MATH 1052H - S62 - Lecture 10

One further thing, simple linear regression

We did confidence intervals for β_0 and β_1, which is enough to let us perform hypothesis tests. We could also compute the **test** *t*-statistic, and use the *p*-value from the computer output, or from the *t*-statistic. Any of the methods we reviewed already will work just fine!

Formulae

For a hypothesis of

$$H_0 : \beta_1 = \beta \qquad\qquad H_A : \beta_1 \neq \beta$$

we compute

$$t_{\text{test}} = \frac{b_1 - \beta}{se(\beta_1)} = \frac{b_1 - \beta}{\frac{s_e}{\sqrt{\sum x^2 - \frac{(\sum x)^2}{n}}}}$$

where, once again,

$$s_e = \sqrt{\frac{\sum y^2 - b_0 \sum y - b_1 \sum xy}{n - 2}}.$$

Usual Test Steps

Once you've computed your t_{test}, just do everything like normal, remembering that you only have $n-2$ degrees of freedom instead of the usual $n-1$.

… or just use the outputs from R, with p-values!

Multiple Regression Example 1: Regression in Heights

Mother	Father	Daughter
63	64	58.6
67	65	64.7
64	67	65.3
60	72	61.0
65	72	65.4

(total of 20)

Doing a Simple Regression

```
summary(lm(daughter ~ mother))
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  16.8436     8.2500   2.042   0.0561 .
mother        0.7363     0.1292   5.697 2.11e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.976 on 18 degrees of freedom
Multiple R-squared:  0.6433,    Adjusted R-squared:  0.6235
F-statistic: 32.46 on 1 and 18 DF,  p-value: 2.106e-05
```

Checking the Results

Lets compute the standard error of β_1 (the mother coefficient) to verify what we see.

$$\begin{aligned} se(\beta_1) &= \frac{\sqrt{\frac{\sum y^2 - b_0 \sum y - b_1 \sum xy}{n-2}}}{\sqrt{\sum x^2 - \frac{\left(\sum x\right)^2}{n}}} \\ &= \frac{\sqrt{\frac{81{,}554.74 - 16.8436 \cdot 1275.6 - 0.7363 \cdot 81{,}491.6}{20-2}}}{\sqrt{81,515 - \frac{(1275)^2}{20}}} \\ &= \frac{1.975717}{15.28888} = 0.1292258 \end{aligned}$$

This is exactly what we saw from the computer:

```
            Estimate Std. Error t value Pr(>|t|)
mother        0.7363     0.1292   5.697 2.11e-05 ***
```

Then our t_{test} statistic is:

$$t_{\text{test}} = \frac{0.7363 - 0}{0.1292} = 5.698916 \approx 5.697.$$

So the output from the computer is exactly this, minus the work!

Multiple Regression: Computer Output

```r
summary(lm(daughter ~ father + mother))
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   7.4543    10.8804   0.685    0.503
father        0.1636     0.1266   1.293    0.213
mother        0.7072     0.1289   5.488    4e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.94 on 17 degrees of freedom
Multiple R-squared:  0.6752,    Adjusted R-squared:  0.637
F-statistic: 17.67 on 2 and 17 DF,  p-value: 7.057e-05
```

Pulling out critical things

We see a number of things. The first is the regression equation:

$$\text{daughter} = 7.4543 + 0.1636 \cdot \text{father} + 0.7072 \cdot \text{mother},$$

while the second is the R^2 value, $R^2 = 67.5\%$. We also see that the **adjusted** R^2 is listed as 63.7%, and that the $p-$value (overall) is 7.06×10^{-5}, a small number.

Definitions

- b_0 in a multiple regression problem is the **predicted** value of y when all predictor variables are zero. This doesn't always make sense!
- b_1 represents the change in y when the first variable (x_1) changes by one unit; everything else stays fixed
- b_2 represents the change in y when the second variable (x_2) changes by one unit; everything else stays fixed
- and so on!

Overall p-value

The overall p-value of the model ($7.057e-05 = 0.00007057$) is a measure of the **overall significance** of the multiple regression equation. The test it is associated with is:

$$H_0 : \text{all parameters are zero} \qquad \text{versus}$$
$$H_A : \text{at least one parameter is non-zero.}$$

(this is actually an ANOVA - it's all connected)

Interpreting the p-value

A p-value which is less than α means “reject the null”: this means at least one parameter is non-zero, and that the model can be used for prediction. There is **something useful** in the model.

A p-value which is greater than α means “fail to reject the null”: we do not have evidence to conclude that any of the parameters are non-zero. This means the model **should not** be used for prediction: there’s nothing useful in the model!

The Height Model

Our overall p-value was quite small: 0.00007057. This is much smaller than $\alpha = 0.01$ or $\alpha = 0.05$, so we see that there **is** useful information in this model, and we can use it to predict the heights of daughters from the heights of their parents.

Definition

The **adjusted** coefficient of determination is a variant of the coefficient of determination, R^2, that we talked about previously. The formula is

$$R^2_{adj} = 1 - \frac{n-1}{n-(k+1)}(1-R^2)$$

where n is the number of data points and k is the number of predictor (x) variables.

Why do we do this?

- in multiple regression, R^2 increases for every variable you add, even if it’s useless
- R^2 is **supposed** to be a measure of how well multiple regression fits the data
- So we adjust the R^2 so the newly added predictors need to be **useful**

Height Results

The results from the daughter/mother/father height regression are:

- $R^2 = 0.6752$
- Adjusted $R^2 = 0.637$

So the naive viewpoint says that 67.5% of the variation in daughter heights is explained by the heights of the father and mother. The adjusted (more sensible) viewpoint says that only 63.7% of the variation is explained.

When you **compare regression models**, you should use the adjusted R^2 value from each.

Variable Selection

The trickiest part of multiple regression is not interpreting the results of a model, or even comparing two models. Since we use computers, it’s also not the actual computation of the coefficients.

The trickiest part: deciding what variables go into the model in the first place!

The topic of **variable selection** can be very complicated, and we won’t cover all of the ways that exist to evaluate variables (many of which are too complex or mathematical for this course!).

Method 1: Common Sense

When trying to setup the model, and choosing the variables, you need some minimal amount of rationale for including a variable. We wouldn’t try to use the daughter’s favourite colour, for example: what would that have to do with her health?

We could, though, use the family’s ethnicity, socioeconomic status or geographic location as predictors, because there are **plausible** links between these things and someone’s height. Remember our Netherlands example: some countries have people who are taller than others! Genetics is a real thing.

Method 2: Overall p-value

We can use the overall model p-value of the resulting model fit(s), and compare them. The lower the p-value, the more “significant” the model fit is.

Method 3: Adjusted R^2

We can compare models by looking at their Adjusted R^2, and assuming all input variables are sensible (Method 1), higher Adjusted R^2 should imply a better fit / more predictive power.

Method 4: Standard Error

We can examine the standard error, s_e, for each of the models, and attempt to find one which has the smallest standard error. This leads to models which have more precise prediction intervals, so they do a good job at prediction $\hat{y}$ for a given x_0.

Method 5: Significant Variable p-values

Rather than looking at the overall model p-value, we could instead examine the specific coefficients in a given model, and look for ones which are not significant. We can then streamline our model, eliminating pieces which aren’t significant, until we find a **parsimonious** model.

Definition

The classical “science” definition of **parsimony** is:

```
The scientific principle that things are usually connected or behave
in the simplest or most economical way, especially with reference
to alternative evolutionary pathways.
```

In statistics, we use **parsimony** to mean:

```
The simplest or most economical model which explains the data well.
```

So a **parsimonious model** is one which has all the terms significant (possibly excepting the intercept / b_0 term), but which still has maximized Adjusted R^2 (or minimized overall p-value; or minimum s_e) for the given set of predictors.

A Tiny Bit of Combinatorics

Combinatorics is an area of mathematics that deals with **counting things**. You might have been introduced to a tiny bit of this area in high school if you ever discussed the “choose notation”:

$$\binom{y}{x}$$

which we read as “y choose x”.

What this lets us do is say “how many different ways can be pick x items out from a set of size y”.

Example: how many different 5-card hands are there from a standard 52-card deck?

$$\binom{52}{5} = 2,598,960.$$

Regression Example: how many different regression models are there if we have 4 possible predictors which we can use?

$$\binom{4}{1} + \binom{4}{2} + \binom{4}{3} + \binom{4}{4} = 15.$$

Slightly Easier

If we just say “well, each variable can be included, or not included”, then we have four possible decisions to make: variable 1, variable 2, etc. Each decision has two outcomes: we either include or don’t include. That becomes:

$$2 \times 2 \times 2 \times 2 = 16$$

different variations. But among those variations is one which has us rejecting all four variables … in that case, are we still doing regression?? So we have $16 - 1 = 15$ different possible models.

How to Build Models

So, we have 15 different variations. How can we possibly choose among them? Logically:

- First, make sure all the variables are actually correlated with the thing you want to predict. If something isn't correlated, throw it out immediately. (a bunch of r values)
- Second, check to see if the variables are correlated with **each other**. If two variables are highly correlated, then they will predict the same thing, so we only include the one that is most correlated with the response (throw one out).
- Third, build a complete model: include everything that's correlated and doesn't duplicate.

Once we've done this, we perform **backward elimination**: we start from this full model, and we start deleting things that aren't as useful until we reach parsimony.

One more note on t-tests

When we were doing simple linear regression earlier, we used $n - 2$ degrees of freedom for the critical values for the confidence intervals and comparison tests for our coefficient β_1. When we move to multiple regression, this isn't true anymore!

For a **multiple regression** model, the degrees-of-freedom of a t distribution for testing individual coefficients is $n - (k + 1)$, where n is the total number of observations, and k is the number of predictors x_i being used.

(so for a simple model, we have one prediction x_1, so $n - (1 + 1) = n - 2$).

Back to Height

Our model was

$$\text{daughter} = 7.4543 + 0.1636 \cdot \text{father} + 0.7072 \cdot \text{mother},$$

and the output from the model was

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   7.4543    10.8804   0.685    0.503
father        0.1636     0.1266   1.293    0.213
mother        0.7072     0.1289   5.488    4e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.94 on 17 degrees of freedom
Multiple R-squared:  0.6752,    Adjusted R-squared:  0.637
F-statistic: 17.67 on 2 and 17 DF,  p-value: 7.057e-05
```

In this case, we used all available predictors: mother **and** father. We see that the overall model is a good one (p-value of 0.00007057). We could use this model for prediction. The Adjusted R^2 is 0.637.

However, scanning the list of coefficients, we see that the **father** variable has a non-significant p-value. We might be able to eliminate the father variable, by the principle of parsimony, and make a simpler, but equally effective, model.

Simpler Model

```
summary(lm(daughter ~ mother))
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  16.8436     8.2500   2.042   0.0561 .
mother        0.7363     0.1292   5.697 2.11e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.976 on 18 degrees of freedom
Multiple R-squared:  0.6433,    Adjusted R-squared:  0.6235
F-statistic: 32.46 on 1 and 18 DF,  p-value: 2.106e-05
```

Results

- Adjusted R^2 went from 0.637 to 0.6235 (decrease)
- Overall p-value went from $7e-5$ to $2e-5$ (decrease)
- Residual standard error went from 1.94 to 1.976 (increase)

Conclusion? Adjusted R^2 decreased and residual standard error increased. While the significance of the simpler model is better (since p-value decreased), the other two metrics are **poorer** for the simpler model. Thus, even though removing the **father** variable from the model simplified things, it did **not** improve our predictive power, or our fit.

So we leave **father** in, and our “best model” from this data is to use both father and mother to predict the height of the daughter.

The Stepwise Algorithm

- Run the model with all sensible variables included
- Look at the t statistic and p-value of each term included. Find the variable that has the largest p-value which is still greater than α. Eliminate this variable.
- Re-run the model with the variable eliminated.
- Repeat until all remaining p-values are less than α.

As we saw in the height example above, while we can do this, sometimes it is worth leaving a variable in the model even if it has a p-value greater than α, if it contributes both increased Adjusted R^2 **and** reduced standard error. It’s quite tricky!

Dummy Variables

So far, all of the predictor and response variables we have considered have been continuous in nature. However, we can easily come up with examples where we would want to use a **dichotomous variable** (also known as a binary variable, or a TRUE/FALSE variable) in our regression.

From the ANOVA unit, you know that we can have lots of variables of interest that are not numeric: yes/no, true/false, red/green! Not numeric, categorical!

To use these variables (which aren't numbers!) in a **numerical** regression problem, we convert them to something known as **dummy variables**. Assign each of the cases to a numerical value 0 or 1; it doesn't matter which is which, so long as you are consistent. Then include the 0/1 variable as part of the regression.

Height Data: More Complicated

Let's go back in time to 1888, and look at Galton's **original** height data.

```
galton <- read_table2("galton.csv")
```

```
## # A tibble: 6 x 6
##   Family Father Mother Gender Height  Kids
##   <chr>   <dbl>  <dbl> <chr>   <dbl> <dbl>
## 1 1        78.5   67   M        73.2     4
## 2 1        78.5   67   F        69.2     4
## 3 1        78.5   67   F        69       4
## 4 1        78.5   67   F        69       4
## 5 2        75.5   66.5 M        73.5     4
## 6 2        75.5   66.5 M        72.5     4
```

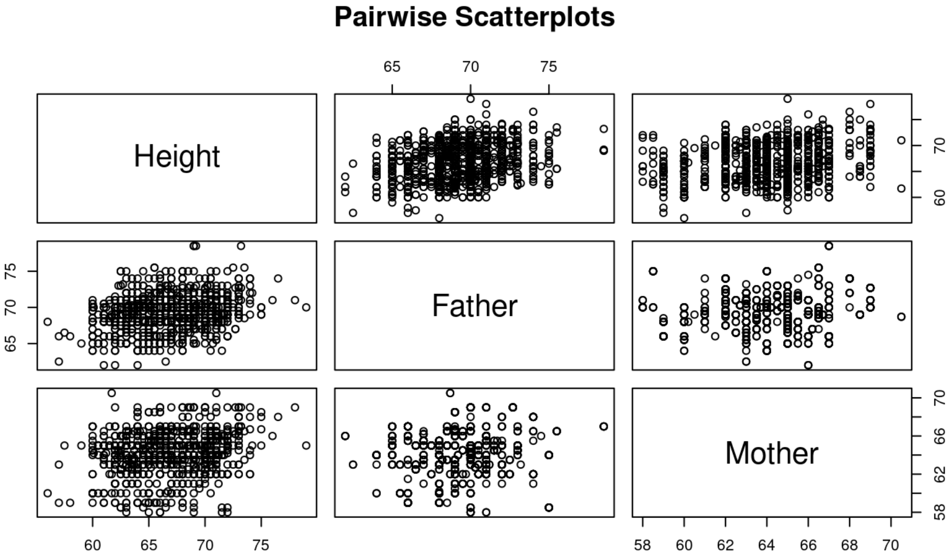
Scatterplots for Multiple Regression?

One problem with multiple regression is that the actual fit is not a **line** anymore: it's actually something known as a **hyperplane**: basically, a line in 3D or 4D. It's much, much harder to visualize!

One way to work around this for the initial screening part of the regression is to do scatterplots of each variable against **each other** variable, two at a time. So we do scatterplots of the response against each predictor, and **also** do scatterplots of the predictors *pair-wise* (done two at a time).

Galton’s Data, Scatterplots

```
pairs(Height ~ Father + Mother, data = galton,  main = "Pairwise Scatterplots")
```



So what do we see?

- positive linear relationships between all pairwise variables
- height of child is positively linearly correlated with height of father
- height of child is positively linearly correlated with height of mother
- height of father is positively linearly correlated with height of mother

Lets Compute

```
cor.test(galton$Height, galton$Father)
```

```
##
##  Pearson's product-moment correlation
##
## data:  galton$Height and galton$Father
## t = 8.5737, df = 896, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.2137851 0.3347455
## sample estimates:
##       cor
## 0.2753548
```

```r
cor.test(galton$Height, galton$Mother)
```

```
##
##  Pearson's product-moment correlation
##
## data:  galton$Height and galton$Mother
## t = 6.1628, df = 896, p-value = 1.079e-09
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.1380554 0.2635982
## sample estimates:
##       cor
## 0.2016549
```

```
cor.test(galton$Father, galton$Mother)
```

```
##
##  Pearson's product-moment correlation
##
## data:  galton$Father and galton$Mother
## t = 2.211, df = 896, p-value = 0.02729
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.008283733 0.138418345
## sample estimates:
##        cor
## 0.07366461
```

Same Model

We try the same model as before:

```
summary(lm(Height ~ Father + Mother, data = galton))
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 22.30971    4.30690   5.180 2.74e-07 ***
Father       0.37990    0.04589   8.278 4.52e-16 ***
Mother       0.28321    0.04914   5.764 1.13e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.386 on 895 degrees of freedom
Multiple R-squared:  0.1089,    Adjusted R-squared:  0.1069
F-statistic: 54.69 on 2 and 895 DF,  p-value: < 2.2e-16
```

Is this a good model?

- all individual coefficients are significant (very low p-values)
- overall model p-value is very small

There's no real path for backwards elimination here.

However: we did not include the final variable – gender!

New, Larger Model

```
summary(lm(Height ~ Father + Mother + as.factor(Gender), data = galton))
```

```
                   Estimate Std. Error t value Pr(>|t|)    
(Intercept)        15.34476    2.74696   5.586 3.08e-08 ***
Father              0.40598    0.02921  13.900  < 2e-16 ***
Mother              0.32150    0.03128  10.277  < 2e-16 ***
as.factor(Gender)M  5.22595    0.14401  36.289  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.154 on 894 degrees of freedom
Multiple R-squared:  0.6397,    Adjusted R-squared:  0.6385 
F-statistic:   529 on 3 and 894 DF,  p-value: < 2.2e-16
```

Even Better!

This reduces the standard error (good!), increases the Adjusted R^2 (good!), and even reduces all of the individual coefficient p-values (great!).

So this is our model: nothing to remove!

Another Example: Bears

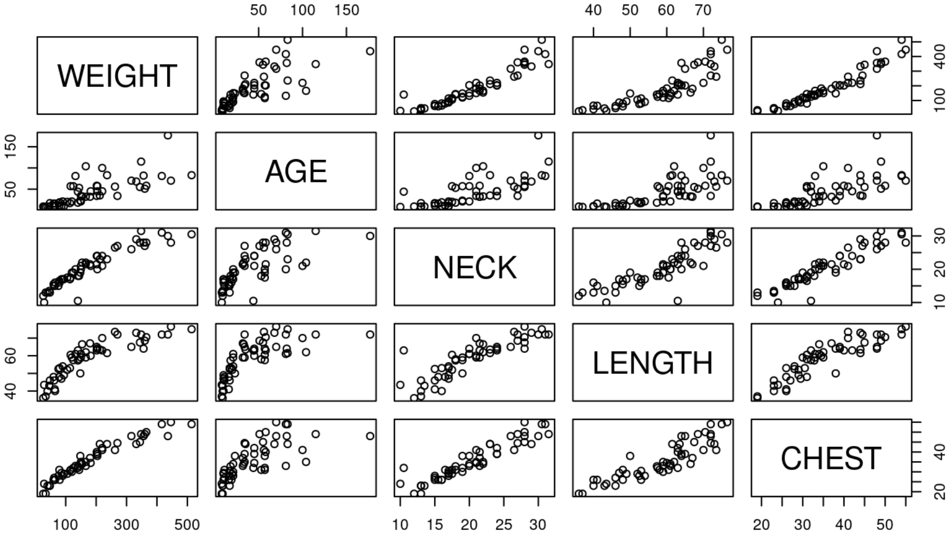
Consider a data set with variables about bears.

```
head(bears)
```

```
##   AGE MONTH SEX HEADLEN HEADWTH NECK LENGTH CHEST WEIGHT
## 1  19     7   1    11.0     5.5 16.0   53.0    26     80
## 2  55     7   1    16.5     9.0 28.0   67.5    45    344
## 3  81     9   1    15.5     8.0 31.0   72.0    54    416
## 4 115     7   1    17.0    10.0 31.5   72.0    49    348
## 5 104     8   2    15.5     6.5 22.0   62.0    35    166
## 6 100     4   2    13.0     7.0 21.0   70.0    41    220
```

Lets predict the weight of the bears by using the other variables, including Age. We will start by considering all the reasonable predictors: AGE, SEX, NECK, LENGTH, and CHEST.

```
pairs(WEIGHT ~ AGE + NECK + LENGTH + CHEST, data = bears)
```



Compute a Full Model

```
summary(lm(WEIGHT ~ as.factor(SEX) + AGE + NECK + LENGTH + CHEST, data = bears))
```

```
                 Estimate Std. Error t value Pr(>|t|)    
(Intercept)     -222.6963    31.0357  -7.175 3.96e-09 ***
as.factor(SEX)2  -12.0102    11.1807  -1.074   0.2881    
AGE                0.5224     0.2159   2.420   0.0194 *  
NECK               3.7588     2.5001   1.503   0.1393    
LENGTH            -0.4660     0.9033  -0.516   0.6083    
CHEST              9.4533     1.4259   6.630 2.72e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 30.4 on 48 degrees of freedom
Multiple R-squared:  0.9436,    Adjusted R-squared:  0.9377 
F-statistic: 160.6 on 5 and 48 DF,  p-value: < 2.2e-16
```

Eliminate

First, we eliminate variable LENGTH, which has the highest p-value:

```
summary(lm(WEIGHT ~ as.factor(SEX) + AGE + NECK + CHEST, data = bears))
```

```
                 Estimate Std. Error t value Pr(>|t|)    
(Intercept)     -233.4418    22.8373 -10.222 9.66e-14 ***
as.factor(SEX)2  -12.5375    11.0502  -1.135   0.2621    
AGE                0.5057     0.2118   2.387   0.0209 *  
NECK               3.5019     2.4315   1.440   0.1562    
CHEST              9.1623     1.2998   7.049 5.57e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 30.17 on 49 degrees of freedom
Multiple R-squared:  0.9433,    Adjusted R-squared:  0.9386 
F-statistic: 203.7 on 4 and 49 DF,  p-value: < 2.2e-16
```

Eliminate

Next, we eliminate the gender (SEX) of the bears:

```
summary(lm(WEIGHT ~ AGE + NECK + CHEST, data = bears))
```

```
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) -248.5247    18.6226 -13.345  < 2e-16 ***
AGE            0.3769     0.1794   2.101   0.0407 *  
NECK           4.8681     2.1185   2.298   0.0258 *  
CHEST          8.8311     1.2703   6.952 7.13e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 30.26 on 50 degrees of freedom
Multiple R-squared:  0.9418,    Adjusted R-squared:  0.9383 
F-statistic: 269.7 on 3 and 50 DF,  p-value: < 2.2e-16
```

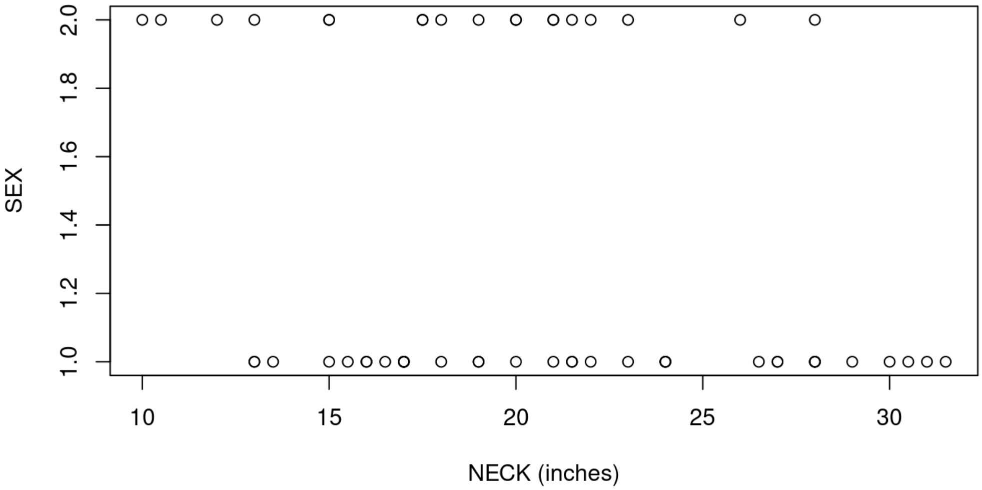
Conclusion

Our final model has Adjusted R^2 of 0.9383, with standard error 30.26 and significant p-values for every term in the regression. The previous model (which included gender) had Adjusted R^2 of 0.9386 and standard error of 30.17, but both NECK and SEX were not significant terms in the model.

This suggests to us that perhaps NECK and SEX were more correlated than average, and were interfering with each other.

Scatterplot

```
plot(bears$NECK, as.factor(bears$SEX), xlab = "NECK (inches)", ylab = "SEX")
```



Correlation

```
cor.test(bears$NECK, as.numeric(as.factor(bears$SEX)))
```

```
##
##  Pearson's product-moment correlation
##
## data:  bears$NECK and as.numeric(as.factor(bears$SEX))
## t = -2.1149, df = 52, p-value = 0.03925
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.51070292 -0.01477946
## sample estimates:
##        cor
## -0.2814263
```

Conclusion

The neck size of bears is correlated with their sex in a way which causes the two to interact when included in the model together. In general, it is always better to remove interacting terms unless you really know how they interact (remember: **domain-specific knowledge**).

Final Idea: Interaction

As we saw in the previous example, sometimes variables can interact in ways which interfere with the regression. One of the assumptions in regression (often violated!) is that all of the predictor variables are independent! When two variables are clearly related, as the SEX and NECK variables seemed to be, problems can happen.

If two variables are said to **interact**, or to have an **interaction**, one way of understanding what is going on is to assume that:

- there is a relationship between Variable 1 and the Response
- there is a relationship between Variable 2 and the Response
- the relationship between Variable 1 and the Response **changes** based on the value of Variable 2 (or vice versa)

We can include interactions in the model

```
summary(lm(WEIGHT ~ AGE + NECK + CHEST + as.factor(SEX) + as.factor(SEX) * NECK,
           data = bears))
```

```
                     Estimate Std. Error t value Pr(>|t|)    
(Intercept)         -258.3563    23.9078 -10.806 1.88e-14 ***
AGE                    0.4934     0.2015   2.449   0.0180 *  
NECK                   5.4896     2.4458   2.244   0.0294 *  
CHEST                  8.6797     1.2510   6.938 9.14e-09 ***
as.factor(SEX)2       67.5893    33.8473   1.997   0.0515 .  
NECK:as.factor(SEX)2   -4.0718     1.6350  -2.490   0.0163 *  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 28.68 on 48 degrees of freedom
Multiple R-squared:  0.9498,    Adjusted R-squared:  0.9445 
F-statistic: 181.5 on 5 and 48 DF,  p-value: < 2.2e-16
```

Even Better!

This is a better fit than what we had before. All but one term is significant at $\alpha = 0.05$, the standard error has been reduced, and the Adjusted R^2 has been increased. All because we realized that NECK and SEX were related, and added a term to the model which accounted for that **interaction**.

This concludes the example.

Non-linear Regressions

The final sections of Chapter 9 are quite light on material, and largely serve as an introduction to an advanced topic: what happens if the data is related, but not in a linear fashion?

This happens all the time in real world data!

Poisson Response Variables

One of the compound assumptions that is required for linear regression is that the relationship between predictor and response is **linear** (straight lines!) and also that the errors (and hence, residuals) are normally distributed.

However, one of the more common response variable types we want to use is not like this at all!

Definition: many forms of **count** data (that is, data which is recorded as the number of events occurring in a fixed amount of time) follow a **Poisson** random variable distribution. These count data will not meet the requirements of linear regression.

Examples: daily numbers of deaths in Toronto, Ontario; daily number of visitors to Disney World in Orlando, Florida; hourly number of customers at a local CIBC bank; hourly number of visitors to a Tim Horton’s drive-through.

Building a Model

We want the normality requirements of linear regression to be met. But our response variable has issues. How do we fix this?

Logarithmic Transformation: (log-transform for short) We can transform a variable in a regression model by applying a **transformation**; that is, by plugging the values into a function and using the results of that function output rather than the original values.

Reminder: the logarithm is a function family that you likely learned about in Grade 11. It is the functional inverse of the exponential function. The **natural logarithm** is one using *base* e: the number 2.71828, and is often written as $\ln(x)$ or $\log(x)$ if context is clear.

$$e^x \leftrightarrow \ln(x)$$

When to use the transform

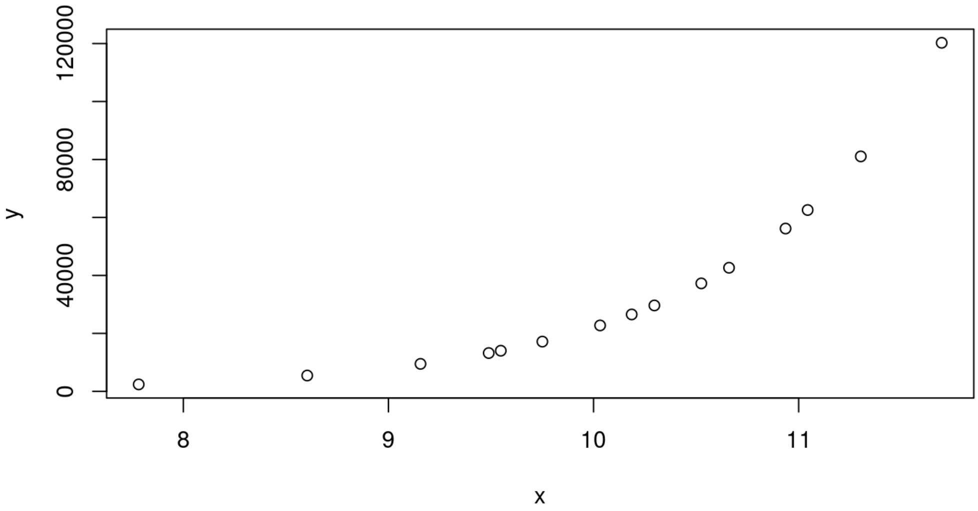
If

- the response variable is counts-per-time
- there is an exponential relationship visible on a scatterplot

then it may be appropriate to take the natural logarithm of the appropriate variable.

Example 1: Response

If our scatterplot looks like



then we appear to have an exponential relationship.

Naive Model

```
summary(mod1 <- lm(y ~ x))
```

```
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  -587157     123121  -4.769 0.000367 ***
x              63385      12119   5.230 0.000162 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 53840 on 13 degrees of freedom
Multiple R-squared:  0.6779,    Adjusted R-squared:  0.6531 
F-statistic: 27.36 on 1 and 13 DF,  p-value: 0.0001623
```

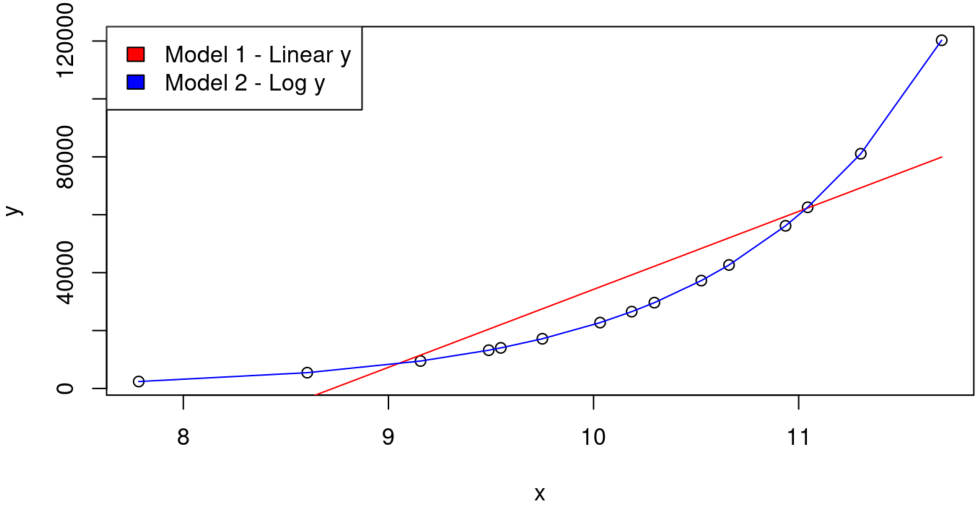
Improved Model

```
summary(mod2 <- lm(log(y) ~ x))
```

```
            Estimate  Std. Error   t value Pr(>|t|)
(Intercept) 9.861e-05  2.573e-04     0.383    0.708
x           1.000e+00  2.533e-05 39480.354   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.0001125 on 13 degrees of freedom
Multiple R-squared:      1, Adjusted R-squared:      1
F-statistic: 1.559e+09 on 1 and 13 DF,  p-value: <2.2e-16
```

Can we tell the difference?



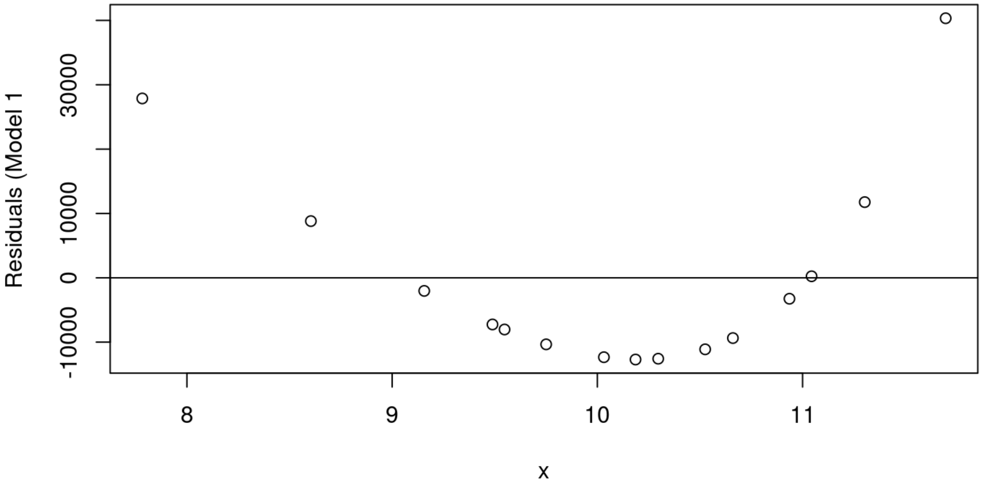
Always Plot!

If possible, we **always** want to plot the data. The two previous examples were both “good” regressions (from p-values, although not from s_e, which is the big hint), but clearly one of them is far superior to the other one!

Using Residuals to Detect Needed Transforms

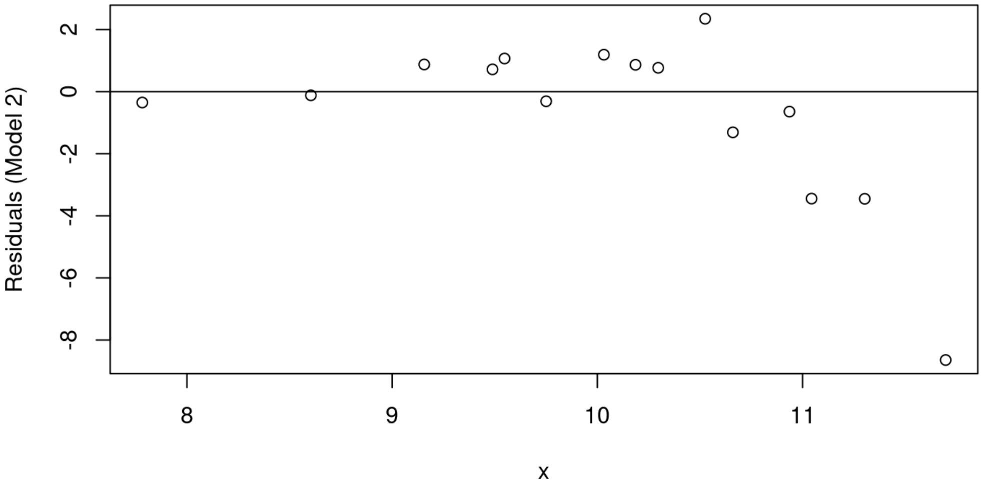
One of the best ways to find a needed transformation in our data is to look at the residuals. If linear is a “bad” choice for the data, the residuals will almost always have a pattern: this pattern can suggest an appropriate transformation.

Residuals from Linear Model (previous)



These residuals look exponential. Clearly they are not random!

Residuals from Log-Transformed Model (previous)



These residuals are random looking. Better model!

Note

There are no questions testing you on nonlinear regression - it’s an optional topic, which I’ve included because it’s really good for you to be exposed to it. If we were doing the in-person 1052H, we’d have a lot more time to breathe, and we’d have done an assignment on that topic. But for the summer … that’s it!

Lecture 11

There will be a Lecture 11 posted next week going quickly through some optional topics, just for fun and for your personal exposure. If you ever encounter the ideas from Lecture 11, at least you'll know what they're called, so you can look them up (or talk to a statistician!)