Regression and RMSE

Oliver and JuJu

## Data Entry

The following code creates a sample data frame that we will use in our examples.

# Import the height and weight data  
 htwt = read.csv("http://facweb1.redlands.edu/fac/jim\_bentley/downloads/math111/htwt.csv")  
 # The Group variable is actually categorical and not numeric  
 htwt$Group = factor(htwt$Group,levels=c(1,2),labels=c("Male","Female"))  
 head(htwt)

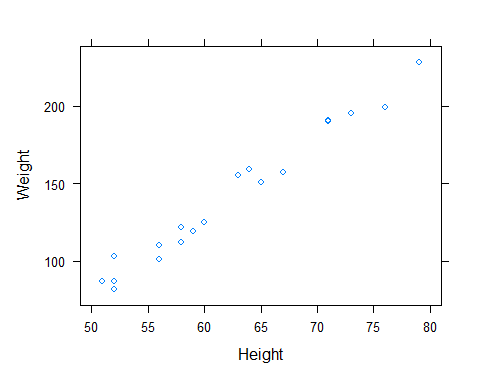
## Height Weight Group  
## 1 64 159 Male  
## 2 63 155 Female  
## 3 67 157 Female  
## 4 60 125 Male  
## 5 52 103 Female  
## 6 58 122 Female

## Esitmating Weight While Correcting for Height

It is fairly clear from graphing **Weight** as a function of **Height** that when modeling a person’s weight we should correct for height.

A linear model makes the correction fairly easy.

p\_load(lattice)  
 xyplot(Weight~Height, data=htwt)



To fit the model Weight = beta.0 + beta.1 **Height** + **error** we can use the ``simple’’ equations given in class or we can use the function ***lm***.

r = cor(htwt$Weight,htwt$Height)  
 b = r \* sqrt(var(htwt$Weight))/sqrt(var(htwt$Height))  
 b

## [1] 5.041217

a = mean(htwt$Weight) - b \* mean(htwt$Height)  
 a

## [1] -173.4596

htwt.lmw = lm(Weight ~ 1 + Height, data=htwt)  
 summary(htwt.lmw)

##   
## Call:  
## lm(formula = Weight ~ 1 + Height, data = htwt)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -10.6729 -5.4001 -0.6165 3.9014 14.3163   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -173.4596 12.2080 -14.21 3.18e-11 \*\*\*  
## Height 5.0412 0.1949 25.87 1.09e-15 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 7.171 on 18 degrees of freedom  
## Multiple R-squared: 0.9738, Adjusted R-squared: 0.9723   
## F-statistic: 669.1 on 1 and 18 DF, p-value: 1.09e-15

print(c(a, b))

## [1] -173.459595 5.041217

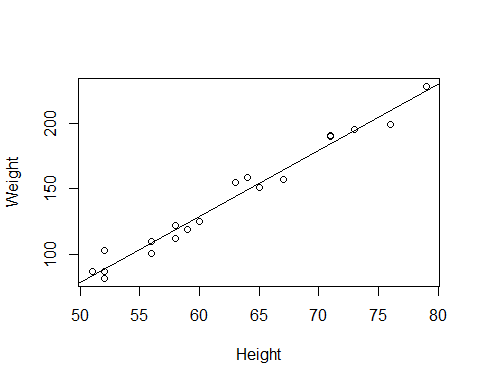
coef(htwt.lmw)

## (Intercept) Height   
## -173.459595 5.041217

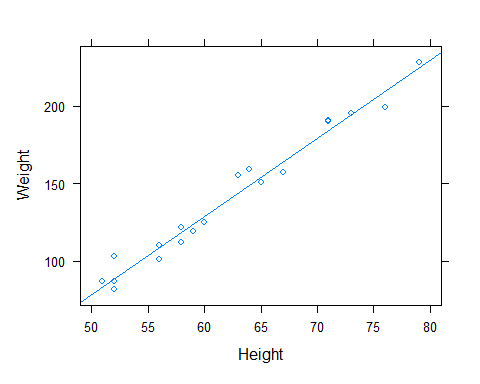
The estimated reqression line is seen to be **Weight** = -173.4595952 + 5.0412173 **Height**.

The regression line can be plotted using standard R functions.

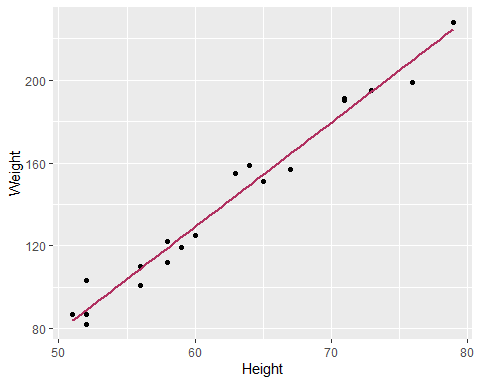
# Using base R graphics  
 plot(htwt$Height, htwt$Weight, xlab="Height", ylab="Weight")  
 abline(htwt.lmw)



# Using lattice graphics  
 p\_load(lattice)  
 xyplot(Weight ~ Height, data=htwt, type=c("p","r"))



# Using ggplot2  
 p\_load(ggplot2)  
 ggplot(data = htwt, aes(x = Height, y = Weight)) + geom\_point() + geom\_smooth(method = "lm", se=FALSE, color="maroon", formula = y ~ 1 + x)



## Root Mean Squared Error (RMSE)

Albert likes to use the root mean squared error (RMSE) as a measure of the quality of the predictor. The model with the smaller RMSE has less error and therefore is a “better” predictor.

We need to find the predicted and residual values for our data. Remember that the residual is **residual = observed - predicted**.

# Using definitions  
 obsserved = htwt$Weight  
 predicted = a + b \* htwt$Height  
 htwt.lmw.resid = obsserved - predicted  
 htwt.lmw.resid

## [1] 9.8216871 10.8629044 -7.3019648 -4.0134436 14.3162949 3.0689910  
## [7] -7.8485744 -6.6837051 3.2034274 -10.6729207 0.4507313 1.1514256  
## [13] 6.5331659 -3.2195302 -4.9722263 -4.9722263 -6.9310090 3.3575122  
## [19] 5.5331659 -1.6837051

# Using internal functions  
 as.numeric(residuals(htwt.lmw))

## [1] 9.8216871 10.8629044 -7.3019648 -4.0134436 14.3162949 3.0689910  
## [7] -7.8485744 -6.6837051 3.2034274 -10.6729207 0.4507313 1.1514256  
## [13] 6.5331659 -3.2195302 -4.9722263 -4.9722263 -6.9310090 3.3575122  
## [19] 5.5331659 -1.6837051

The RMSE is literally the square root of the average of the squared errors.

# Get the residuals  
 res = residuals(htwt.lmw)  
 # Square them  
 res2 = res^2  
 # Create one of Albert's tables  
 tbl = cbind(height=htwt$Height, obs=htwt$Weight, pred=predict(htwt.lmw), res, res2)  
 round(tbl, 4)

## height obs pred res res2  
## 1 64 159 149.1783 9.8217 96.4655  
## 2 63 155 144.1371 10.8629 118.0027  
## 3 67 157 164.3020 -7.3020 53.3187  
## 4 60 125 129.0134 -4.0134 16.1077  
## 5 52 103 88.6837 14.3163 204.9563  
## 6 58 122 118.9310 3.0690 9.4187  
## 7 56 101 108.8486 -7.8486 61.6001  
## 8 52 82 88.6837 -6.6837 44.6719  
## 9 79 228 224.7966 3.2034 10.2619  
## 10 76 199 209.6729 -10.6729 113.9112  
## 11 73 195 194.5493 0.4507 0.2032  
## 12 56 110 108.8486 1.1514 1.3258  
## 13 71 191 184.4668 6.5332 42.6823  
## 14 65 151 154.2195 -3.2195 10.3654  
## 15 59 119 123.9722 -4.9722 24.7230  
## 16 59 119 123.9722 -4.9722 24.7230  
## 17 58 112 118.9310 -6.9310 48.0389  
## 18 51 87 83.6425 3.3575 11.2729  
## 19 71 190 184.4668 5.5332 30.6159  
## 20 52 87 88.6837 -1.6837 2.8349

# Compute the column (second dimension) sums  
 tblsums = apply(tbl,2,sum)  
 round(tblsums, 4)

## height obs pred res res2   
## 1242.0000 2792.0000 2792.0000 0.0000 925.5001

# Divide by the number of observations to get column means  
 tblmeans = tblsums/20  
 round(tblmeans, 4)

## height obs pred res res2   
## 62.100 139.600 139.600 0.000 46.275

# Now take the square root of the mean of the squared resids to get the RMSE  
 sqrt(tblmeans[5])

## res2   
## 6.802573

# Or, create and use a function rmse  
 rmse = function(lmmodel) {sqrt(mean(residuals(lmmodel)^2))}  
 rmse(htwt.lmw)

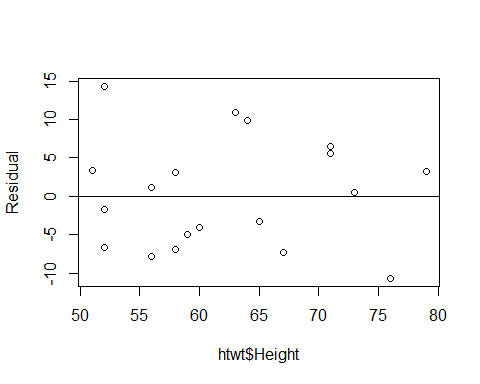
## [1] 6.802573

The RMSE of the model of ***Weight*** based on ***Height*** is 6.8026. If we had another model of ***Weight*** based on a different variable, we could compare the two using RMSE. The model with a smaller RMSE would be preferred as it reduces the size of the errors.

## Residual Analysis

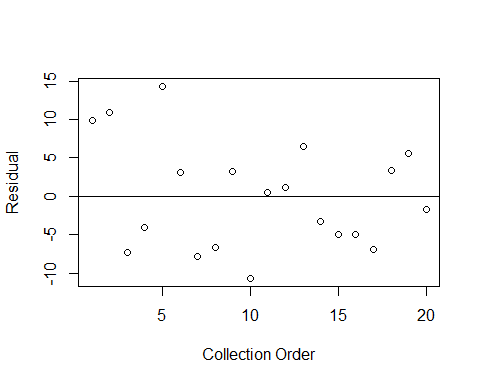
When fitting a regression model, it is always a good idea to check the residuals to make certain that underlying model assumptions are satisfied. A first step is to plot the residuals against the explanatory variables.

plot(htwt$Height, res, ylab="Residual")  
 abline(h=0)



Since no obvious patterns exist in the residuals, we can be fairly sure that we have not missed some non-linear pattern in the data. Since the vertical spread of the residuals seems to be the same across the x values, we can assume that the variability is constant.

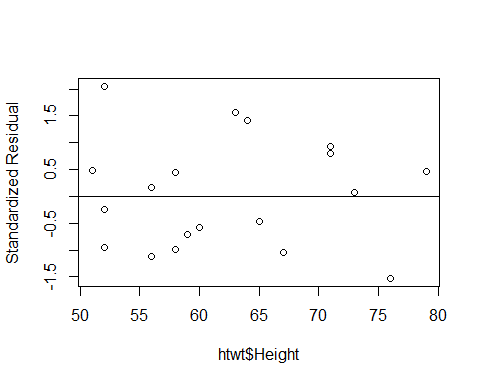
plot(1:length(res), res, ylab="Residual", xlab="Collection Order")  
 abline(h=0)



A lack of pattern in the residuals when plotted against the collection order indicates that we do not have a serial autocorrelation problem. That is, that there is not information in a previous observation about a subsequent observation.

Outliers can be detected using the z-scores for the residuals. These are known as the standardized residuals.

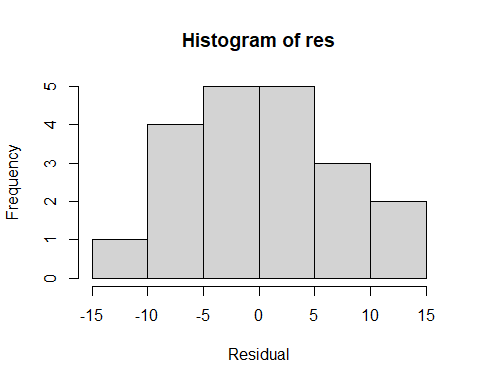
plot(htwt$Height, (res-mean(res))/sqrt(var(res)), ylab="Standardized Residual")  
 abline(h=0)



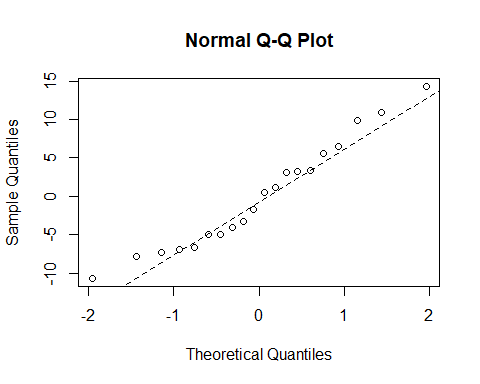
The residual corresponding to a height of 52 inches appears to be a little large. We should probably check this observation to see if it is an outlier.

If the data were collected using random selection and we wish to test to see if the slope is different from zero, the residuals must be normally distributed.

hist(res, xlab="Residual")



qqnorm(res)  
 qqline(res, lty=2)



Neither the histogram nor the normal quantile plot suggest any reason to question the normality of the residuals. Statistical testing appears to be valid in this situation.

R generates a number of plots automatically.

plot(htwt.lmw)

