 21.1301539 3.1301539 -1.7698461 21.1301539   
## 7 8 9 10 11 12   
## -8.8811021 20.1188979 -18.8811021 11.1188979 21.1188979 6.1132699   
## 13 14 15 16 17 18   
## 3.1132699 -8.8923581 21.1076419 -88.8923581 4.1076419 1.1076419   
## 19 20 21 22 23 24   
## 6.0963859 6.0963859 -34.9036141 -8.9036141 6.0963859 11.0963859   
## 25 26 27 28 29 30   
## 1.0963859 -5.9103678 -3.9148702 19.0851298 15.0851298 7.0851298   
## 31 32 33 34 35 36   
## -6.9193726 -3.9204982 -10.9204982 -16.4261262 3.0738738 7.0738738   
## 37 38 39 40 41 42   
## -3.9261262 -8.9261262 1.0738738 -0.7277020 1.2700468 -4.9306286   
## 43 44 45 46 47 48   
## 7.0693714 -6.4306286 -2.9317542 -1.9317542 -1.9317542 -0.7328798   
## 49 50 51 52 53 54   
## 6.4671202 4.0659946 4.0153193 10.0648690 14.0626178 11.0626178

mod1$fitted.values

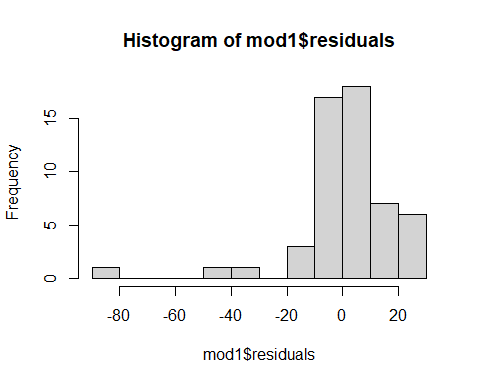
## 1 2 3 4 5 6   
## 198.8585901 168.8698461 168.8698461 168.8698461 168.8698461 168.8698461   
## 7 8 9 10 11 12   
## 138.8811021 138.8811021 138.8811021 138.8811021 138.8811021 123.8867301   
## 13 14 15 16 17 18   
## 123.8867301 108.8923581 108.8923581 108.8923581 108.8923581 108.8923581   
## 19 20 21 22 23 24   
## 78.9036141 78.9036141 78.9036141 78.9036141 78.9036141 78.9036141   
## 25 26 27 28 29 30   
## 78.9036141 60.9103678 48.9148702 48.9148702 48.9148702 48.9148702   
## 31 32 33 34 35 36   
## 36.9193726 33.9204982 33.9204982 18.9261262 18.9261262 18.9261262   
## 37 38 39 40 41 42   
## 18.9261262 18.9261262 18.9261262 14.7277020 8.7299532 6.9306286   
## 43 44 45 46 47 48   
## 6.9306286 6.9306286 3.9317542 3.9317542 3.9317542 0.9328798   
## 49 50 51 52 53 54   
## 0.9328798 -2.0659946 -3.8653193 -5.0648690 -11.0626178 -11.0626178

plot(mod1$residuals~mod1$fitted.values)  
abline(0,0)



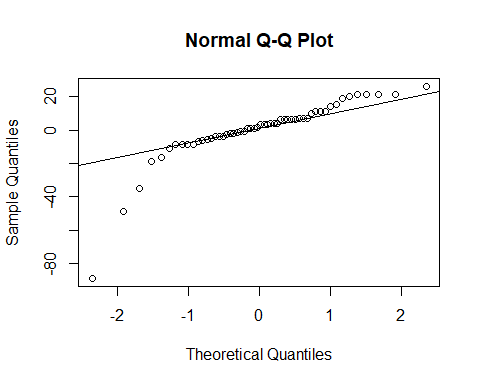
*Residuals* Look at a histogram of the residuals Look for clear skewness and outliers - skew and outliers are **bad**

hist(mod1$residuals, breaks=10)

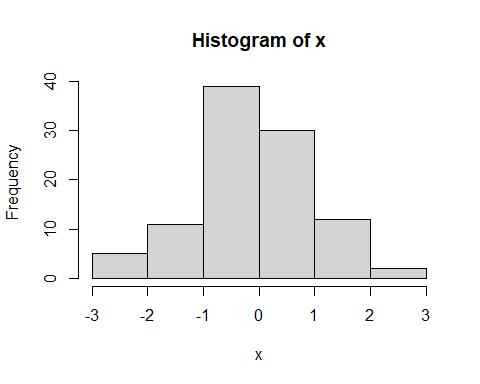


*How much Variability is Expected?*

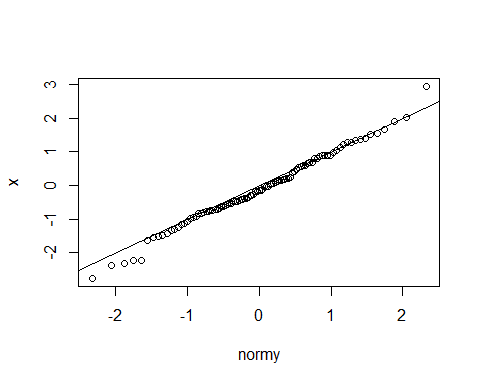
qqnorm(mod1$residuals)  
qqline(mod1$residuals)



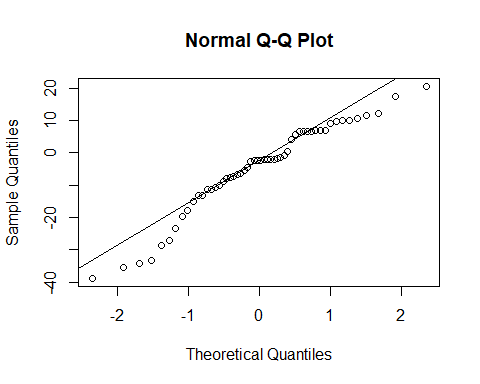
#Sets randomization  
set.seed(455001)  
  
#Sample of 99 values from a Normal distribution; mean=0; sd=1; sorted ascending  
x = sort(rnorm(99,0,1))  
hist(x)



#list of integers 1 through 99.  
y = c(1:99)  
  
#z-scores of dataset of 99 values if perfectly normally distributed  
normy = qnorm(y/100)  
  
plot(x~normy)  
abline(0,1)



x <- rnorm(54, 0, 18.26)  
qqnorm(x)  
qqline(x)



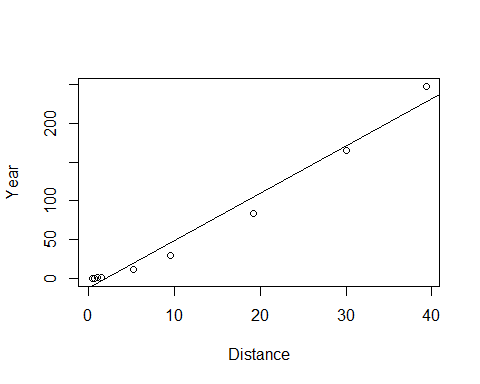
## STOR 455 Class 4 assessing conditions and transformations

# message=FALSE, warning=FALSE suppress warnings and messages from appearing in knitted html  
  
library(readr)  
library(Stat2Data)  
  
Planets <- read\_csv("https://raw.githubusercontent.com/JA-McLean/STOR455/master/data//Planets.csv")

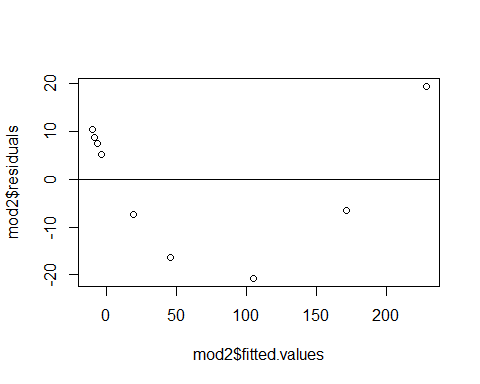
head(Planets, 9)

## # A tibble: 9 x 7  
## Planet Distance Year Mass Day Diameter Gravity  
## <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 Mercury 0.39 0.24 0.055 1408. 3.04 0.37  
## 2 Venus 0.72 0.61 0.815 5832. 7.52 0.88  
## 3 Earth 1 1 1 24 7.92 1   
## 4 Mars 1.52 1.88 0.107 24.6 4.22 0.17  
## 5 Jupiter 5.2 11.9 318. 9.9 88.8 2.64  
## 6 Saturn 9.52 29.5 95.2 10.2 74.6 1.15  
## 7 Uranus 19.2 84.1 14.5 17.2 31.6 1.15  
## 8 Neptune 30.0 165. 17.2 16.1 30.2 1.12  
## 9 Pluto 39.3 248. 0.003 153. 1.86 0.04

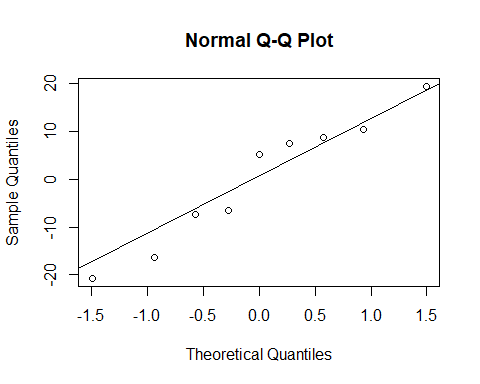
plot(Year~Distance, data=Planets)  
mod2=lm(Year~Distance, data=Planets)  
abline(mod2)

 *Simple Linear Model- Conditions* **Model:** 1. Linearity: The means for Y vary as a linear function of X. **Error:** 2.Zero Mean: The distribution of the errors is centered at zero. 3.Constant variance: The variance for Y is the same at each X. (Homoscedasticity) 4.Independence: No relationships among errors. 5.Normality: - Residuals are normally distributed - (sometimes) At each X, the Y’s follow a normal distribution.

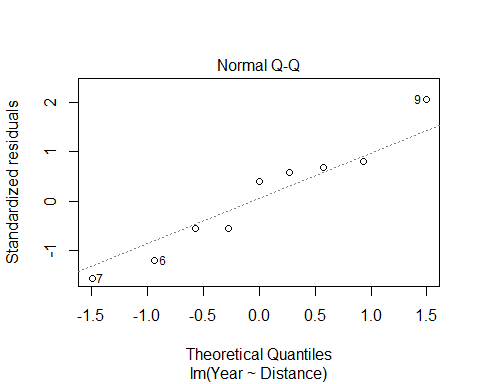
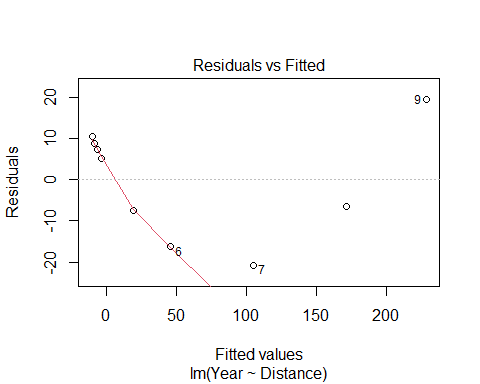
plot(mod2$residuals~mod2$fitted.values)  
abline(0,0)



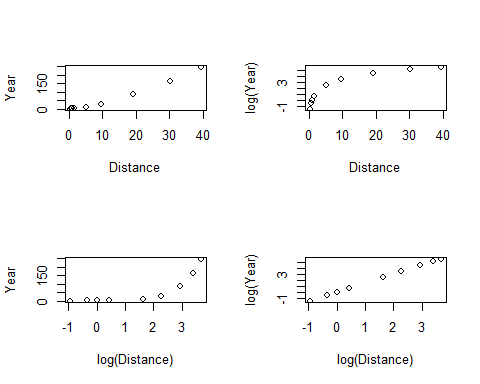
qqnorm(mod2$residuals)  
qqline(mod2$residuals)



plot(mod2, 1:2)

 **What to do when regression assumptions are violated?** *Examples:* 1.Nonlinear patterns in residuals 2.Heteroscedasticity (nonconstant variance) 3.Lack of normality in residuals 4.Outliers: influential points, large residuals

par(mfrow=c(2,2))  
  
plot(Year~Distance, data=Planets)  
plot(log(Year)~Distance, data=Planets)  
plot(Year~log(Distance), data=Planets)  
plot(log(Year)~log(Distance), data=Planets)

 **Data Transformations** *Can be used to:* 1.Address non-linear patterns 2.Stabilize variance 3.Remove skewness from residuals 4.Minimize effects of outliers

**Common Transformations** For either the response (Y) or predictor (X)… - Logarithm: 𝑌→l𝑜𝑔⁡(𝑌) - Square root: 𝑌→√𝑌 - Exponentiation: 𝑌→𝑒^Y - Power function: 𝑌→𝑌^3 - Reciprocal: 𝑌→1/𝑌

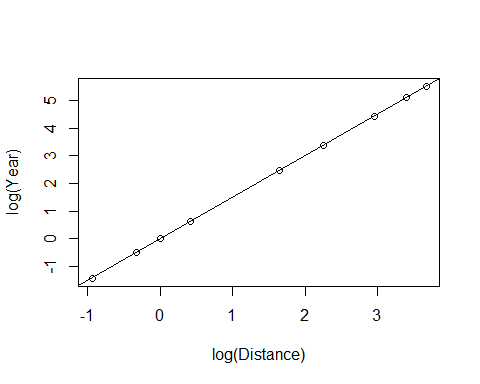
**Example: Planets**

Y = Length of the “year” for planets X = Distance from the sun

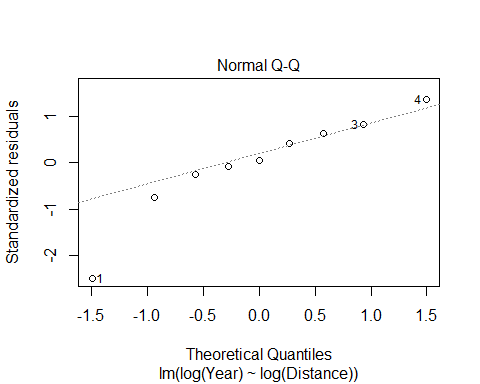
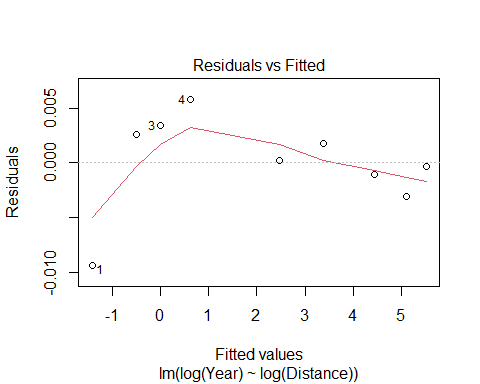
Try scatterplots and LM with Year vs. Distance log(Year) vs. Distance Year vs. log(Distance) log(Year) vs. log(Distance)

*Which transformation gives the best linearity?*

mod3 = lm(log(Year)~log(Distance), data=Planets)  
  
plot(log(Year)~log(Distance), data=Planets)  
abline(mod3)



plot(mod3, 1:2)



summary(mod3)

##   
## Call:  
## lm(formula = log(Year) ~ log(Distance), data = Planets)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.0093289 -0.0010233 0.0002193 0.0025708 0.0057772   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -0.0034339 0.0020852 -1.647 0.144   
## log(Distance) 1.5020611 0.0009567 1570.016 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.004662 on 7 degrees of freedom  
## Multiple R-squared: 1, Adjusted R-squared: 1   
## F-statistic: 2.465e+06 on 1 and 7 DF, p-value: < 2.2e-16

log(Year) = -0.0034399 + 1.5020611\*log(Distance)

Year = e ^(-0.0034399 + 1.5020611(log(Distance))

Year = e ^(-0.0034399) e ^((1.5020611)(log(Distance))

Year = e ^(-0.0034399) e ^(log(Distance ^1.5020611))

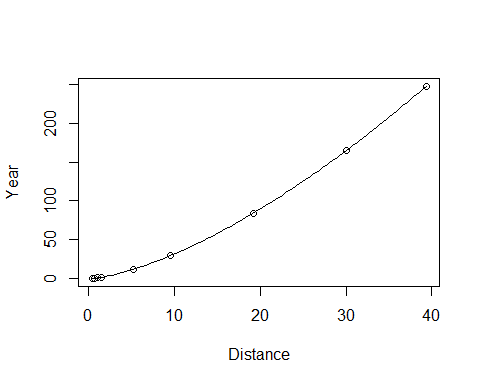
Year = e ^(-0.0034399) (Distance^1.5020611)

exp(-0.0034339)

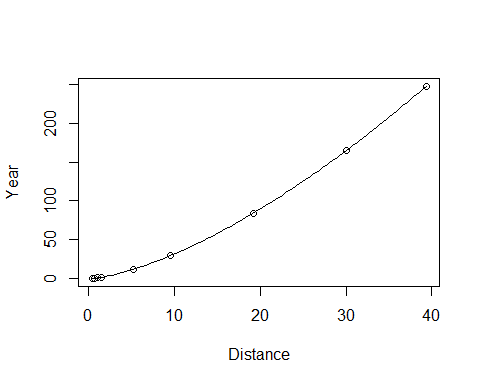
## [1] 0.996572

Year = 0.996572(Distance^1.5020611)

plot(Year~Distance, data=Planets)  
curve(0.996572\*(x^1.5020611), add=TRUE)



B0 = summary(mod3)$coefficients[1,1]  
B1 = summary(mod3)$coefficients[2,1]  
  
plot(Year~Distance, data=Planets)  
curve(exp(B0)\*x^B1, add=TRUE)



## STOR 455 Class 5 Transformations

# message=FALSE, warning=FALSE suppress warnings and messages from appearing in knitted files  
  
library(readr)  
library(Stat2Data)

**What to do when regression assumptions are violated?** *Examples of violations:* 1. Nonlinear patterns in residuals 2. Heteroscedasticity (nonconstant variance) 3. Lack of normality in residuals 4. Outliers: influential points, large residuals

**Data Transfomrations** Can be used to: 1. Address non-linear patterns 2. Stabilize variance 3. Remove skewness from residuals 4. Minimize effects of outliers

**Common Transformations** - Log - Square root - Exponentiation - Power function - Reciprocal

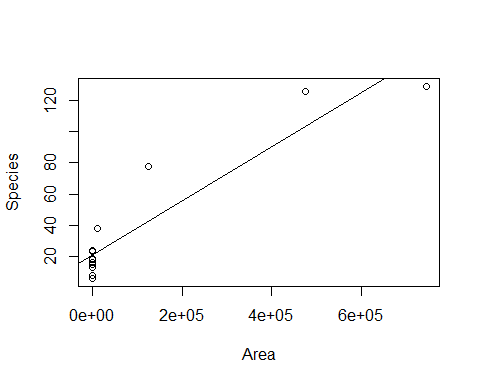
*Example: Mammal Species* Y = Number of mammal species on an island X = Area of the island

Data on fourteen islands in Southeast Asia are stored in SpeciesArea (in Stats2Data)

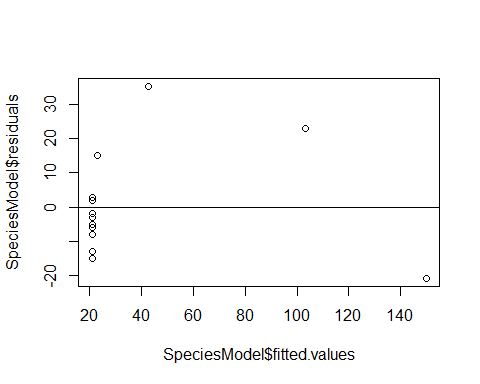
*Notes* - Make sure you check your working environment

**Why is log of something useful?** - Log goes up and then trails off forever - If have a big right skew, or extreme values or outliers in the right side, it helps squeeze the outliers back in - If we have constant variance issues, with fanning patterns, use log - DOesn’t effect low values very much, but helps smoosh bigger data

# Pull in the data   
data("SpeciesArea")  
  
# The log of the species are going to be the most useful   
# Plot the data, make a linear model   
# Want to predict the number of species on each island, based on teh island area  
plot(Species~Area, data=SpeciesArea) # Just the scatterplot, see that is appears to follow a log, but we'll keep make the other models to jsut see how bad it violates the other conditions of linear  
  
SpeciesModel=lm(Species~Area, data=SpeciesArea)  
abline(SpeciesModel) # See that the line deosn't work for us



# Doing Residual analysis   
plot(SpeciesModel$residuals~SpeciesModel$fitted.values)  
abline(0,0)



# When looking at plots, you can see that it's pretty bad   
# Residual analysis can also be done with plot(mod1, 1:2)

# Tells you which point is biggests value of residual   
max(SpeciesModel$residuals)

## [1] 35.24152

# Tells you where the value is in the table   
which.max(SpeciesModel$residuals)

## 3   
## 3

# Gives you the row of the max, this is the number that you got from which.max above  
SpeciesArea[3,]

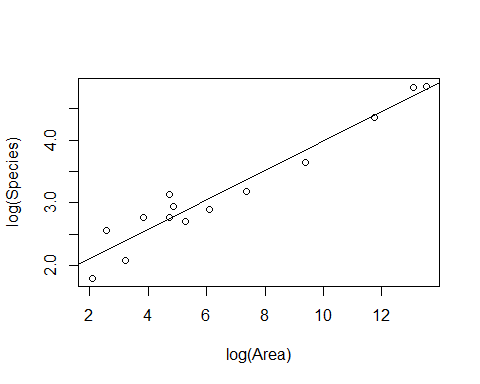
## Name Area Species logArea logSpecies  
## 3 Java 125628 78 11.7411 4.35671

# Just another way to call SpeciesArea[3,]  
SpeciesArea[SpeciesArea$Name=="Java",]

## Name Area Species logArea logSpecies  
## 3 Java 125628 78 11.7411 4.35671

**New Transformation Model**

# New transformation model   
plot(log(Species)~log(Area), data=SpeciesArea)  
SpeciesModel2=lm(log(Species)~log(Area), data=SpeciesArea)  
abline(SpeciesModel2) # Plotting the linear regression line on the scatterplot of the data

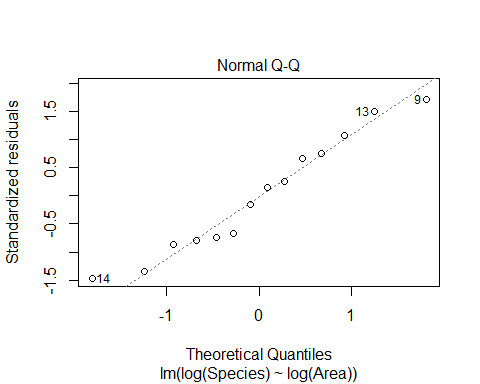
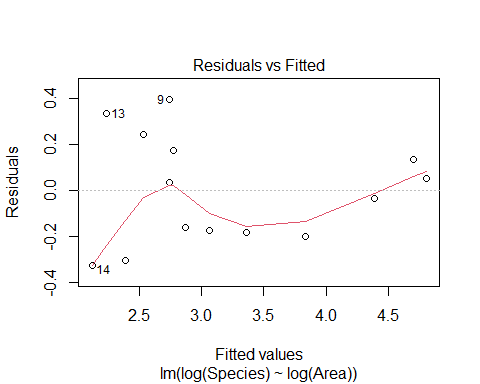


**HOw to interpret the summary table of a transformed linear model** - *Interpret:* For every 1 unit increase in the log(area), there is a 0.2355 increase in log(species)

summary(SpeciesModel2) # Gives you the output of the linear model

##   
## Call:  
## lm(formula = log(Species) ~ log(Area), data = SpeciesArea)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.32280 -0.18071 0.00079 0.16356 0.39534   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.6249 0.1326 12.26 3.81e-08 \*\*\*  
## log(Area) 0.2355 0.0175 13.46 1.34e-08 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2427 on 12 degrees of freedom  
## Multiple R-squared: 0.9379, Adjusted R-squared: 0.9327   
## F-statistic: 181.1 on 1 and 12 DF, p-value: 1.335e-08

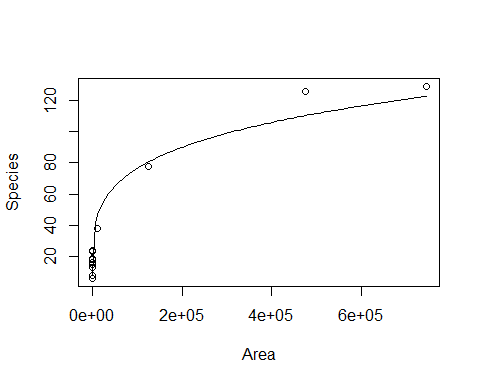
# Look at the coefficients table: Intercept and slope for the Log(area)   
# Intercept (B0) for Log(area) = 1.6249, intercept of the reg line   
# Slope (B1): Log(Area): 0.2335  
# For every 1 unit increase in the log(area), there is a 0.2355 increase in log(species)  
  
# Checking the conditions of the transformed linear model   
plot(SpeciesModel2, 1:2)



**Pulling out the coeffecients of the linear model** - Also shows how to plot the linear model - You need to solve for the same variables before you plot the linear model curve on the base plot because otherwise, it’s in different variables

*BELOW: HOW TO SOLVE OUT FOR LOG OF BOTH SIDES* **IMPORTARNT**

B0 = summary(SpeciesModel2)$coefficients[1,1] # Intercept  
B1 = summary(SpeciesModel2)$coefficients[2,1] # Slope  
  
plot(Species~Area, data=SpeciesArea)  
curve(exp(B0)\*x^B1, add=TRUE) # This is the linear model curve, on the normal data, but solved so that they are in the same units.



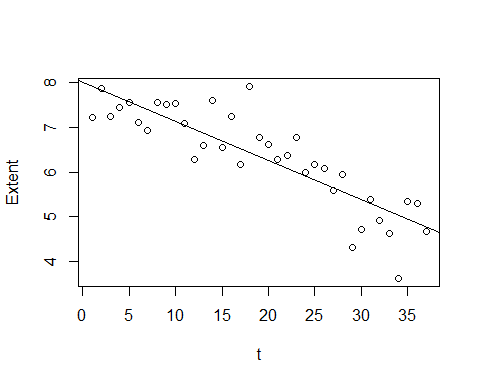
**Artic Sea Ice** The SeaIce data gives information about the amount of sea in the arctic region as measured in September (the time when the amount of ice is at its least) since 1979. The basic research question is to see if we can use time to model the amount of sea ice.

In fact, there are two ways to measure the amount of sea ice: Area and Extent. Area measures the actual amount of space taken up by ice. Extent measures the area inside the outer boundaries created by the ice. If there are areas inside the outer boundaries that are not ice (think about a slice of swiss cheese), then the Extent will be a larger number than the Area. In fact, this is almost always true.

data("SeaIce")  
head(SeaIce)

## Year Extent Area t  
## 1 1979 7.22 4.54 1  
## 2 1980 7.86 4.83 2  
## 3 1981 7.25 4.38 3  
## 4 1982 7.45 4.38 4  
## 5 1983 7.54 4.64 5  
## 6 1984 7.11 4.04 6

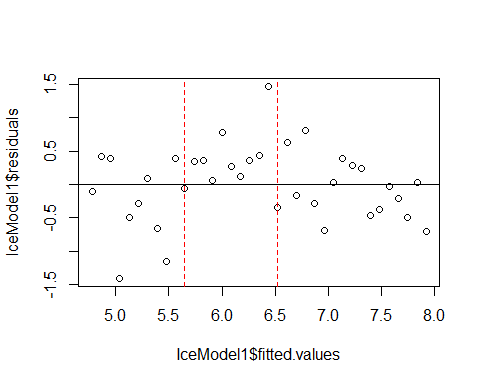
plot(Extent~t, data = SeaIce) # Basic plot of the oringial data   
IceModel1=lm(Extent~t, data = SeaIce) # Linear model of extent by time   
abline(IceModel1) # Plotting the line on the plot



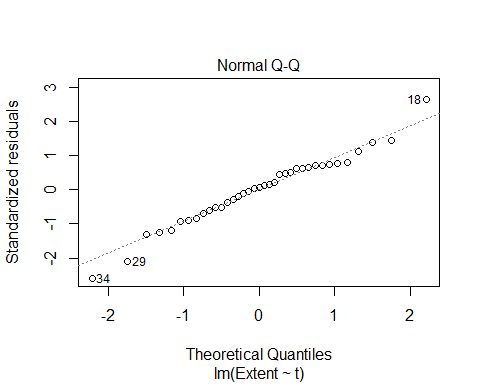
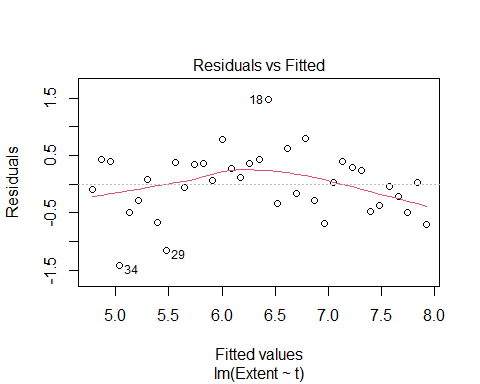
# We see that th eline look spretty good on the data

**Looking at Residuals** - The residual plot has a slight curve - The red line = the benefit of using the plotted of the model itself, rather than the residuals by fitted separate; shows that there is some pattern and curve there - We can see this really well at the middle region - There is some region bt 5 and 6.5 where all the residuals are above the line - So your prediction in that range will always be below what it should be

plot(IceModel1$residuals~IceModel1$fitted.values)  
abline(0,0) #Could also write plot(mod1, 1:2)  
  
# Below used the line to draw two vertical lines where the residual plot was looking weird  
# Will draw 2 vertical lines, one at X of 5.65 and the other 6.52  
abline(v=c(5.65,6.52),   
 col=c("red", "red"),   
 lty=c(2,2), # Look like a dash line   
 lwd=c(1, 1)) #Draws red dashed vertical lines; width

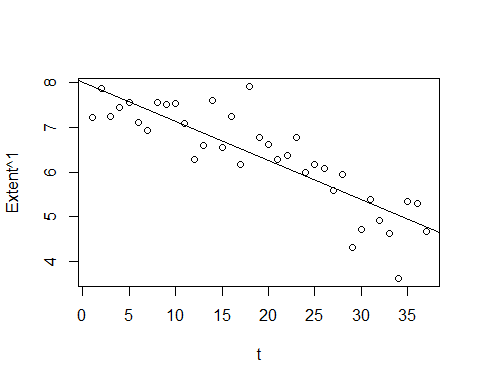


# The abline above shows you where the plot is under predicting   
  
plot(IceModel1, 1:2)

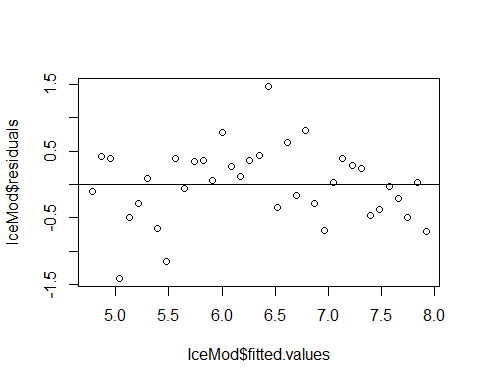


**How does doing transformations change the output?** - Trying exponential to 1

plot(Extent^1~t, data = SeaIce) # Basic plot of the oringial data   
IceMod=lm(Extent^1~t, data = SeaIce) # Linear model of extent by time   
abline(IceMod)



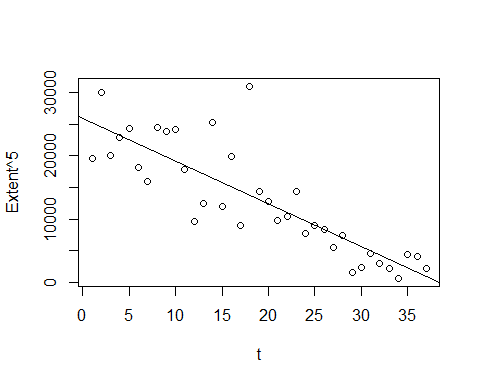
plot(IceMod$residuals~IceMod$fitted.values)  
abline(0,0) #Could also write plot(mod1, 1:2)



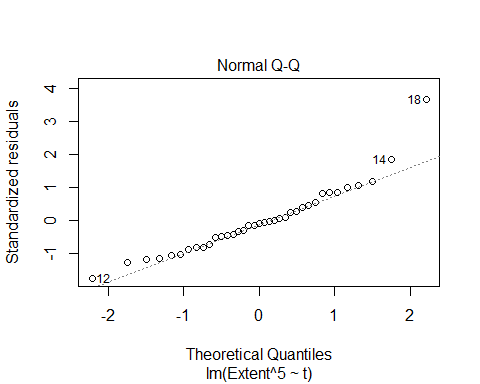
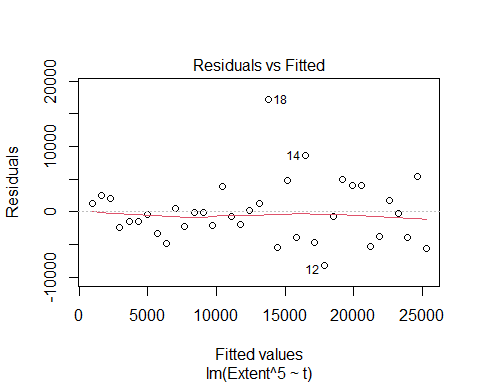
**Trying to raise to the 5th power** - Notice: This looks a bit better on the residual plot - Subjective - No very defined curvature - Look at teh oringal data, we see that there are not some region anymore where all the dots are above or below the line - This might have helped

* We may have also made other problems. See that one middle point that is well above the line? The transformation might have made things worse in different ways

plot(Extent^5~t, data = SeaIce) # Basic plot of the oringial data   
IceMod=lm(Extent^5~t, data = SeaIce) # Linear model of extent by time   
abline(IceMod)



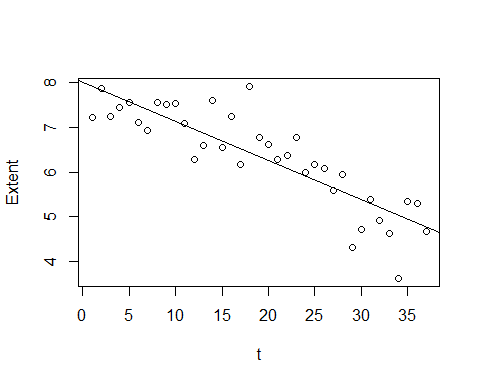
plot(IceMod, 1:2)



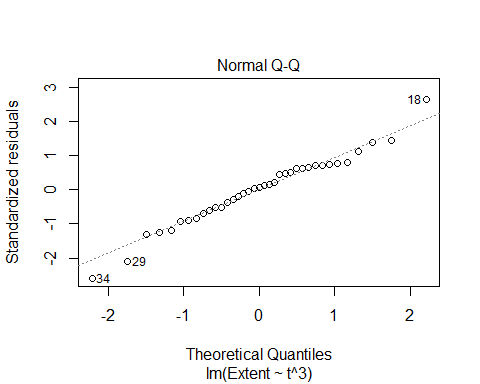
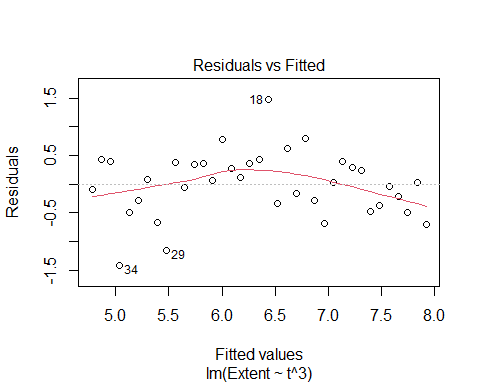
# The reisudal by fitted in teh orinigal plot had a red curve;but in this one the line has appeared to be subdued, but it may be decieving ebcause it plots the range of the data   
# it might just be the extreme case fo 18 appear to stretch it out and tone down what it actualyl looks like when I show it   
# Not as defined as it was before   
  
# The normal QQplot looks pretty good, there the 2 on teh tail that are flying out, but for the small dataset, things appear to fit well   
  
# No constant variance issues here   
  
# Might not be a good model, but it's a better model

**What if we take off the 5th power? and raise time tot he power instead?** - THis gets the curve back in the residual plot, and we don’t want that. so this makes it worse/same - R is ignoring you when you trying to raise the predictor to a power. It thinks that you’re not trying to do that

# THIS IS THE WRONG WAY  
plot(Extent~t^3, data = SeaIce) # Basic plot of the oringial data   
IceMod=lm(Extent~t^3, data = SeaIce) # Linear model of extent by time   
abline(IceMod)



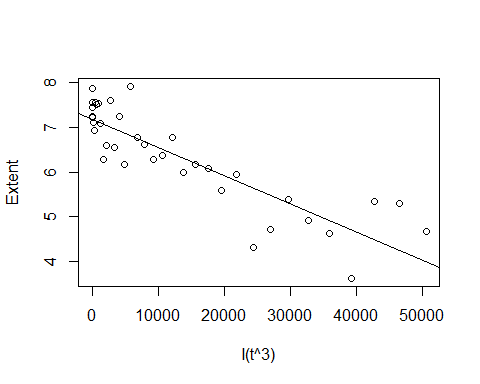
plot(IceMod, 1:2)



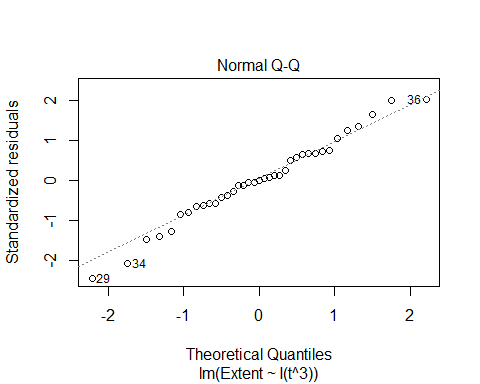
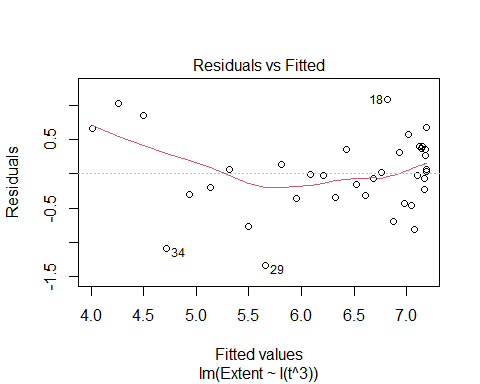
**How to power the predictor**  
-Somethings that work outside of functions may work differently inside of functions. - The carrot that should be raising it, worked on the response, but it won’t work on the predictor, inside of lm - It thinks you’re trying to do an interaction between variables, and you’re not trying to do that.

**Below: HOW TO Power PREDICTOR PROPERLY** - Still doens’t make a good model in this instance, but it could help in the future

# THIS IS THE RIGHT WAY  
plot(Extent~I(t^3), data = SeaIce) # Basic plot of the oringial data   
IceMod=lm(Extent~I(t^3), data = SeaIce) # Linear model of extent by time   
abline(IceMod)

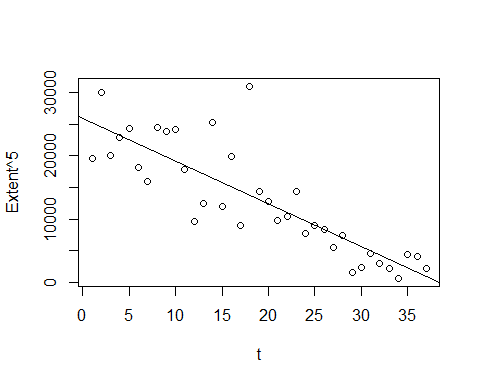


plot(IceMod, 1:2)

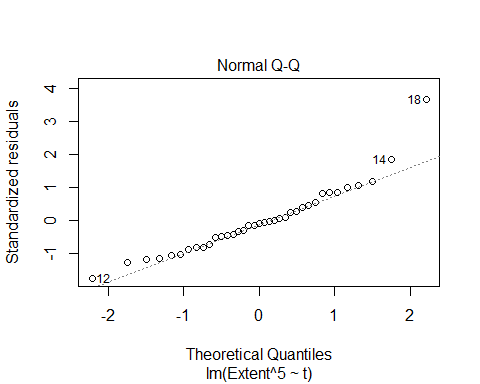
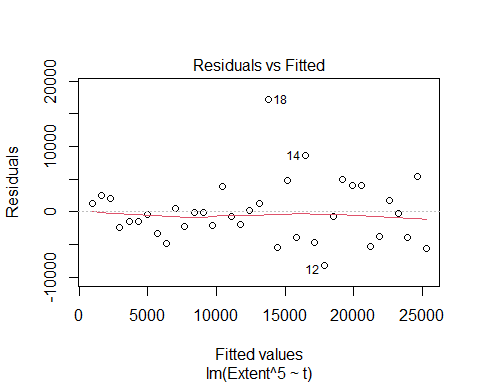


**USING THE BEST MODEL** - WE are using the 5th power model - What if we want to plot this on the orignial raw data?

plot(Extent^5~t, data = SeaIce) # Basic plot of the oringial data   
IceMod=lm(Extent^5~t, data = SeaIce) # Linear model of extent by time   
abline(IceMod)



plot(IceMod, 1:2)

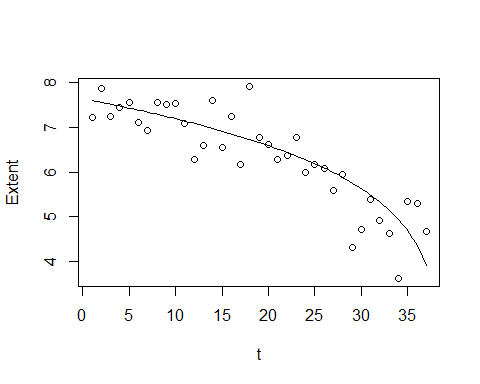


**PLOTTING TRANSFORMATION ON RAW DATA**

plot(Extent~t, data = SeaIce)  
  
summary(IceMod)

##   
## Call:  
## lm(formula = Extent^5 ~ t, data = SeaIce)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -8157.2 -3296.7 -438.1 2102.9 17179.4   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 25969.69 1597.27 16.259 < 2e-16 \*\*\*  
## t -676.85 73.29 -9.235 6.51e-11 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 4760 on 35 degrees of freedom  
## Multiple R-squared: 0.709, Adjusted R-squared: 0.7007   
## F-statistic: 85.29 on 1 and 35 DF, p-value: 6.512e-11

# Pull out coeff of the transform model   
#INtercept  
B0\_Ice = summary(IceMod)$coefficients[1,1] # Intercept  
   
#Slope  
B1\_Ice = summary(IceMod)$coefficients[2,1] # Slope   
  
# Solve for the curve with math   
curve((B0\_Ice+B1\_Ice\*x)^(1/5), add = TRUE) # Tke 5th root of each side, jsut solved



## STOR 455 Class 6 Outliers and Points of Influence

# message=FALSE, warning=FALSE suppress warnings and messages from appearing in knitted files  
  
library(readr)  
library(Stat2Data)

**Single Quantitative Predictor Model** Notation:  
- Y = Response variable - X = Predictor variable

Assume (for now) that both Y and X are quantitative variables.

**Simple Linear Model** X = Single quantitative predictor Y = Quantitative response

Find a line that best summarizes the trend in the data.

Y = Bo + B1X + E Response = intercept + Slope\*Predictor + Random Error

**Simple Linear Model- Conditions** **Model:** 1.Linearity: The means for Y vary as a linear function of X. **Error:** 2.Zero Mean: The distribution of the errors is centered at zero. 3.Constant variance: The variance for Y is the same at each X. (Homoscedasticity) 4.Independence: No relationships among errors. 5.Normality: Residuals are normally distributed (sometimes) At each X, the Y’s follow a normal distribution.

*Look at* What potent do these points have to influence our model?

**Types of “Unusual” Points in SLM** - Two Types - **Outlier:** A data point that is far from the regression line. –Points really above or below the regression line – Doesn’t always have a lot of influence on the model – Could be big enough that it has influence, but mostly depends on the value of the predictor – Data points that are closer to the edges of the predictor value (high or low) have a higher chance of having inlfuence in our model - **Influential point:** A data point that has a large effect on the regression fit. – can come from many things

**Detecting Unusual Cases - Overview** 1. Compute residuals “raw”, standardized, studentized 2. Plots of residuals (or std. residuals) Boxplot, scatterplot, normal plot 3. Leverage Unusual values for the predictors 4. Cook’s distance Cases with large influence

*This notebook covers the first two from above* - leverage = the potiential for a certain value to have infleuence on the model - Cook’s distance combines a lot of the things to do some calculations for us

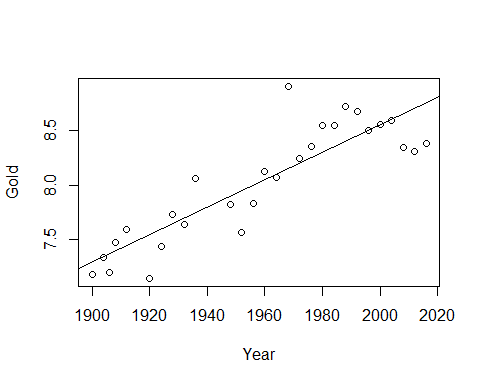
**Raw Residual** ei = yi - yhati *How can we tell if a residual is unusually large?*  CONTEXT! Example: Y = GPA ei = 2.6 is very large Y = SAT ei = 2.6 is very small

**Example: Men’s Olympic Long Jump**

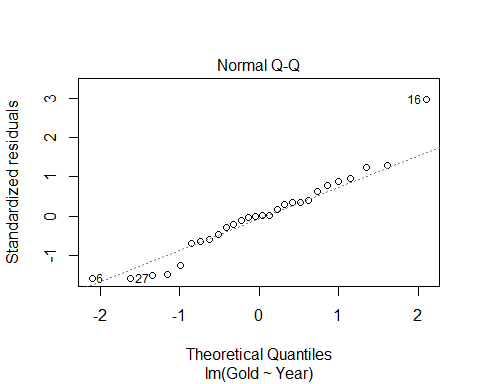
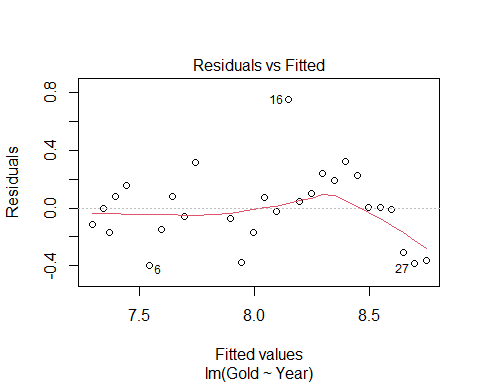
data("LongJumpOlympics2016")  
head(LongJumpOlympics2016)

## Year Gold  
## 1 1900 7.185  
## 2 1904 7.340  
## 3 1906 7.200  
## 4 1908 7.480  
## 5 1912 7.600  
## 6 1920 7.150

plot(Gold~Year, data=LongJumpOlympics2016) # Predict longjump distance by year   
GoldModel = lm(Gold~Year, data=LongJumpOlympics2016)  
abline(GoldModel) # Draw the line we made onto the plot



plot(GoldModel, 1:2) # To see the residual models, we see that there is one really big outlier



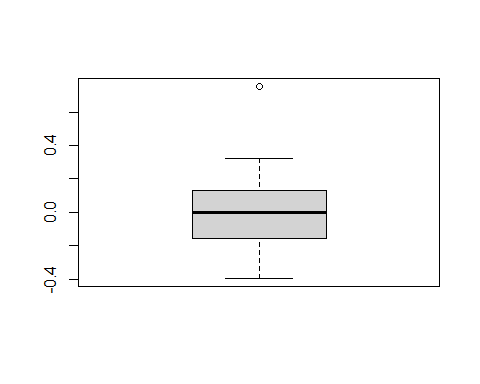
# Linearity looks like an issue because of the fitted plot. The red line goes down; the prediction curves   
# Point 16 is our outlier   
# R tells us which row this is because it looks like an ourlier to R   
# Looking at the normal QQ Plot, normal might be an issue   
# Constant variance doesn't appear to be an issue   
summary(GoldModel)

##   
## Call:  
## lm(formula = Gold ~ Year, data = LongJumpOlympics2016)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.39610 -0.15495 -0.00137 0.11606 0.75349   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -16.470194 2.666282 -6.177 1.56e-06 \*\*\*  
## Year 0.012508 0.001361 9.191 1.19e-09 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2595 on 26 degrees of freedom  
## Multiple R-squared: 0.7646, Adjusted R-squared: 0.7556   
## F-statistic: 84.47 on 1 and 26 DF, p-value: 1.192e-09

# Look at estimate of the year: For every 1 year increase than that's how many meters we think the winning long distance jump is going to increase as well   
# Every 4 years it looks like its increasing by 5 cms

**What if we wanted to see what the data looked like without the outlier?**

boxplot(GoldModel$residuals)



# Outliers are more than 1.5 IQR’s beyond the Quartiles  
# Will give idea of how different that one value is from the others   
  
# Wee see there is an outlier, but how much of an outlier is this?   
# LOOK AT STANDARDIZED VALUES OF RESIDUALS INSTEAD  
  
max(GoldModel$residuals)

## [1] 0.7534932

which.max(GoldModel$residuals)

## 16   
## 16

**Standardized Residuals** - HOW TO TELL HOW MUCH OF AN OUTLIER THIS IS - ROughly equal to the acutal - predicted/stdeve; basically a zscore, but not exactly

* Fact: If X has mean mu and std. dev, then (𝑋−𝜇)/𝜎 has mean 0 and std. dev.=1.
* For residuals: mean=0 and std. dev. of errors
* Standardized Residuals about equals (yi - yhat)/stad dev of the population errors
* Look for values beyond +/-2 (mild) or beyond +/-3
* Once you have fit mymodel=lm(Y~X)
* Use: rstandard(mymodel)

*notes* - It will give us a thing centered at zero +/- unites; and that’s how many std they are away from teh average - THink about this as: - Once you are +/- 2 std away and its normally dist, then you’re in teh outer 5% of the data, so that’s starting to eb an outlier - IF you’re +/-3 away, then you’re into the .05 of the data and outliers, so it’s pretty extreme

*Below* - If we look at the standaized residual, it’s 2.96. so it’s 2.96 std above the line, which is an outlier

rstandard(GoldModel) # Put the model in

## 1 2 3 4 5 6   
## -0.45846037 -0.02447194 -0.69927805 0.34245659 0.62395356 -1.58872766   
## 7 8 9 10 11 12   
## -0.60354515 0.33349306 -0.22282237 1.24041457 -0.28038843 -1.47783119   
## 13 14 15 16 17 18   
## -0.65303672 0.28865616 -0.10391829 2.96025507 0.17095826 0.40756162   
## 19 20 21 22 23 24   
## 0.96124413 0.76569814 1.28443310 0.89029407 0.01303494 0.01295953   
## 25 26 27 28   
## -0.02754601 -1.24791995 -1.58338033 -1.51206192

# Will show standardized residuals   
# Taht's great when we have a small dataset, but if its not small, then don't use the above code   
  
which.max(GoldModel$residuals)

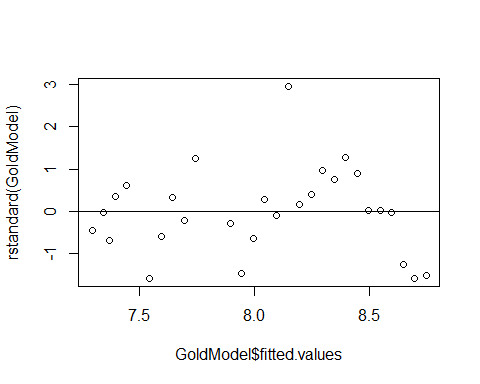
## 16   
## 16

rstandard(GoldModel)[16] # This will target the key point we ar elooking at

## 16   
## 2.960255

**We now know it’s an outlier, BUT WE DONT KNOW IF IT HAS ANY INFLUENCE ON TEH MODEL YET** - Can plot the rstand of the residuals by the fitted values - The plot is going to look identical to the others, other htan the axies, so but it’s a different measure of scale - How much infleunce is that really having?

plot(rstandard(GoldModel)~GoldModel$fitted.values)  
abline(0,0)

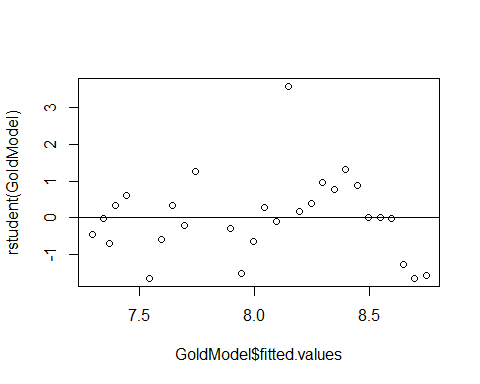
 **THING ABOUT THE STUDNTIZED RESIDUALS TO SEE HOW MUCH INFLUENCE SOMETHING HAS** - Standard = Outliers - Student = Influence (uses a different standard deviation)

**Studentized Residual** - Takes the single data out of the dataset, make a new regression line and get a new std of resid and seeing how far away the new line is from the outlier point in terms of the new standard residuals - If we take out an oulter, the std of the residuals is going to go down because we are removing an extreme case, so now its going to take more std to get to the outlier, and its going to give us a value that is bigger - When we see a studentized residual that is larger thant the stadnard residual, then it tells us that by removing this point, we are reallying changing the varibaility of the model and condenseing it more. - uses a different standard deviation than standariza

* **Concern:** An unusual value may exert great influence on the fit
* Its residual might be underestimated because the model “moves” a lot to fit it and/or
* The standard error of regression may be inflated due to the outlier error
* **Studentize:** Fit the model without that case, then find new 𝑦−𝑦hat and 𝜎\_𝜀 (of the population) to standardize. (R does this for every point)

**Influence** The effect of a single data point on the regression line depends on: - how well it matches the “trend” of the rest of the points - how “unusual” is its predictor value

plot(rstudent(GoldModel)~GoldModel$fitted.values)  
abline(0,0)



rstudent(GoldModel)[16]

## 16   
## 3.565083

# When we took the outlier from teh thing, its more away, which is bigger   
# There is some kind of influence here, but is it really noticable influence?

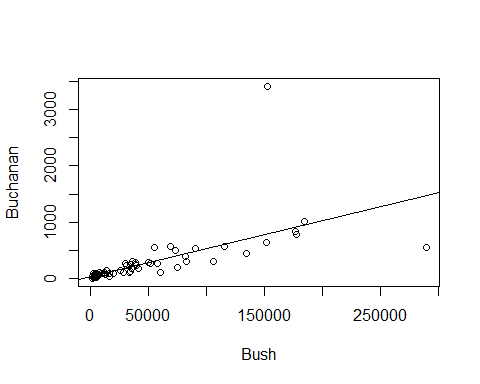
# plot(IceModel3) # From the previous notes   
  
# max(rstandard(IceModel3))  
# max(rstudent(IceModel3))  
# When we look at the values, themore different they are the mode influence they have in the model   
# The more close they are, then they have less influence on the model   
# No real bounds on what is a big or little influence on set number ot look at

**Dataset: PalmBeach** - County vote counts in Florida (n=67) for George Bush and Pat Buchanan in 2000. - Model: Use Bush votes to predict Buchanan votes.

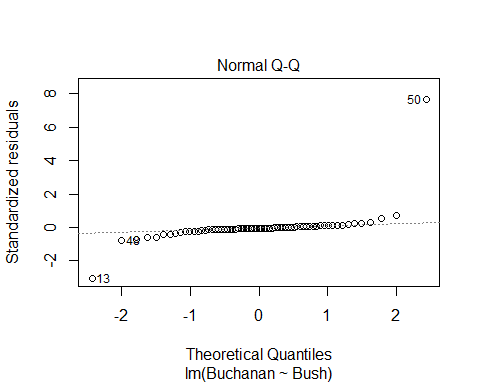
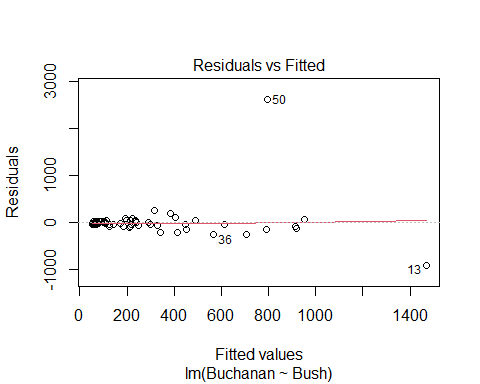
data(PalmBeach)  
head(PalmBeach)

## County Buchanan Bush  
## 1 ALACHUA 262 34062  
## 2 BAKER 73 5610  
## 3 BAY 248 38637  
## 4 BRADFORD 65 5413  
## 5 BREVARD 570 115185  
## 6 BROWARD 789 177279

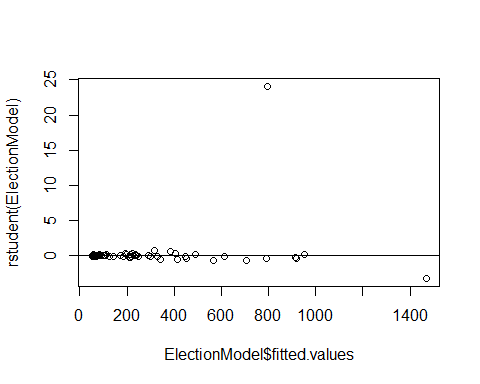
ElectionModel = lm(Buchanan~Bush, data=PalmBeach)  
# Bush = republican   
# Buchana - the other person   
  
plot(Buchanan~Bush, data=PalmBeach) # Look at the residuals and check the conditions   
abline(ElectionModel)

 **Example: Palm Beach Butterfly Ballot** - Palm Beach is the outlier

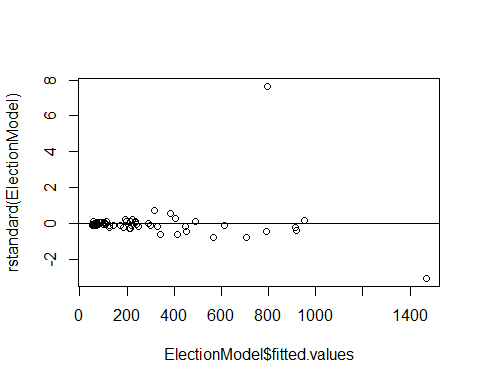
plot(ElectionModel, 1:2)



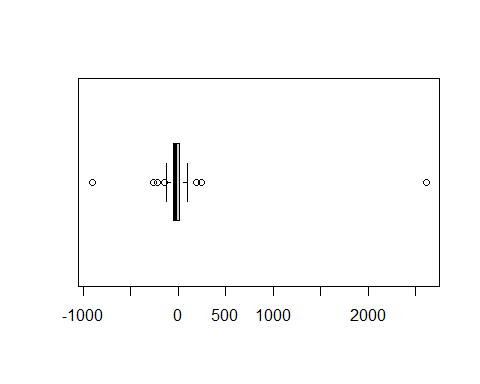
plot(rstudent(ElectionModel)~ElectionModel$fitted.values)  
abline(0,0)



plot(rstandard(ElectionModel)~ElectionModel$fitted.values)  
abline(0,0)



boxplot(ElectionModel$residuals, horizontal=TRUE) # Look at the outliers, there looks like ther eare a lot of outliers, but it's okay we are only looking at one of them



**What to do with an extreme residual?** - Try a transformation (Loging works really well, log(Data)) - Redo the analysis with the point omitted

*Below, we redo with the point omitted*

newdata = subset(PalmBeach, County!="PALM BEACH")  
  
ElectionModel\_noPB = lm(Buchanan~Bush, data=newdata)  
  
summary(ElectionModel)

##   
## Call:  
## lm(formula = Buchanan ~ Bush, data = PalmBeach)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -907.50 -46.10 -29.19 12.26 2610.19   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 4.529e+01 5.448e+01 0.831 0.409   
## Bush 4.917e-03 7.644e-04 6.432 1.73e-08 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 353.9 on 65 degrees of freedom  
## Multiple R-squared: 0.3889, Adjusted R-squared: 0.3795   
## F-statistic: 41.37 on 1 and 65 DF, p-value: 1.727e-08

summary(ElectionModel\_noPB)

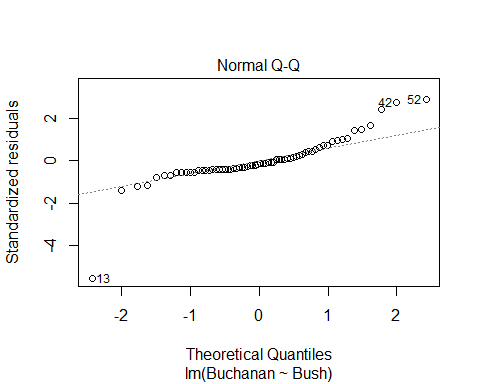
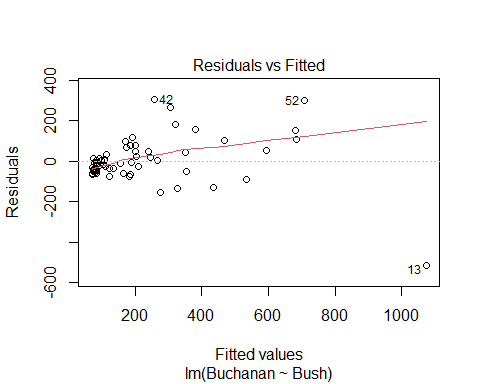
##   
## Call:  
## lm(formula = Buchanan ~ Bush, data = newdata)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -512.43 -47.97 -17.09 41.78 305.45   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 6.557e+01 1.733e+01 3.784 0.000343 \*\*\*  
## Bush 3.482e-03 2.501e-04 13.923 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 112.5 on 64 degrees of freedom  
## Multiple R-squared: 0.7518, Adjusted R-squared: 0.7479   
## F-statistic: 193.8 on 1 and 64 DF, p-value: < 2.2e-16

# Compare teh summary of old vs new model   
# We see that the new model residuals std error is really high; it s std value   
# The origanl value is a lot smaller for std

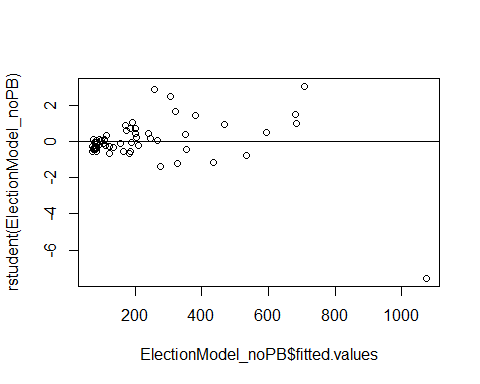
**Model with/without Palm Beach** - If compare the rstudent and rstandard of the with palm beach, we see that there is a really big jump between tehse numbers; tells us we have a value that is taking the whole regression and dragging it - So all our other predictiosn are going to be higher becuase of this vlaue

* If we look at the intercept and slope
* the slope is more dramatic when we take out the point, with the intercept goes down
* the residual standarad errors are very different between teh two
* the linearirity is really good without the palm beach
* the normal residuals look pretty bad when you take out palm beach

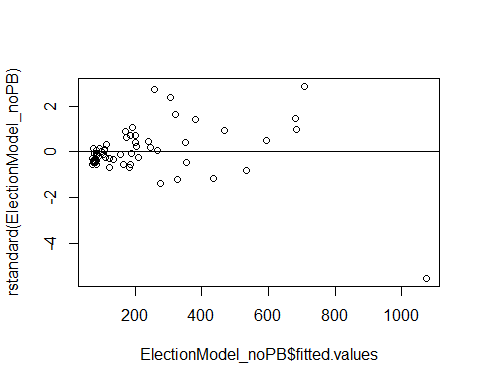
plot(ElectionModel\_noPB, 1:2)



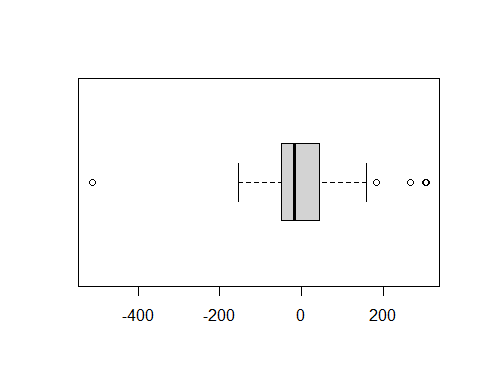
plot(rstudent(ElectionModel\_noPB)~ElectionModel\_noPB$fitted.values)  
abline(0,0)



plot(rstandard(ElectionModel\_noPB)~ElectionModel\_noPB$fitted.values)  
abline(0,0)



boxplot(ElectionModel\_noPB$residuals, horizontal=TRUE)



**BE careful with the student and standard** - look at how differen tbetween student and standard, you cna’t just know based on the one value

## STOR 455 Class 7 Outliers and Points of Influence Again

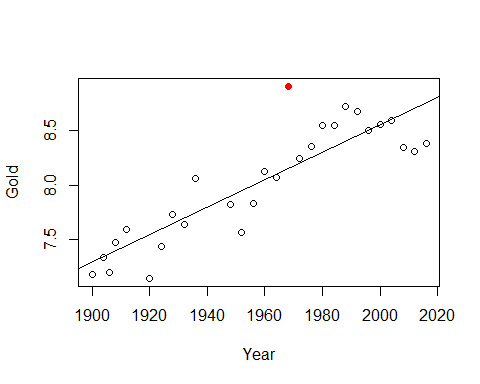
library(Stat2Data)  
  
data("LongJumpOlympics2016")  
data("PalmBeach")

**Types of “Unusual” Points in SLM** -**Outlier:** A data point that is far from the regression line. -**Influential point:** A data point that has a large effect on the regression fit.

**Detecting Unusual Cases - Overview** 1. Compute residuals - “raw”, standardized, studentized 2. Plots of residuals (or std. residuals) - Boxplot, scatterplot, normal plot 3. Leverage - Unusual values for the predictors 4. Cook’s distance - Cases with large influence

*Below:* We are making the same model as last time, in class 06

GoldModel=lm(Gold~Year, data=LongJumpOlympics2016)  
  
plot(Gold~Year, data=LongJumpOlympics2016)  
abline(GoldModel)  
points(LongJumpOlympics2016$Year[16], LongJumpOlympics2016$Gold[16], col="red", pch=16)



# Points just stands out a certain point   
# We need to give it the x and y coordinates, the the color and style   
  
summary(GoldModel) # The 0.2595 meaters is the standard error of the residual

##   
## Call:  
## lm(formula = Gold ~ Year, data = LongJumpOlympics2016)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.39610 -0.15495 -0.00137 0.11606 0.75349   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -16.470194 2.666282 -6.177 1.56e-06 \*\*\*  
## Year 0.012508 0.001361 9.191 1.19e-09 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2595 on 26 degrees of freedom  
## Multiple R-squared: 0.7646, Adjusted R-squared: 0.7556   
## F-statistic: 84.47 on 1 and 26 DF, p-value: 1.192e-09

**Standardized Residuals** - Roughly equal to zscore - For residuals: mean=0 and std. dev. ≈𝜎 ̂\_𝜀 - Standardized Residual about = (yi-yhat)/std - Look for values: beyond +/-2 (mild) or beyond +/-3 - **Definition:** The standardized residuals are: – std. resi = ((yi-yhat)/(stdsqrt(1-hi))); where hi = leverage

*SQRT 1=leverae* - It wont tell us that it has influence, but it will tell us the potential for influence -

StanResidEst = GoldModel$resid/summary(GoldModel)$sigma # Pulls out the standard error of the residuals   
  
StanResidEst - rstandard(GoldModel) # What this leverage is

## 1 2 3 4 5   
## 0.0310213900 0.0014925005 0.0404484002 -0.0187751023 -0.0306759692   
## 6 7 8 9 10   
## 0.0624609303 0.0211917539 -0.0104655385 0.0062653144 -0.0314031236   
## 11 12 13 14 15   
## 0.0055095767 0.0275829381 0.0118383564 -0.0052073979 0.0019121259   
## 16 17 18 19 20   
## -0.0568637884 -0.0034991146 -0.0090387925 -0.0233983067 -0.0206451799   
## 21 22 23 24 25   
## -0.0385895552 -0.0299054544 -0.0004902075 -0.0005455954 0.0012967412   
## 26 27 28   
## 0.0655674849 0.0926377694 0.0982533934

**What is this leverage?** - Depending on where the value is, depending ont eh x coord, determines the leverage - Things that follow the mean along the predictor because it doesn’t have a lot of leverage - If it’s twoards the right or left of end the range of values, then even a small, very little values could have a lot fo influence on the model - Think of leverage as a seesaw. – if you have a long seesaw on one side, but short on the other side, it doesn’t matter who is on the short side, they won’t have as much leverage. – If you have a long seesaw on the other side, then if it’s an adult or child may impact the stronger amount of leverage. Think of if it was an adult of child by how far up or down the y axis it goes. If it’s an adult, it’s oging to be a bigger different from the axis, but if it’s a child it’s going to be smaller and have less influence. - **Bottom line** A lower weight, if its further away fromt eh balance point, can have more effect on the model than a higher weight that is closer to the balance point – **The balance point is the mean**

StanResid = GoldModel$resid/(summary(GoldModel)$sigma \* sqrt(1 - hatvalues(GoldModel)))  
  
StanResid - rstandard(GoldModel)

## 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 27 28   
## 0 0

**Studentized Residuals** - If we took that point out of the model, how does that impact the variability of the residuals? - Definition: The studentized residuals are: stud.resi = *same as standardized, but without the one point* - If th epoint had a lot of influence on the varibaility, then the studentized value would be much bigger than the standarized value

* **The more different the standardized vs studentized values are, the more influence that point has on the model**

**Typical Leverage** - **For one point:** - For a simple linear model: hi = (1/n)+(((xi-xbar)2)/sum(xi-xbar)2) - **For all leverage in a model:** - sum(leverage) = sum(1/n) + ((sum(xi-xbar)2)/sum(xi-xbar)2) = 1+1 = 2 - The sum of the leverages of all of the points = 2 – this is usefulbecause we can think of what a typical leverage value is – If all teh leverages are 2, then 2/n = give you what the mean leverage is - *Look for:* – leverage > 2(2/n) – leverage hi > 3(2/n)

*Notes* - How different is this point from the average predictive value in compraision to all those other values - Xi - xbar = predictor where the xi = a specific predictor value, and xbar = the mean of the values - Xi-xbar squared = in a horizontal way, how different is that point regardless if above or below mean - The sum(xi-xbar)^2 = how different is this point proportionally to the rest of the data - scared accoriding the the sample of the data with the 1/n - One point will haev mor einfluence if the data is small - values that are double or triple the average leverage are the ones we have to worry about because they may have influence in the model – **We are still talking about potenital here because we are only looking at the predictors** –Because we are only looking at years in this model, vs the entire context of the data –It tells us the data points we should look at more closely

# average leverage : 2/28   
# IF poitns are bigger than this, might have a lot of influence on the model   
  
2\*(2/28)

## [1] 0.1428571

3\*(2/28)

## [1] 0.2142857

# Above are just the cut off points for influence, we don't really care too much about them   
  
# The below code will show the ordered data because of the way the data is organized in the dataset   
# We can see how the leverages are going to be similar there where the points that are small, very different for the mean, they wil start with a higher leverage, and by the time we get the middle values, its going down to a leverage of nearly zero,   
# Get further waya from average year, we will get higher leverages reported   
  
hatvalues(GoldModel)

## 1 2 3 4 5 6 7   
## 0.13075010 0.11825693 0.11234035 0.10664378 0.09591065 0.07708445 0.06899139   
## 8 9 10 11 12 13 14   
## 0.06177835 0.05544533 0.04999234 0.03891349 0.03698058 0.03592770 0.03575484   
## 15 16 17 18 19 20 21   
## 0.03646200 0.03804918 0.04051638 0.04386361 0.04809086 0.05319813 0.05918543   
## 22 23 24 25 26 27 28   
## 0.06605275 0.07380009 0.08242745 0.09193483 0.10232224 0.11358967 0.12573712

# Iif you look at the 16th value, it has a leverage of 0.038; based onteh creitera foor what tis a big lleverage value, this point doesn't appear to be high leverage   
# the 16the point has to be a REALLY big outlier for it to have influence, because its potential to have influence is low

**Wanted to look at the leverages better** - Sort them to have a better idea of where the potenitals are to gave influence - Sort the hat values - sort will by default go from acending order/increasing order - mostly interested in the biggest leverages - Want to change it to decreasing = TRUE so we see the biggest leverages first instead of the smallest ones first

2\*(2/28)

## [1] 0.1428571

3\*(2/28)

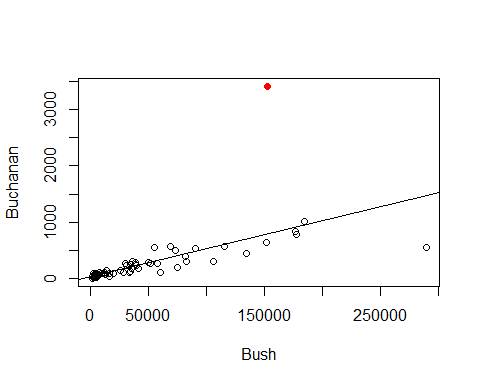
## [1] 0.2142857

sort(hatvalues(GoldModel), decreasing = TRUE)

## 1 28 2 27 3 4 26   
## 0.13075010 0.12573712 0.11825693 0.11358967 0.11234035 0.10664378 0.10232224   
## 5 25 24 6 23 7 22   
## 0.09591065 0.09193483 0.08242745 0.07708445 0.07380009 0.06899139 0.06605275   
## 8 21 9 20 10 19 18   
## 0.06177835 0.05918543 0.05544533 0.05319813 0.04999234 0.04809086 0.04386361   
## 17 11 16 12 15 13 14   
## 0.04051638 0.03891349 0.03804918 0.03698058 0.03646200 0.03592770 0.03575484

# The one with the biggests index is 0.1307, even the smallest and biggest predicotr value aresnt so far from teh mean predictor compared to the other values that hey have potiental for influence.   
# They would need to be big outliers to have influence   
# If we had more datapoints, then we could also just look at some of the values   
# The head function can be added to the sorting hatvalues

ElectionModel=lm(Buchanan~Bush,data=PalmBeach)  
plot(Buchanan~Bush,data=PalmBeach)  
abline(ElectionModel)  
points(PalmBeach$Bush[50], PalmBeach$Buchanan[50], col="red", pch=16)



# Red is palm beach

#Average leverage = 2/67  
  
#Potiential for influence   
2\*(2/67)

## [1] 0.05970149

3\*(2/67)

## [1] 0.08955224

head(sort(hatvalues(ElectionModel), decreasing=TRUE), n=10)

## 13 52 6 29 50 16 48   
## 0.29747301 0.10761608 0.09859725 0.09820784 0.07085197 0.07007421 0.05365982   
## 5 36 53   
## 0.03899504 0.03331541 0.02511932

# We see that point 13 probably has potiential leverage   
  
# We want to see the first 6 with the highest potienital for influence so we can look at them closer  
# We want all columns, so just leave it with the ,  
# This tells us teh counties in the order of teh output, not the leverages though   
# Want to add the leverages to it though   
PalmBeach[c(6,13,16,29,50,52),]

## County Buchanan Bush  
## 6 BROWARD 789 177279  
## 13 DADE 561 289456  
## 16 DUVAL 650 152082  
## 29 HILLSBOROUGH 836 176967  
## 50 PALM BEACH 3407 152846  
## 52 PINELLAS 1010 184312

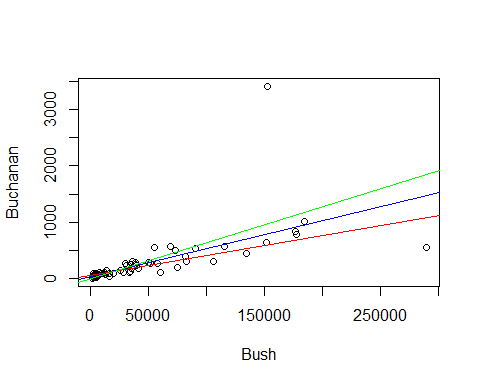
# Defines new variable inside of the dataframe  
PalmBeach$Leverage = hatvalues(ElectionModel)  
# This is putting the leverage into the dataframe   
  
# Now run this again: PalmBeach[c(6,13,16,29,50,52),]  
PalmBeach[c(6,13,16,29,50,52),]

## County Buchanan Bush Leverage  
## 6 BROWARD 789 177279 0.09859725  
## 13 DADE 561 289456 0.29747301  
## 16 DUVAL 650 152082 0.07007421  
## 29 HILLSBOROUGH 836 176967 0.09820784  
## 50 PALM BEACH 3407 152846 0.07085197  
## 52 PINELLAS 1010 184312 0.10761608

# This will now include the palmbeach levearge

*Want to visualize the impact dave county has vs palm beach* - Dave has a smaller outlier, it’s leverage is high so it could havea similar impact on the model

plot(Buchanan~Bush,data=PalmBeach)  
abline(ElectionModel, col="blue")  
# The above is the model we have now   
  
# Below, make new model with no palmbeach   
NoPalmBeach=subset(PalmBeach,County!="PALM BEACH")  
ElectionModel\_noPB=lm(Buchanan~Bush,data=NoPalmBeach)  
abline(ElectionModel\_noPB, col="red")  
  
# Below, make a new model with no dade  
NoDade = subset(PalmBeach,County!="DADE")  
ElectionModel\_noD=lm(Buchanan~Bush,data=NoDade)  
abline(ElectionModel\_noD, col="green")



# Then we compare the lines that we have written to see how the slope is different.   
# Red = no palm beach   
# Green = no dade   
# Blue = with both   
# The change in slope appears the same, so it tells me that the impact these datapoint shave on our model are about similar   
# Even though one is a bigger outlier, it has a lower potenital to have infleunce,   
# The other is smaller outlier, but has a high leverage, it has abotu the same typ eof influence

*NOtes* - FOr high leverage, doesnt need to be a big outlier to have an impact on the model - for high outlier, it doesnt have to have that much leverage - We are looking at cook’s distance, which combines teh two above things and quantify the value instead of the other vlaues - Cook’s will tell you if the pointhas influence on the model - Brings in stanard resid; the leverage of the point; and how many predictors there are - Cooks’ distance is acombien of leverage, student, adn standard

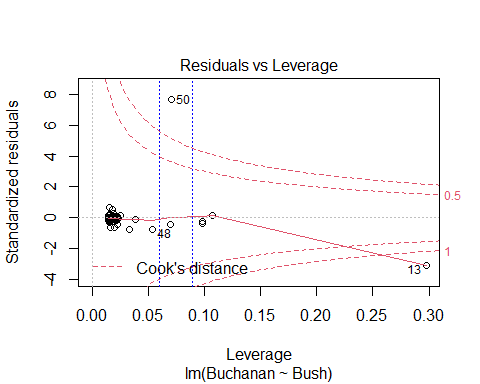
**Cook’s Distance** - How much would the fit change if one data value were omitted? - Di increases with either poor fit (std.resi) and high leverage (hi). 1. Compare to other Di’s. 2. Study any case with Di > 0.5; worry if Di > 1.0.

*Cook’s Di:* = (((std.resi)^2)/(k+1))\*(hi/(1-hi))

head(sort(cooks.distance(ElectionModel), decreasing=TRUE), n=5)

## 50 13 48 36 6   
## 2.231935359 1.981365681 0.016228381 0.009781056 0.007928585

# Two ar emuch bigger adn the rest of smaller, those two points rae the main that are havign influence on our model   
# 50 = =palm beach   
# 13 = Dade   
# They're not exactly the same, but have about teh same impact ont eh model we are making   
  
#We've seen Cook's distance between with teh plots for the model   
plot(ElectionModel,5)  
# Will show you what point shave leverage   
# LOOK AT WHAT THE THING IS   
# HOw big of an outlier is this value   
# Mode values have standardiex values within +/- 2  
# Th etwo that have big differences are point 50 and 13  
# Point 50 = std score of 8   
# Point 13, Dade, has a -3.0 std value; but it is high leverage   
# Points low leverage = closer to the left side of the graph   
# Further right = different data, more leverage   
  
# O.5  
# Gives us a cook's distance of 0.5  
# OUtler line: 1 = cook's distance of 1   
# Above these lines, we can claim that by putting the leverage and output together it's having an influence on the model   
# POint 50, has small leverage, but a big outlier that it's goig outside of the curve and having influence on the model   
# Dade county has a huge leverage, but it's not an outlier, but it's enough to put it outside of the curve and have an high impact on teh model   
  
# Think about: WHY DO THESE POINTS HAVE IMPACT ON THE MDOEL>   
# DO WE WANT THEM TO IMPACT THE MODEL?   
# sometimes might not want to inlcude some things int eh model, but it's good to look at it anyways so you know what is going on   
abline(v = 4/67, col="blue", lty=3)  
abline(v = 6/67, col="blue", lty=3)



## STOR 455 Class 8 Inference for Regression Slope

library(readr)  
library(Stat2Data)  
library(metRology)  
  
DistanceHome <- read\_csv("https://raw.githubusercontent.com/JA-McLean/STOR455/master/data/DistanceHome.csv")  
Domestic=subset(DistanceHome,Distance<250)  
  
source("https://raw.githubusercontent.com/JA-McLean/STOR455/master/scripts/CIPIPlot.R")   
# Function that is not built into the thing   
# The source will allow you to call the function as you want

*NOtes* - If the data is packed around the line, then we can predict that the data will be by the line

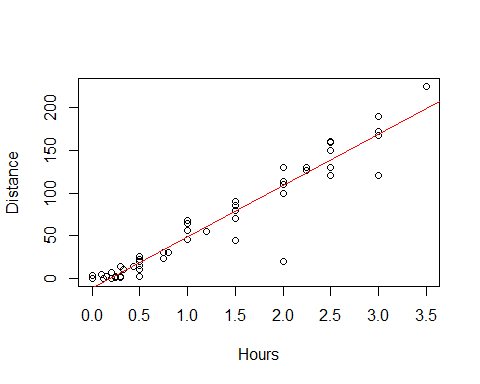
**Inference for Slope and Intercept** - Find a confidence interval of plausible values for the parameter. - Test a hypothesis about a possible value for the parameter

**Bootstrap Distribution** - THink of poopulation - Estimate the distribution and variability (SE) of β ̂\_𝑖 from the bootstraps - WE ar elooking atht the distnace data, - THink of the population and there is some population slope here - If I knew everyone’s distance from home, this would be the realsiton they have; we know the sample but not the population - It would be nice if we could take a lot of samples ad keep calculating that regression line (then you could see teh distribution) - Think of seeds that fall from a tree; from each seed that falls from teh tree we get a slope from that line. - We cant keep taking from teh ample, and we need to find where the actual population vlaue is

* Hypothesis what the population might look like based on this sample
* Maybe the population is just a lot of copies of this sample
* If we take samples form teh bootstrap poulation, what variability is there?
* We are making many copies of them
* We just have out 54 key people with our data, taking one out and seeing where they fall on the regression line, putting htem back and then pulling someone else – If you do this a bunch of times, sometimes you ll get a sample that is like the population or not; youll get a range of values

**Simple Linear Regression** *See below*

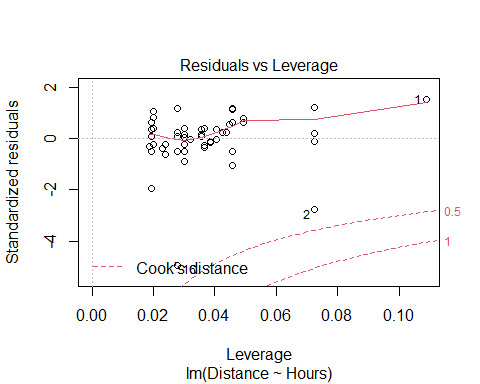
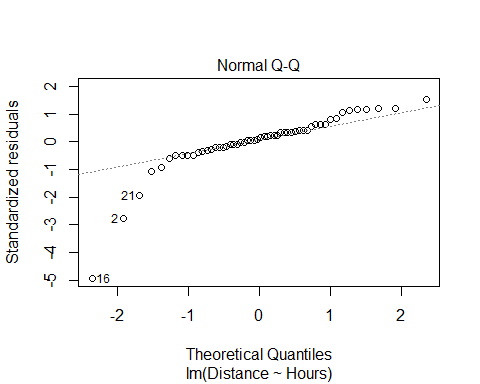
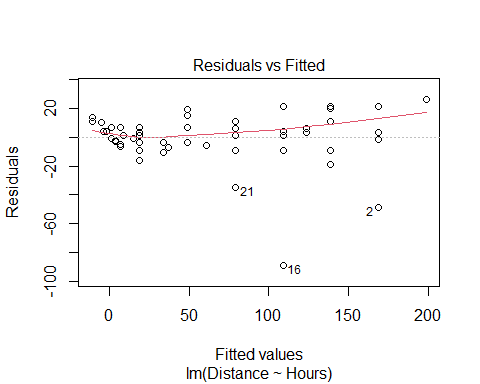
moddist=lm(Distance~Hours, data=Domestic)  
# Hours per distnace; predict distance form home based on hours it takes you to get home   
  
plot(Distance~Hours, data=Domestic)  
abline(moddist, col="red")



summary(moddist)

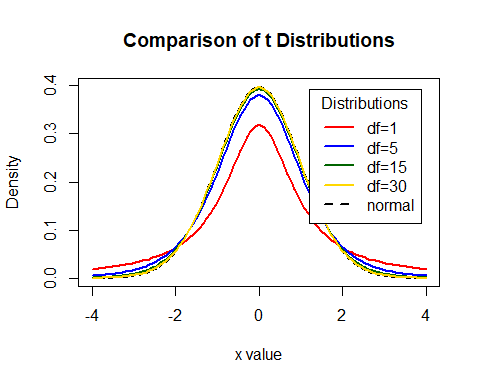
##   
## Call:  
## lm(formula = Distance ~ Hours, data = Domestic)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -88.892 -4.680 2.172 7.082 26.141   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -11.063 4.056 -2.727 0.00868 \*\*   
## Hours 59.977 2.484 24.144 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 18.26 on 52 degrees of freedom  
## Multiple R-squared: 0.9181, Adjusted R-squared: 0.9165   
## F-statistic: 582.9 on 1 and 52 DF, p-value: < 2.2e-16

# Slope = 59; for every 1 hour people ar eaway from home, then they are about 60 miles more away from home   
# We want to think about; How close is this to the population value?   
# What claims can I make about the population?  
  
plot(moddist, c(1, 2, 5))



# THis whole chunk of code creates a linear model, plots it against the data, and checks the conditions.   
# Pretty linear relaitonship overall: hours away from home, distance increase   
# Residuals have a curve that s alittle ; overall it doesnt seem very bad   
# normal QQ plot, ther eis a problem where one side has a skew   
  
#COok's ditance   
# High leverage, far right   
# High influence, up/down

# Display the Student's t distributions with various  
# degrees of freedom and compare to the normal distribution  
  
x <- seq(-4, 4, length=100)  
hx <- dnorm(x)  
  
degf <- c(1, 5, 15, 30)  
colors <- c("red", "blue", "darkgreen", "gold", "black")  
labels <- c("df=1", "df=5", "df=15", "df=30", "normal")  
  
plot(x, hx, type="l", lty=2, xlab="x value",  
 ylab="Density", main="Comparison of t Distributions")  
  
for (i in 1:4){  
 lines(x, dt(x,degf[i]), lwd=2, col=colors[i])  
}  
  
legend("topright", inset=.05, title="Distributions",  
 labels, lwd=2, lty=c(1, 1, 1, 1, 2), col=colors)



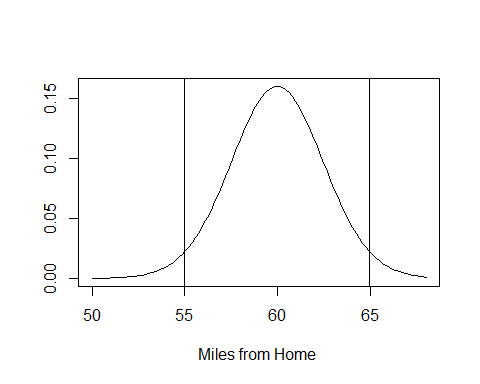
# What the tdist looks like   
#df comes down to sample size and other factors (predicotrs and modeltype)   
# Bigger sample, more degree of freedom   
# For 30 df, a sample of 32, this is pretty much a normal curve   
# As it gests smaller, we are pressing down and pressing out the sides of the graph   
#We are triyng to make this based on what the standard error of our sample is

**CI for Slope or Intercept** - HOw R is doing the math - BUild a distribution where the cente ris where the sample value is adnd we are making a curve on top of that and how far in each direction do we need to go were the middle 95% of the area under the curve is met? Bihar +/- tstar*StandardError or the Bihat Y = Bo + B1X+E - t* comes from a t-distribution with n-2 d.f and depends on the level of confidence - For 1−𝛼 level confidence, use qt(1-α/2,df)in R - **e.g. for 95% confidence and 52 df, qt(0.975,52)** - tstar = tells us how many STDERRORS in each direction we need to go

qt(0.975, 52)

## [1] 2.006647

# T = t distribution   
# area under curve to teh left of the point we want = first arguement; 0.975 = 95% confi int   
# df = 2nd argument; sampel size minus 2   
# Result = 2.00664 ish   
#If i want to see the confidence interval, you have to start at your sample of slope  
#qt gives you tstar  
  
curve(  
 dt.scaled(  
 x,   
 52,  
 mean = summary(moddist)$coef[2,1],  
 sd = summary(moddist)$coef[2,2]  
 ),   
 from = 50, to = 68,  
 xlab = "Miles from Home ",  
 ylab = " "  
 )  
  
  
abline(  
 v=c(  
 qt.scaled(  
 0.025,   
 52,   
 mean = summary(moddist)$coef[2,1],   
 sd = summary(moddist)$coef[2,2]  
 ),  
 qt.scaled(  
 0.975,   
 52,   
 mean = summary(moddist)$coef[2,1],   
 sd = summary(moddist)$coef[2,2]  
 )  
 )  
 )



# IF you want to see the confidence interval   
summary(moddist)$coef[2,1]-qt(0.975, 52)\*summary(moddist)$coef[2,2] #LOwer bound for confidence interval

## [1] 54.99264

summary(moddist)$coef[2,1]+qt(0.975, 52)\*summary(moddist)$coef[2,2] #Upper bound for confidence interval

## [1] 64.96233

#We are predicting that with 95% confidence the solution is between the 54.99-64.96

If we think tha the population might not be normal, then we probably want to do a bootstrap method

**How to find a confidence interval?** -confint(mymodel,level =0.XX) and adjust for the confidence level.

# HOW TO FIND CONFIDENCE INTERVAL; default 95% confidence   
# generally the intercept is not very useful to think about   
# We are predicting taht it could be close to zero   
# WE are mostly looking at the coeffs   
# The hours are about teh same as above   
confint(moddist, level=0.95)

## 2.5 % 97.5 %  
## (Intercept) -19.20166 -2.92358  
## Hours 54.99264 64.96233

**Accuracy of Predictions** Example: It takes a student 2.25 hours to drive from home. How many miles do we predict that thy are away from home? How accurate is that prediction? - Want to make a prediction for a specific case - Wnted regalr prediction, just plug the 2.25 into the regression line - It matterse what you are rtrying to predict - all people or the specific person’s distance from home? - There is a difference; one person has ore variability (Say they’re biking) - If we have the ditribution, ontop of that is some normal curve; most of the people are close to thtat, but they trail off a bit

**Two Forms of Intervals for Regression** 1. Confidence Interval for μY (mean Y) Where is the “true” line for that x? or Where is the average Y for all with that x? 2. Prediction Interval for Individual Y Where are most Y’s for that x?

\_\_CI for μY when X=x\*\_\_ - Predicting in general SSX = ∑▒〖(𝑥\_𝑖 − 𝑥 ̅)〗^2 yhat +/- tstar*standerror*sqrt((1/n)+((xstar-xbar)^2)/SSX)

\_\_Prediction Interval for Individual Y’s when X=x\*\_\_ - predicting for one person yhat +/- tstar*standerror*sqrt(1+ (1/n)+((xstar-xbar)^2)/SSX) Just add 1 in the sqrt

\_\_CI and PI via R when X=x\*\_\_

newx=data.frame(Hours=2.25) # Creat e a new person   
head(newx)

## Hours  
## 1 2.25

predict.lm(moddist, newx, interval="confidence") # Predict mean for all people who are 2.25 away from hoe

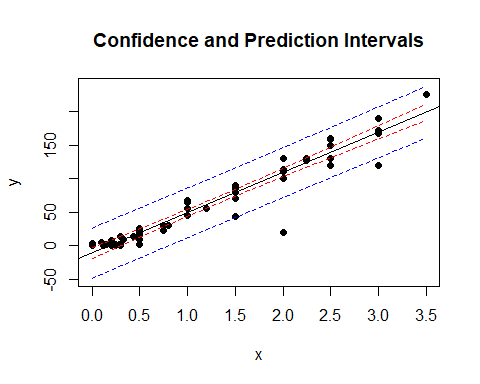
## fit lwr upr  
## 1 123.8867 116.9764 130.797

predict.lm(moddist, newx, interval="prediction") # Distance home for one specific person

## fit lwr upr  
## 1 123.8867 86.59458 161.1789

# Both gives us a fitted value, its a point in the regression lie   
# THis si what we would get if we plugged and chugged   
# One person is going to be a wider range because we want to make sure we get teh one person

CIPIPlot(Domestic$Hours, Domestic$Distance) # Visualize different between confidence and prediction



# calculates   
# For every possible point in teh data, or for the  
# What would be the confidence interveral for that value and what would be the prediction interval

* The red lines are the confidence interval
* if we are trying to predict the mean vlaue for people’s distance away from home based on tehse hours, we will predict the mean is somewhere between the red lines and its tight by the regression line with 95% confidence Th eblue line = much wider
* there’s a lot more variability there
* much wider

## STOR 455 Class 8 Inference for Regression Slope

library(readr)  
library(Stat2Data)  
library(metRology)  
  
DistanceHome <- read\_csv("https://raw.githubusercontent.com/JA-McLean/STOR455/master/data/DistanceHome.csv")  
Domestic=subset(DistanceHome,Distance<250)  
  
source("https://raw.githubusercontent.com/JA-McLean/STOR455/master/scripts/CIPIPlot.R")   
# Function that is not built into the thing   
# The source will allow you to call the function as you want

*NOtes* - If the data is packed around the line, then we can predict that the data will be by the line

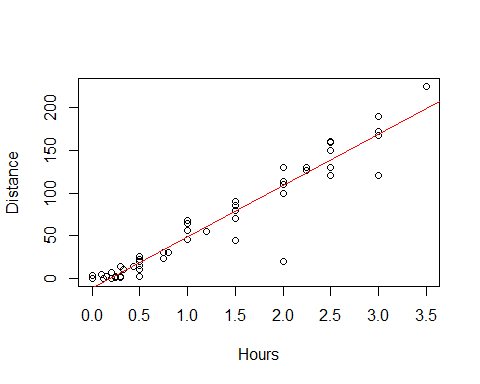
**Inference for Slope and Intercept** - Find a confidence interval of plausible values for the parameter. - Test a hypothesis about a possible value for the parameter

**Bootstrap Distribution** - THink of poopulation - Estimate the distribution and variability (SE) of β ̂\_𝑖 from the bootstraps - WE ar elooking atht the distnace data, - THink of the population and there is some population slope here - If I knew everyone’s distance from home, this would be the realsiton they have; we know the sample but not the population - It would be nice if we could take a lot of samples ad keep calculating that regression line (then you could see teh distribution) - Think of seeds that fall from a tree; from each seed that falls from teh tree we get a slope from that line. - We cant keep taking from teh ample, and we need to find where the actual population vlaue is

* Hypothesis what the population might look like based on this sample
* Maybe the population is just a lot of copies of this sample
* If we take samples form teh bootstrap poulation, what variability is there?
* We are making many copies of them
* We just have out 54 key people with our data, taking one out and seeing where they fall on the regression line, putting htem back and then pulling someone else – If you do this a bunch of times, sometimes you ll get a sample that is like the population or not; youll get a range of values

**Simple Linear Regression** *See below*

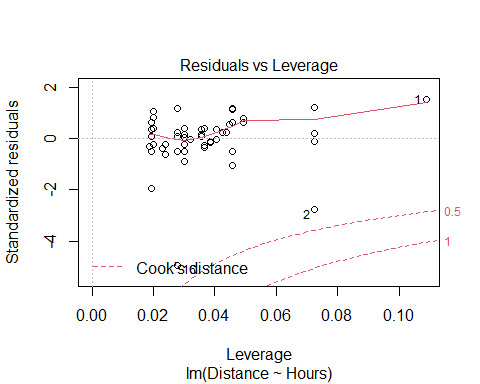
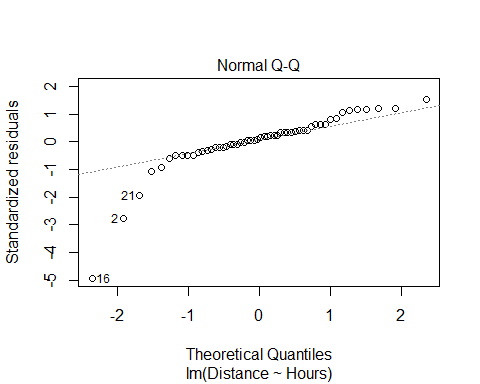
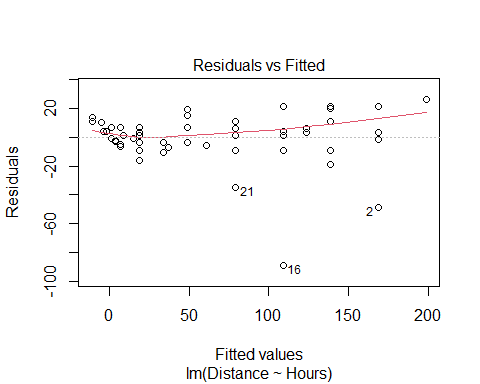
moddist=lm(Distance~Hours, data=Domestic)  
# Hours per distnace; predict distance form home based on hours it takes you to get home   
  
plot(Distance~Hours, data=Domestic)  
abline(moddist, col="red")



summary(moddist)

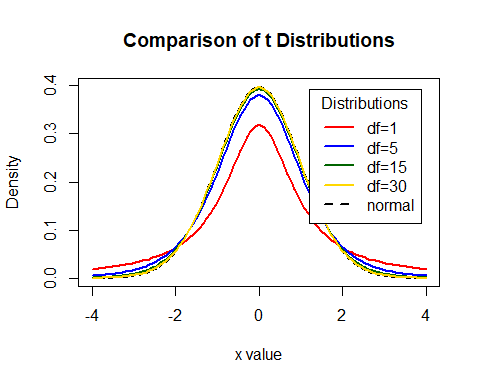
##   
## Call:  
## lm(formula = Distance ~ Hours, data = Domestic)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -88.892 -4.680 2.172 7.082 26.141   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -11.063 4.056 -2.727 0.00868 \*\*   
## Hours 59.977 2.484 24.144 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 18.26 on 52 degrees of freedom  
## Multiple R-squared: 0.9181, Adjusted R-squared: 0.9165   
## F-statistic: 582.9 on 1 and 52 DF, p-value: < 2.2e-16

# Slope = 59; for every 1 hour people ar eaway from home, then they are about 60 miles more away from home   
# We want to think about; How close is this to the population value?   
# What claims can I make about the population?  
  
plot(moddist, c(1, 2, 5))



# THis whole chunk of code creates a linear model, plots it against the data, and checks the conditions.   
# Pretty linear relaitonship overall: hours away from home, distance increase   
# Residuals have a curve that s alittle ; overall it doesnt seem very bad   
# normal QQ plot, ther eis a problem where one side has a skew   
  
#COok's ditance   
# High leverage, far right   
# High influence, up/down

# Display the Student's t distributions with various  
# degrees of freedom and compare to the normal distribution  
  
x <- seq(-4, 4, length=100)  
hx <- dnorm(x)  
  
degf <- c(1, 5, 15, 30)  
colors <- c("red", "blue", "darkgreen", "gold", "black")  
labels <- c("df=1", "df=5", "df=15", "df=30", "normal")  
  
plot(x, hx, type="l", lty=2, xlab="x value",  
 ylab="Density", main="Comparison of t Distributions")  
  
for (i in 1:4){  
 lines(x, dt(x,degf[i]), lwd=2, col=colors[i])  
}  
  
legend("topright", inset=.05, title="Distributions",  
 labels, lwd=2, lty=c(1, 1, 1, 1, 2), col=colors)



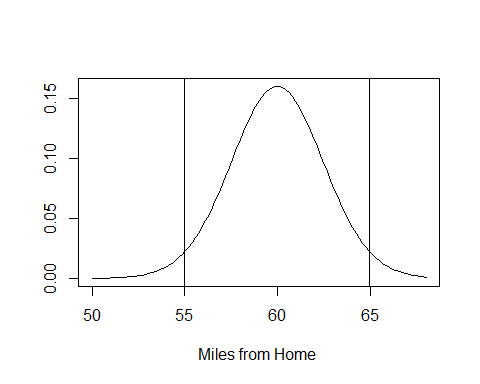
# What the tdist looks like   
#df comes down to sample size and other factors (predicotrs and modeltype)   
# Bigger sample, more degree of freedom   
# For 30 df, a sample of 32, this is pretty much a normal curve   
# As it gests smaller, we are pressing down and pressing out the sides of the graph   
#We are triyng to make this based on what the standard error of our sample is

**CI for Slope or Intercept** - HOw R is doing the math - BUild a distribution where the cente ris where the sample value is adnd we are making a curve on top of that and how far in each direction do we need to go were the middle 95% of the area under the curve is met? Bihar +/- tstar*StandardError or the Bihat Y = Bo + B1X+E - t* comes from a t-distribution with n-2 d.f and depends on the level of confidence - For 1−𝛼 level confidence, use qt(1-α/2,df)in R - **e.g. for 95% confidence and 52 df, qt(0.975,52)** - tstar = tells us how many STDERRORS in each direction we need to go

qt(0.975, 52)

## [1] 2.006647

# T = t distribution   
# area under curve to teh left of the point we want = first arguement; 0.975 = 95% confi int   
# df = 2nd argument; sampel size minus 2   
# Result = 2.00664 ish   
#If i want to see the confidence interval, you have to start at your sample of slope  
#qt gives you tstar  
  
curve(  
 dt.scaled(  
 x,   
 52,  
 mean = summary(moddist)$coef[2,1],  
 sd = summary(moddist)$coef[2,2]  
 ),   
 from = 50, to = 68,  
 xlab = "Miles from Home ",  
 ylab = " "  
 )  
  
  
abline(  
 v=c(  
 qt.scaled(  
 0.025,   
 52,   
 mean = summary(moddist)$coef[2,1],   
 sd = summary(moddist)$coef[2,2]  
 ),  
 qt.scaled(  
 0.975,   
 52,   
 mean = summary(moddist)$coef[2,1],   
 sd = summary(moddist)$coef[2,2]  
 )  
 )  
 )



# IF you want to see the confidence interval   
summary(moddist)$coef[2,1]-qt(0.975, 52)\*summary(moddist)$coef[2,2] #LOwer bound for confidence interval

## [1] 54.99264

summary(moddist)$coef[2,1]+qt(0.975, 52)\*summary(moddist)$coef[2,2] #Upper bound for confidence interval

## [1] 64.96233

#We are predicting that with 95% confidence the solution is between the 54.99-64.96

If we think tha the population might not be normal, then we probably want to do a bootstrap method

**How to find a confidence interval?** -confint(mymodel,level =0.XX) and adjust for the confidence level.

# HOW TO FIND CONFIDENCE INTERVAL; default 95% confidence   
# generally the intercept is not very useful to think about   
# We are predicting taht it could be close to zero   
# WE are mostly looking at the coeffs   
# The hours are about teh same as above   
confint(moddist, level=0.95)

## 2.5 % 97.5 %  
## (Intercept) -19.20166 -2.92358  
## Hours 54.99264 64.96233

**Accuracy of Predictions** Example: It takes a student 2.25 hours to drive from home. How many miles do we predict that thy are away from home? How accurate is that prediction? - Want to make a prediction for a specific case - Wnted regalr prediction, just plug the 2.25 into the regression line - It matterse what you are rtrying to predict - all people or the specific person’s distance from home? - There is a difference; one person has ore variability (Say they’re biking) - If we have the ditribution, ontop of that is some normal curve; most of the people are close to thtat, but they trail off a bit

**Two Forms of Intervals for Regression** 1. Confidence Interval for μY (mean Y) Where is the “true” line for that x? or Where is the average Y for all with that x? 2. Prediction Interval for Individual Y Where are most Y’s for that x?

\_\_CI for μY when X=x\*\_\_ - Predicting in general SSX = ∑▒〖(𝑥\_𝑖 − 𝑥 ̅)〗^2 yhat +/- tstar*standerror*sqrt((1/n)+((xstar-xbar)^2)/SSX)

\_\_Prediction Interval for Individual Y’s when X=x\*\_\_ - predicting for one person yhat +/- tstar*standerror*sqrt(1+ (1/n)+((xstar-xbar)^2)/SSX) Just add 1 in the sqrt

\_\_CI and PI via R when X=x\*\_\_

newx=data.frame(Hours=2.25) # Creat e a new person   
head(newx)

## Hours  
## 1 2.25

predict.lm(moddist, newx, interval="confidence") # Predict mean for all people who are 2.25 away from hoe

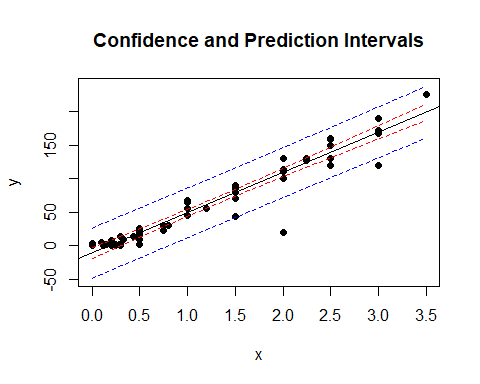
## fit lwr upr  
## 1 123.8867 116.9764 130.797

predict.lm(moddist, newx, interval="prediction") # Distance home for one specific person

## fit lwr upr  
## 1 123.8867 86.59458 161.1789

# Both gives us a fitted value, its a point in the regression lie   
# THis si what we would get if we plugged and chugged   
# One person is going to be a wider range because we want to make sure we get teh one person

CIPIPlot(Domestic$Hours, Domestic$Distance) # Visualize different between confidence and prediction



# calculates   
# For every possible point in teh data, or for the  
# What would be the confidence interveral for that value and what would be the prediction interval

* The red lines are the confidence interval
* if we are trying to predict the mean vlaue for people’s distance away from home based on tehse hours, we will predict the mean is somewhere between the red lines and its tight by the regression line with 95% confidence Th eblue line = much wider
* there’s a lot more variability there
* much wider

## STOR 455 Class 9 Partitioning Variability - ANOVA

library(readr)  
library(Stat2Data)  
  
DistanceHome <- read\_csv("https://raw.githubusercontent.com/JA-McLean/STOR455/master/data/DistanceHome.csv")  
  
Domestic = subset(DistanceHome, Distance<250)

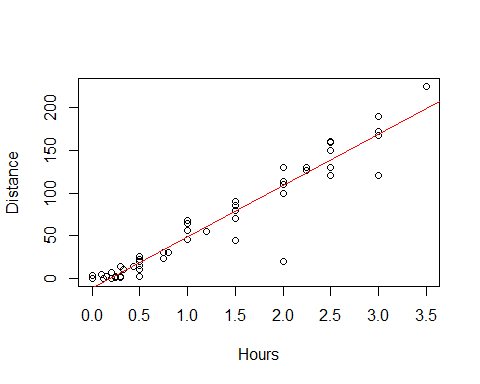
*NOtes* - Evidence to say if there is some kind of relationship exists? - How we center teh distributios is a little different

**T-test for Slope**  Ho: B1 = 0 (for Y = Bo + Error) Ha: B1 != 0 (for Y = Bo + B1X + Error)

t.s. = Bhat1/SEofBhat1 - Find p-value using a t-distribution and n-2 d.f. -p-value is small then Reject Ho

**How to find a P-value?** - Statistically significant evidence suggests ( p-value < 2.22\*10-16) that there is a relationship between the hours that a student spend travelling to campus and their distance from campus.

plot(Distance~Hours, data=Domestic) # Predict distance home, based onhow many hours it takes to get there   
moddist = lm(Distance~Hours, data=Domestic)  
abline(moddist, col="red") # Draw linear model on top of plot



summary(moddist) # Take summary of plot model

##   
## Call:  
## lm(formula = Distance ~ Hours, data = Domestic)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -88.892 -4.680 2.172 7.082 26.141   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -11.063 4.056 -2.727 0.00868 \*\*   
## Hours 59.977 2.484 24.144 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 18.26 on 52 degrees of freedom  
## Multiple R-squared: 0.9181, Adjusted R-squared: 0.9165   
## F-statistic: 582.9 on 1 and 52 DF, p-value: < 2.2e-16

# In coef table,   
# Does a hypothesis test for us   
# COmpares two models to eachother   
# The linear model to the constant model, where the model is just the mean   
# If the model is just a horizontal line, the it's that the slope would be zero  
# slope zero = no cheged based on distance from home   
# Alternative: AS hours change, the ditsnce from home changes   
# How likely is it that we get a sampel slope like we did, what the chance we would get what we did   
# The test stat is the tstat/Se = how many std err are we from teh null; how unlikely is it that we get the thing that we did   
  
# Slope: 59.99  
# STd err: 2.48 = Tightly bound   
# tvalue = 24. = if our null is 0, then we are 24 std errs from the null hyp of 0; thats unlikely to happen by chance   
# Pvalue: How unlikely we are to get a sample that we did, if the null is true; and we have a really low pvalue for this, which means that it is not likely we would get the null hypothesis

**Finding Correlation in R** -For data in two variables:

cor(Domestic$Distance, Domestic$Hours)

## [1] 0.9581758

**Finding Correlation in R** For all variables in a dataframe:

data(Houses)  
head(Houses)

## Price Size Lot  
## 1 212000 4148 25264  
## 2 230000 2501 11891  
## 3 339000 4374 25351  
## 4 289000 2398 22215  
## 5 160000 2536 9234  
## 6 85000 2368 13329

# Can put in whole dataframes   
# This will give you a corr matrix; a table with a ll possible combination snad the correlations for those variables   
  
cor(Houses)

## Price Size Lot  
## Price 1.0000000 0.6848219 0.7157072  
## Size 0.6848219 1.0000000 0.7668722  
## Lot 0.7157072 0.7668722 1.0000000

**Finding Correlation in R** Watch out – variables must be numeric! May need to choose numeric columns:

data(Cereal)  
head(Cereal)

## Cereal Calories Sugar Fiber  
## 1 Common Sense Oat Bran 100 6 3  
## 2 Product 19 100 3 1  
## 3 All Bran Xtra Fiber 50 0 14  
## 4 Just Right 140 9 2  
## 5 Original Oat Bran 70 5 10  
## 6 Heartwise 90 5 6

#cor(Cereal) <- this doesn't work because it has character vectors in it   
  
cor(Cereal[c(2:4)]) # Tells R Which columns we want to take from the thing

## Calories Sugar Fiber  
## Calories 1.0000000 0.5154008 -0.7150123  
## Sugar 0.5154008 1.0000000 -0.5025772  
## Fiber -0.7150123 -0.5025772 1.0000000

**Test for a Linear Relationship via Correlation** - Let p (rho) denote the population correlation

Ho: p = 0 Ha: p =!= 0

t = ((r\*sqrt(n-2))/sqrt(1-r^2)) - Compare to a t-distribution with n-2 d.f.

**Correlation t-test in R** *SEe below:*

cor.test(Domestic$Distance, Domestic$Hours) # If null is true (if 0 correlation) how likely is it that we have a sample with this strong correlation? That's what this will tell you

##   
## Pearson's product-moment correlation  
##   
## data: Domestic$Distance and Domestic$Hours  
## t = 24.144, df = 52, p-value < 2.2e-16  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.9286787 0.9756277  
## sample estimates:  
## cor   
## 0.9581758

# OUtputs: Correlation bt the two things   
# What the cof int is for the population based here   
# t = 24.144 =how many stand dev from the null we are (Same value as before for test for slope)  
# Tells us really low pvalue, which means that its ar really low chance we could get this result by chance   
  
# All the results are going to be the same if it's simple linear regression   
# Gives us F test stat and p value

**ANOVA for Regression** Data = Model + ERror Total variation in response, Y = variation explained by MODEL + Unexplained variation in RESIDUALS

Key question: Does the MODEL explain a “significant” amount of the TOTAL variability?

**Partitioning Variability - SLM** Y = Bo + B1X + E SSTotal = SSModel +SSE

**NOtes** - ANOVA = analaysis of variances - we cant explain all variability, but we can try -*Total variations in reponse:* Is how far away each of our points are from the mean value - *Variation explained by model* total variability = full distance from teh mean down to the point; by fitting teh line to it, most of the variability is explained by the variation ; - *Resisudals* Distance that is left over from teh points - WE want a sig amount of variability in the model to be explained by the regression line vs a horizontal line - We really only care about teh sum of squares - The f test stat = how likely this variability is explained by model compared to what is left over in the error term if the null model were true that there is no relationship between tehse things **Can look at this by using ANOVA**

anova(moddist)

## Analysis of Variance Table  
##   
## Response: Distance  
## Df Sum Sq Mean Sq F value Pr(>F)   
## Hours 1 194417 194417 582.93 < 2.2e-16 \*\*\*  
## Residuals 52 17343 334   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# We have teh SSModel and SSError terms; does not give SSTotal, but if we add teh two, we could get that   
# What it is is that the SS in the HOurs row, is for the model, so if we had the line going back; how far away for each point the line is for that value away from teh mean of that value; how much variability is explained by the model and all squared and summed together   
# Residual SS; looks at the similar thing, but how far each point is from the line squared and summing them totgether   
  
# Think about that proportionally; how different are these two? Scale them by df.  
# MSE = comparing models for the vriabiilty and i'm taking the SSTotal/df. Hours df = 1, and residual df = 52 (which is 54-2)  
# F test stat = MSModel/MSError; that value is big it says that a lot of variability is explained by the model; when that value is small, it's saying not so much varibility is being explained by the model   
  
# Big and small are relative, depends on sample size   
# F test stat says its unlikely we would get this result by chance

**ANOVA Test for Regression** - Basic idea: Find two estimators of 2 - Model: SSModel/1 = MSModel ERror = variance of error = SSE/n-2 = MSE - We want to compare the model and the error

t.s. = MSModel/MSE - COmpare to F-dist with I and n-2 d.f

* ANOVA Test for regression Ho: B1 = 0 (ybar) Ha: B1 != 0 (yhat)
* Same null and alternative as slope test; asusming there is no relationship bt predictor and response
* so the slope of the model is 0 and the mean = the model that we use
* Alternative = some nonzero slope better descrbes this model and how likely we would get this kind of model if the null were true

Source, d.f, Sum of Squares, Mean Swuare, F, Pvalue Model, 1, SSModel, SSModel/1, MSModel/MSE, F(1,n-2) REsidual, n-2, SSE, SSE/n-2, See above got F and PValue TOtal, n-1, SSTOtal - in R use 1-pf(Fstat,1,n-2)

**What is r2?** r2 = proportion of total variability in the response (Y) that is “explained” by the model. 𝑟^2=𝑆𝑆𝑀𝑜𝑑𝑒𝑙/𝑆𝑆𝑇𝑜𝑡𝑎𝑙=1−𝑆𝑆𝐸/𝑆𝑆𝑇𝑜𝑡𝑎𝑙 - The amount of variability explaiend by the model out of the total variability - If we look at a plot ; the total variability (how far away each point is away from teh mean line), then our error term; varibaility explained by model = mean down tot the line - high values of rsquares = most of the variability in teh response is explained by the predictors a - low values = oppsitie

**Visualizing r2 for a SLM** Basic Idea: How much “better” does the least squares line do than a “prediction” that doesn’t use X at all? - Using NO predictor: 𝑦 ̂=𝑦 ̄

* Least Squares Line: 𝑦 ̂=𝛽 ̂\_𝑜+𝛽 ̂\_1 𝑥
* the cor coof being squared
* if corr coef is always bt -1 and 1; r^2 has to be bt 0 and 1
* big difference; r-sqqare we can look at multipel predictors at once

**Why is it called r2?** - Def: The correlation, r, measures the strength of linear association between two quantitative variables. -1< r <1 to 0 < r2 < 1 - 0 = Explains no variability - 1 = Explains all variability **Simple Linear Regression - R**

summary(moddist)

##   
## Call:  
## lm(formula = Distance ~ Hours, data = Domestic)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -88.892 -4.680 2.172 7.082 26.141   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -11.063 4.056 -2.727 0.00868 \*\*   
## Hours 59.977 2.484 24.144 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 18.26 on 52 degrees of freedom  
## Multiple R-squared: 0.9181, Adjusted R-squared: 0.9165   
## F-statistic: 582.9 on 1 and 52 DF, p-value: < 2.2e-16

**Which test is best?** -**T-test for slope:** Ho: B1=0 Ha: B1!=0

𝑡=𝑏\_1/(𝑆𝐸\_(𝑏\_1 ) )

Compare to t n-2 - How well related is this predictor with this response after taking into account the whole rest of the model, all other predicotrs - Not vacume - look at all in one room

* **ANOVA for regression:** Ho: B1=0 Ha: B1!=0

𝐹=𝑀𝑆𝑀𝑜𝑑𝑒𝑙/𝑀𝑆𝐸

Compare to F1,n-2 - Bigger picutre - Have 1 predictor, but once we have more, it will let us see if there is any relation bt any of these predictors and repsonse here - one test to see if there is any relation anywhere - useful for more tests

* **T-test for correlation:** Ho: p =0 Ha: p !=0

𝑡=(𝑟√(𝑛−2))/√(1−𝑟^2 )

Compare to t n-2 (tn-2)2 = F1,n-2

* Focuse on a predictor ans response
* how are they related in a vaccume ingnoring everyhting else

**We have 3 different tests for when we get to multiple regression**

## STOR 455 Quiz 1 Solutions

LEGO is a type of building toy created and made by the Lego Group, a company in Denmark. “Lego Bricks” are colorful plastic building blocks that can be joined together easily and are the most popular building toy in the world. Since the 1950s, the Lego Group has released thousands of sets with a variety of themes, including space, robots, pirates, trains, Vikings, castle, dinosaurs, undersea exploration, and wild west. Over the years, Lego has licensed themes from numerous cartoon and film franchises and even some from video games. These include Batman, Indiana Jones, Pirates of the Caribbean, Harry Potter, Star Wars, and Minecraft. For this quiz you will focus on the relationship between the number of pieces in a LEGO set and the set’s selling price on Amazon.

library(readr)

lego = read\_csv('https://raw.githubusercontent.com/JA-McLean/STOR455/master/data/lego.csv')

**Quiz A**

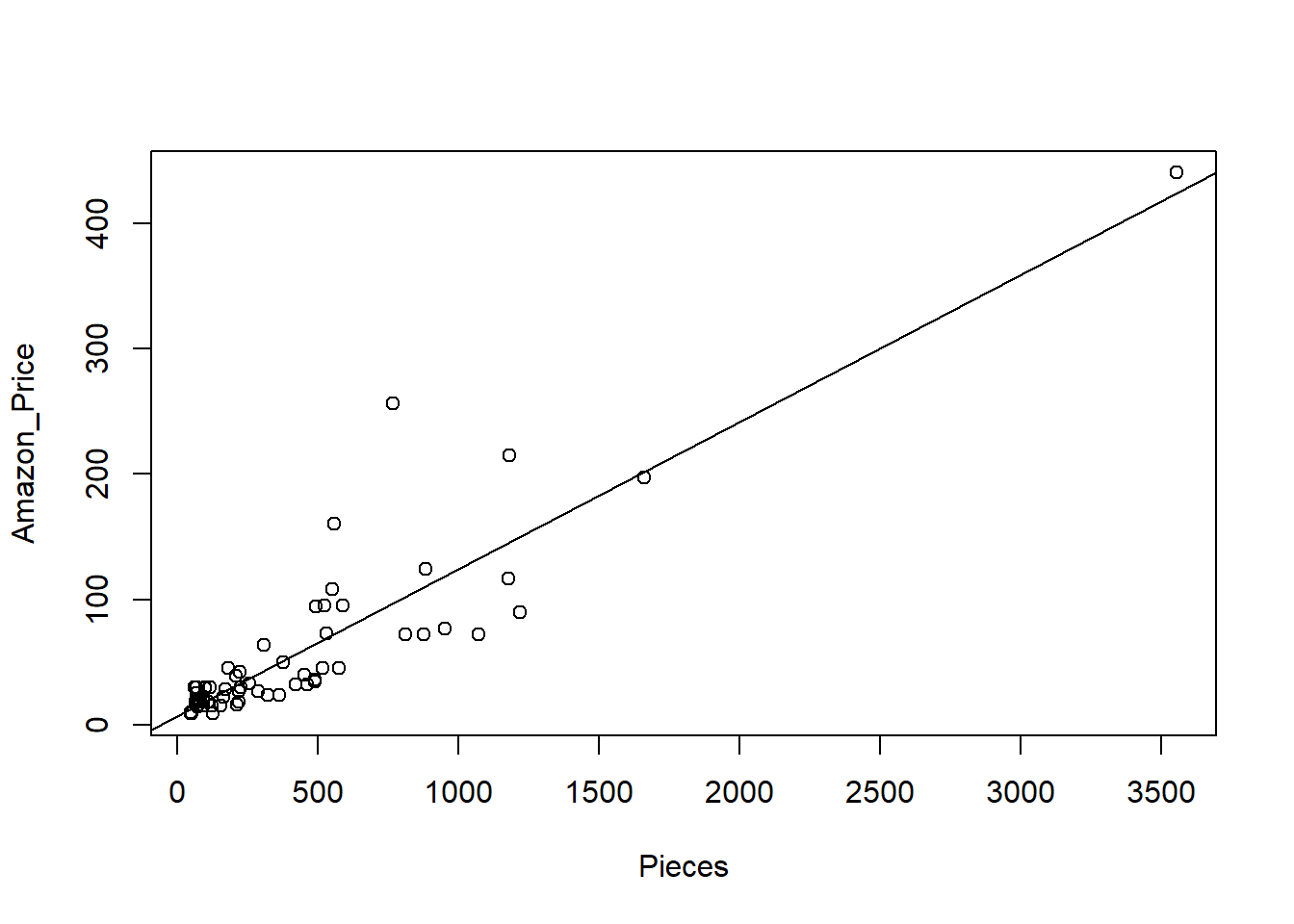
1. Construct a dataframe that contains LEGO sets that only have the *Theme* of *NINJAGO*. Use this dataframe to construct a linear model that predicts the *Amazon\_Price* based on the number of *Pieces* in the LEGO set. Plot the *Amazon\_Price* and *Pieces* data from this new dataframe, as well as your linear model, on one plot. You do not need to check the conditions for a linear model nor consider models with transformations. **5 pts**

ninjago = subset(lego, Theme == 'NINJAGO')

plot(Amazon\_Price ~ Pieces, data = ninjago)

ninjago\_mod = lm(Amazon\_Price ~ Pieces, data = ninjago)

abline(ninjago\_mod)



1. For each increase in 20 lego pieces in a *NINJAGO* LEGO set, how much does your model predict will be the increase in *Amazon\_Price*? **1.5 pts**

20 \* summary(ninjago\_mod)$coef[2,1]

## [1] 2.345026

1. Which NINJAGO LEGO set (by *Set\_Name*) has the largest influence on the regression model? How do you know this? Would this be considered an influential **1.5 pts**

max(cooks.distance(ninjago\_mod))

## [1] 0.3822737

which.max(cooks.distance(ninjago\_mod))

## 65

## 65

ninjago$Set\_Name[65]

## [1] "NINJAGO City Docks"

The NINJAGO City Docks set has the largest Cook’s Distance, therefore has the most influential on the model. Since this Cook’s Distance is less than 0.5, we would not consider this LEGO set to be influential.

1. What does your model predict, with 90% confidence, will be the *Amazon\_Price* for a *NINJAGO* LEGO set with 742 *Pieces*? Your answer should be an interval of values, and not one number. **2pts**

new\_set = data.frame(Pieces = 742)

predict.lm(ninjago\_mod, new\_set, interval="prediction", level=0.90)

## fit lwr upr

## 1 94.19487 40.97408 147.4157