Hmwk #5 - Regression Analysis

Problem #1 - Trees

Foresters want to measure the volume of a tree in order to estimate how much lumber they would get for that tree. Knowing the girth and height of the tree, one could apply a cylinder approximation, but the tree does not have a uniform girth along its entire length -- the tree gets much smaller at the top. The UsingR library has a data set "trees", that contains a set of measurements of the girth and height of a tree. Use the library to assess a linear model. For example, you might try:

In [5]:
```
library(UsingR)
summary(trees)
```

Out[5]:
```
     Girth           Height       Volume     
 Min.   : 8.30   Min.   :63   Min.   :10.20  
 1st Qu.:11.05   1st Qu.:72   1st Qu.:19.40  
 Median :12.90   Median :76   Median :24.20  
 Mean   :13.25   Mean   :76   Mean   :30.17  
 3rd Qu.:15.25   3rd Qu.:80   3rd Qu.:37.30  
 Max.   :20.60   Max.   :87   Max.   :77.00  
```

If we look at the regression model

In [6]:
```
m = lm(Volume ~ Height + Girth, data=trees)
summary(m)
```

Out[6]:
```
Call:
lm(formula = Volume ~ Height + Girth, data = trees)

Residuals:
    Min      1Q  Median      3Q     Max 
-6.4065 -2.6493 -0.2876  2.2003  8.4847 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -57.9877     8.6382  -6.713 2.75e-07 ***
Height        0.3393     0.1302   2.607   0.0145 *  
Girth         4.7082     0.2643  17.816  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.882 on 28 degrees of freedom
Multiple R-squared:  0.948,     Adjusted R-squared:  0.9442 
F-statistic:   255 on 2 and 28 DF,  p-value: < 2.2e-16
```

This provides a model, but doesn't say if it's any good. Do the following using R -- i.e. use the output of the 'lm' linear model where possible. Show your work.

What is the R^2 for this model

In [21]:
```
#summary(m)$r.squared
volume=trees$Volume
sse = sum((predict(m) - volume)^2)
sst = sum((volume-mean((volume)))^2)
ssr = sst-sse
r2=ssr/sst
r2
```

Out[21]: 0.947950037781675

Out[21]: 0.947950037781675

Compute the 95% confidence interval for the Height parameter

(**nb:** you're being asked to calculate the CI of the parameter from the multi-linear model, not the CI of the Height data itself). To do this, you would use the standard error (shown in the summary of the linear model -- e.g. 0.1302 for parameter Height) for each parameter -- this is the standard error for that parameter derived from the MSE as described in the slides concerning multi-linear regression. You would then conduct the T-distribution using that standard error with n-k-1 degrees of freedom. See the slides on linear regression.

In [112]:
```
confint(m,'Height',level=0.95)
#coef(summary(m))
```

Out[112]:

	2.5 %	97.5 %
Height	0.07264863	0.60585384

Compute the 95% confidence interval for the Girth parameter

(**nb:** as above). Verify your value using the **confint** function in R.

In [96]:
```
confint(m,'Girth',level=0.95)
```

Out[96]:

	2.5 %	97.5 %
Girth	4.166839	5.249482

Does the Height parameter statistically equal zero? Does the Girth?

In [22]:
```
print("No. At confidence interval 95%, both height and girth is non-zero")
```

```
[1] "No. At confidence interval 95%, both height and girth is non-zero"
```

Complete this sentence:

"For every unit increase in Girth, the Volume increases by __ and for every unit increase in Height, the Volume increases by __"

In [25]:
```
print("For every unit increase in Girth, the Volume increases by 4.7")
print("and for every unit increase in Height, the Volume increases by 0.
33")
```

```
[1] "For every unit increase in Girth, the Volume increases by 47%"
[1] "and for every unit increase in Height, the Volume increases by 3.
3%"
```

Problem #2 - Body Fat

In [28]:
```
summary(fat)
```

Out[28]:
```
      case           body.fat     body.fat.siri      density      
 Min.   :  1.00   Min.   : 0.00   Min.   : 0.00   Min.   :0.995  
 1st Qu.: 63.75   1st Qu.:12.80   1st Qu.:12.47   1st Qu.:1.041  
 Median :126.50   Median :19.00   Median :19.20   Median :1.055  
 Mean   :126.50   Mean   :18.94   Mean   :19.15   Mean   :1.056  
 3rd Qu.:189.25   3rd Qu.:24.60   3rd Qu.:25.30   3rd Qu.:1.070  
 Max.   :252.00   Max.   :45.10   Max.   :47.50   Max.   :1.109  
      age            weight          height           BMI        
 Min.   :22.00   Min.   :118.5   Min.   :29.50   Min.   :18.10  
 1st Qu.:35.75   1st Qu.:159.0   1st Qu.:68.25   1st Qu.:23.10  
 Median :43.00   Median :176.5   Median :70.00   Median :25.05  
 Mean   :44.88   Mean   :178.9   Mean   :70.15   Mean   :25.44  
 3rd Qu.:54.00   3rd Qu.:197.0   3rd Qu.:72.25   3rd Qu.:27.32  
 Max.   :81.00   Max.   :363.1   Max.   :77.75   Max.   :48.90  
    ffweight          neck           chest           abdomen      
 Min.   :105.9   Min.   :31.10   Min.   : 79.30   Min.   : 69.40  
 1st Qu.:131.3   1st Qu.:36.40   1st Qu.: 94.35   1st Qu.: 84.58  
 Median :141.6   Median :38.00   Median : 99.65   Median : 90.95  
 Mean   :143.7   Mean   :37.99   Mean   :100.82   Mean   : 92.56  
 3rd Qu.:153.9   3rd Qu.:39.42   3rd Qu.:105.38   3rd Qu.: 99.33  
 Max.   :240.5   Max.   :51.20   Max.   :136.20   Max.   :148.10  
      hip            thigh            knee            ankle          bic
ep
 Min.   : 85.0   Min.   :47.20   Min.   :33.00   Min.   :19.1   Min.
:24.80
 1st Qu.: 95.5   1st Qu.:56.00   1st Qu.:36.98   1st Qu.:22.0   1st Q
u.:30.20
 Median : 99.3   Median :59.00   Median :38.50   Median :22.8   Median
:32.05
 Mean   : 99.9   Mean   :59.41   Mean   :38.59   Mean   :23.1   Mean
:32.27
 3rd Qu.:103.5   3rd Qu.:62.35   3rd Qu.:39.92   3rd Qu.:24.0   3rd Q
u.:34.33
 Max.   :147.7   Max.   :87.30   Max.   :49.10   Max.   :33.9   Max.
:45.00
    forearm          wrist      
 Min.   :21.00   Min.   :15.80  
 1st Qu.:27.30   1st Qu.:17.60  
 Median :28.70   Median :18.30  
 Mean   :28.66   Mean   :18.23  
 3rd Qu.:30.00   3rd Qu.:18.80  
 Max.   :34.90   Max.   :21.40  
```

The Body Mass Index (BMI) is a model to predict the percentage of body fat based on your weight and height. The VMI is defined as the ratio of weight (in kilograms) to the square of height (in metres). A BMI of 18.5 to 25 is considered "healthy", a BMI of 25 to 30 is "overweight" and a BMI over 30 is "obese". The dataset 'fat' from UsingR contains 19 factors. The true body fat is body.fat and BMI field is the BMI.

Using the 'fat' data set, build a linear model

of the body.fat predicted by BMI. Describe the linear model -- what are the intercept and slope. What is the r^2 of that model?

In [48]:
```
n=lm(body.fat~BMI, data=fat)
#summary(n)
n$coefficients[1]
n$coefficients[2]
cat("R2: ")
cat(summary(n)$r.squared)
```

Out[48]: **(Intercept):** -20.4050815911524

Out[48]: **BMI:** 1.54671230729168

```
R2: 0.5299755
```

Come up with a minimal model that predicts body.fat using the other factors with an R2 of at least 0.72

We will use these different factors to come up with a minimal model that predicts body.fat using the other factors with an R2 of at least 0.72. Each factor in the final model should be significant at the 95% level -- this means that the confidence interval of the parameter at the 95% confidence level should not include zero.

Include your description showing both your final result and the process by which you achieve that result. Your model shouldn't use the "density" or "body.fat.siri" factors, as those are the "gold standard" measurements used to calculate body fat (using a dunk tank). I'm not certain what "ffweight" is, but don't use that either.

You may want to understand the "p-value" for problem - basically, the p-value is the probability of observing a value at least as large as the "t-value" (which is the estimate / std. error). Effectively, this is providing a way to determine if that value is statistically equal to zero.

You should describe the process you use, not just the end result. You should also insure that the linear model is valid, meaning that the predictors are not correlated, etc. Wikipedia has a good article on such "stepwise refinement" mechanisms [http://en.wikipedia.org/wiki/Stepwise_regression (http://en.wikipedia.org/wiki/Stepwise_regression)]. See also the discussion on page 125 of the book by Faraway on Practical Regression Analysis using R (on the couse web page or http://cran.r-project.org/doc/contrib/Faraway-PRA.pdf (http://cran.r-project.org/doc/contrib/Faraway-PRA.pdf)). They describe a "backward" and "forward" mechanism where you either start with all terms & toss some away or start adding terms to a null model.

In [65]:
```
o=lm(body.fat~hip+thigh+abdomen+weight*height, data=fat)
summary(o)$r.squared
print("Initially included all parameters which gave less R2 value. Then
removed some values and cross-product factors ")
print("like height and weight which gave high R2")
```

Out[65]:
```
0.731258230838149

[1] "Initially included all parameters which gave less R2 value. Then r
emoved some values and cross-product factors "
[1] "like height and weight which gave high R2"
```

Problem #3 - Wireless networks

This data contains measurements from 3 wireless network devices. One is an 802.11 "Wifi" interface running at 1Mbs/s. The second is that same interface running at 11Mb/s. Lastly, the same interface running at 54Mb/s. The data is stored as R vectors named x1, y1, x11, y11 and x54, y54. The X value is the packet size for the time measurement recorded at the corresponding Y value. The units are milliseconds. For example, x11[1] is 350 bytes and y11[1] is 1.436782 millseconds. It took 1.436782 milliseconds to transmit a 350 byte packet.

In [67]:
```
x11 = c(350, 350, 350, 350, 350, 450, 450, 450, 450, 450, 550, 550, 550,
550, 550, 650, 650, 650, 650, 650, 750, 750, 750, 750, 750, 850, 850,
850, 850, 850, 950, 950, 950, 950, 950, 1050, 1050, 1050, 1050, 1050,
1150, 1150, 1150, 1150, 1150, 1250, 1250, 1250, 1250, 1250, 1350,
1350, 1350, 1350, 1350, 1450, 1450, 1450, 1450, 1450 )

y11 = c(
1.436782, 1.407063, 1.436782, 1.426737, 1.416832, 1.50015, 1.533978,
1.50015, 1.522533, 1.511259, 1.619433, 1.619433, 1.576541, 1.576541,
1.587050, 1.682935, 1.662787, 1.693193, 1.682935, 1.693193, 1.745505,
1.755310, 1.765225, 1.755310, 1.755002, 1.833853, 1.805054, 1.824152,
1.853568, 1.853568, 1.915342, 1.915342, 1.915342, 1.904762, 1.915342,
1.992429, 1.982161, 1.992429, 1.961169, 1.971998, 2.072539, 2.052124,
2.062281, 2.062281, 2.082466, 2.15378, 2.143623, 2.133561, 2.113718,
2.103934, 2.182453, 2.182453, 2.182453, 2.154244, 2.182453, 2.241147,
2.26142, 2.241147, 2.301496, 2.282063)

x1 = c( 350, 350, 350, 350, 350, 450, 450, 450, 450, 450, 550, 550,
550, 550, 550, 650, 650, 650, 650, 650, 750, 750, 750, 750, 750, 850,
850, 850, 850, 850, 950, 950, 950, 950, 950, 1050, 1050, 1050, 1050,
1050, 1150, 1150, 1150, 1150, 1150, 1250, 1250, 1250, 1250, 1250,
1350, 1350, 1350, 1350, 1350, 1450, 1450, 1450, 1450, 1450)

y1 = c( 4.161465, 4.078303, 4.078303, 4.078303, 4.078303, 4.741584,
4.741584, 4.741584, 4.741584, 4.741584, 5.534034, 5.534034, 5.534034,
5.534034, 5.534034, 6.30517, 6.30517, 6.30517, 6.30517, 6.30517,
7.097232, 7.097232, 7.097232, 6.939625, 7.097232, 7.830854, 7.830854,
7.830854, 7.830854, 7.830854, 8.403361, 8.403361, 8.403361, 8.403361,
8.403361, 9.14913, 9.14913, 9.14913, 9.14913, 9.14913, 9.910803,
9.910803, 9.910803, 9.910803, 9.910803, 10.81081, 10.81081, 10.55966,
10.81081, 10.81081, 11.60093, 11.60093, 11.33787, 11.33787, 11.33787,
12.16545, 12.16545, 12.16545, 12.16545, 12.16545)

x54 = c( 350, 350, 350, 350, 450, 450, 450, 450, 550, 550, 550, 550,
650, 650, 650, 650, 750, 750, 750, 750, 850, 850, 850, 850, 950, 950,
950, 950, 1050, 1050, 1050, 1050, 1150, 1150, 1150, 1150, 1250, 1250,
1250, 1250, 1350, 1350, 1350, 1350, 1450, 1450, 1450, 1450)

y54 = c( 0.2812386, 0.2804341, 0.2798769, 0.2815553, 0.2995088,
0.2986679, 0.3006705, 0.298454, 0.311886, 0.3199386, 0.3163556,
0.3186439, 0.3333333, 0.3369953, 0.3340236, 0.3311258, 0.3412969,
0.3373933, 0.341006, 0.3411456, 0.3531198, 0.3563284, 0.3578714,
0.3521871, 0.3733154, 0.3752768, 0.3761803, 0.3780432, 0.3957888,
0.3914660, 0.3928656, 0.3961651, 0.4116921, 0.4088307, 0.4083966,
0.4038935, 0.4210704, 0.4251339, 0.4259488, 0.4252243, 0.4428698,
0.4405869, 0.4386157, 0.4402959, 0.4577287, 0.4561211, 0.4573938,
0.4607658)
```

Answer the following questions:

To answer these questions, you should use the lecture slides on linear regression and section 13 from the SimpleR.pdf book from the course website (page 77). You can also refer to the material by Boudec, but his write-up is, as per normal, complex.

- Using R as a calculator, but not using the built-in regression functions, calculate using the equations in the Linear Regression slides:
- The slope (b) and intercept (a) of the regression model for the 1Mb/s network (using x1, y1)
- The coefficient of determination for the model for the 1Mb/s network (using x1,y1)
- The standard deviation for slope & intercept and the 95% confidence interval for the 1Mb/s network (using x1,y1)
- The predicted time, standard deviation and 95% confidence interval for the predicted time to transmit a 40 byte Wifi packet for the 1Mb/s network (using x1,y1)
- The predicted time, standard deviation and 95% confidence interval to transmit a 750 byte packet for the 1Mb/s network (using x1,y1)

In [166]:

```
p=lm(formula=y1~x1)
coef=coef(p)
Sxx=sum((x1-mean(x1))^2)
Syy=sum((y1-mean(y1))^2)
Sxy=sum(((x1-mean(x1)))*((y1-mean(y1))))
cat("b value:")
#cat(coef[2])
cat(Sxy/Sxx)
cat("\na value:")
cat(mean(y1)-b*mean(x1))
#cat(coef[1])

sse = sum((predict(p) - y1)^2)
sst = sum((y1-mean((y1)))^2)
ssr = sst-sse
cat("\nr2: ")
cat(ssr/sst)
#summary(p)$r.squared


se=sqrt(sse/(length(x1)-2))
cat("\nstd dev for intercept:")
sa2=se*(sqrt(sum(x1^2)))/(length(x1)*sqrt(Sxx))
cat(sa2)

cat("\nstd dev for slope:")
sb2=se/sqrt(Sxx)
cat(sb2)

z_95=qnorm(1-0.05/2)
#z_95=qt(0.975,length(x1)-2)
cat("\nCI for b: ")
ci_a=c(b-z_95*sb2, b+z_95*sb2)
cat(ci_a)

cat("\nCI for a: ")
ci_b=c(a-z_95*sa2, a+z_95*sa2)
cat(ci_b)

predict(lm(y1~x1), data.frame(x1=40), interval="confidence", conf.level=
0.95)
predict(lm(y1~x1), data.frame(x1=750), interval="confidence", conf.level
=0.95)
model1=a+b*40
model2=a+b*750

sd1=sqrt((model1-mean(a+b*x1))^2)
sd2=sqrt((model2-mean(a+b*x1))^2)
cat("\n SD for both models: ")
cat(sd1,sd2)

ci_40l=model1-se*z_95*sqrt(1+1/length(x1)+(40-mean(x1))^2/Sxx)
ci_40u=model1+se*z_95*sqrt(1+1/length(x1)+(40-mean(x1))^2/Sxx)
```

```
cat("\n CI for 40 byte packet: ")
cat(ci_40l,ci_40u)

ci_750l=model1-se*z_95*sqrt(1+1/length(x1)+(750-mean(x1))^2/Sxx)
ci_750u=model1+se*z_95*sqrt(1+1/length(x1)+(750-mean(x1))^2/Sxx)
cat("\n CI for 750 byte packet: ")
cat(ci_750l,ci_750u)

#confint(p,'x1',level=0.95)
#coefficients(summary(p))['x1','Std. Error']
#es = resid(p)
#b1 =(coef(p))[['x1']]
#n=length(x1)
#s=sqrt( sum( es^2 ) / (n-2) )
#predict(p, data.frame=(x1=40))[1]
#predict(p, data.frame=(x1=750))[1]
#sd(predict(p, data.frame=(x1=40)))
#confint(p,'x1',level=0.95)
```

```
b value:0.007361049
a value:1.492115
r2: 0.9991551
std dev for intercept:0.003497775
std dev for slope:2.81074e-05
CI for b: 0.00730596 0.007416139
CI for a: 1.485259 1.49897
```

Out[166]:

	fit	lwr	upr
1	1.786557	1.734418	1.838695

Out[166]:

	fit	lwr	upr
1	7.012901	6.991725	7.034078

```
SD for both models: 6.330502 1.104157
CI for 40 byte packet: 1.630655 1.942459
CI for 750 byte packet: 1.637798 1.935315
```

Settle a law suit

Using the 'lm' functions in R or the simple.lm functions from the UsingR package (documented in the Simple R guide, section 13 page 77), answer the following questions

- Prepare a plot of the data, the regression model and the 95% confidence interval for each of the data sets. Label the plot with the parameters of the regression model. If you plot all the data on a single plot, you can put the parameters in the legend rather than the title
- Larry the Lawyer wants to sue because the 11Mb/s network is not 11 times "faster" than the 1Mb/s network. Compare the slope (time per byte) and intercept (overhead per packet). Is the 11Mb/s network 11 times faster at a 95% confidence level? Is the overhead different?
- Using your models, predict the time to transmit a 40 byte packet using each network (using a 95% confidence interval). What's Larry doing now (i.e. crying or filiing a suit?)
- Repeat that for a 750 and 1500 byte packet. Does Larry still have a case? What if you compare the time to transmit a 40 byte packet to a 1500 byte packet, which is 37 times bigger?
- For the 54Mb/s data, argue that the regression model is or is not appropriate for the data. Use the full range of techniques described in Jain and in class. Are there specific measurement samples which seem to be more problematic than others? Which ones?

In [151]:
```
par(mfrow=c(2,2))
q1=simple.lm(x1,y1,show.ci=TRUE, conf.level=0.95)
title(sub="1 Mbps", xlab="x1",ylab="y1")

q2=simple.lm(x11,y11,show.ci=TRUE, conf.level=0.95)
title(sub="11 Mbps", xlab="x11",ylab="y11")

q3=simple.lm(x54,y54,show.ci=TRUE, conf.level=0.95)
title(sub="54 Mbps", xlab="x54",ylab="y54")

confint(q1)
confint(q2)
confint(q3)
print("Intercept of x11 is not 4 times intercept of x1 network, so larry
has the case ")

pre <- function(x,y,val) {
  predict(lm(y~x), data.frame(x=val), interval="confidence", conf.level=
0.95)
}

pre(x1,y1,40)
pre(x11,y11,40)
pre(x54,y54,40)

pre(x1,y1,750)
pre(x11,y11,750)
pre(x54,y54,750)

pre(x1,y1,1500)/pre(x1,y1,40)
pre(x11,y11,1500)/pre(x11,y11,40)
pre(x54,y54,1500)/pre(x54,y54,40)

print("Larry has the case since since ratio is not equal to 37")

cat("\nRegression model is not sufficient for this case as time to trans
fer a packet cannot be linearly compared with")
cat("size of packet. Other factors like propogation delay of network aff
ects travel time")
```

Out[151]:

	2.5 %	97.5 %
(Intercept)	1.437881	1.546349
x	0.007304786	0.007417312

Out[151]:

	2.5 %	97.5 %
(Intercept)	1.166855	1.195275
x	0.0007413121	0.0007707959

Out[151]:

	2.5 %	97.5 %
(Intercept)	0.2234496	0.2293243
x	0.0001552897	0.0001613841

```
[1] "Intercept of x11 is not 4 times intercept of x1 network, so larry
has the case "
```

Out[151]:

	fit	lwr	upr
1	1.786557	1.734418	1.838695

Out[151]:

	fit	lwr	upr
1	1.211307	1.197646	1.224968

Out[151]:

	fit	lwr	upr
1	0.2327204	0.2298965	0.2355443

Out[151]:

	fit	lwr	upr
1	7.012901	6.991725	7.034078

Out[151]:

	fit	lwr	upr
1	1.748105	1.742557	1.753654

Out[151]:

	fit	lwr	upr
1	0.3451396	0.3439926	0.3462865

Out[151]:

	fit	lwr	upr
1	7.015556	7.203998	6.837802

Out[151]:

	fit	lwr	upr
1	1.911279	1.924560	1.898295

Out[151]:

	fit	lwr	upr
1	1.993346	2.008655	1.978404

```
[1] "Larry has the case since since ratio is not equal to 37"
```

Regression model is not sufficient for this case as time to transfer a packet cannot be linearly compared withsize of packet. Other factors li ke propogation delay of network affects travel time