Lesson 19

Clark

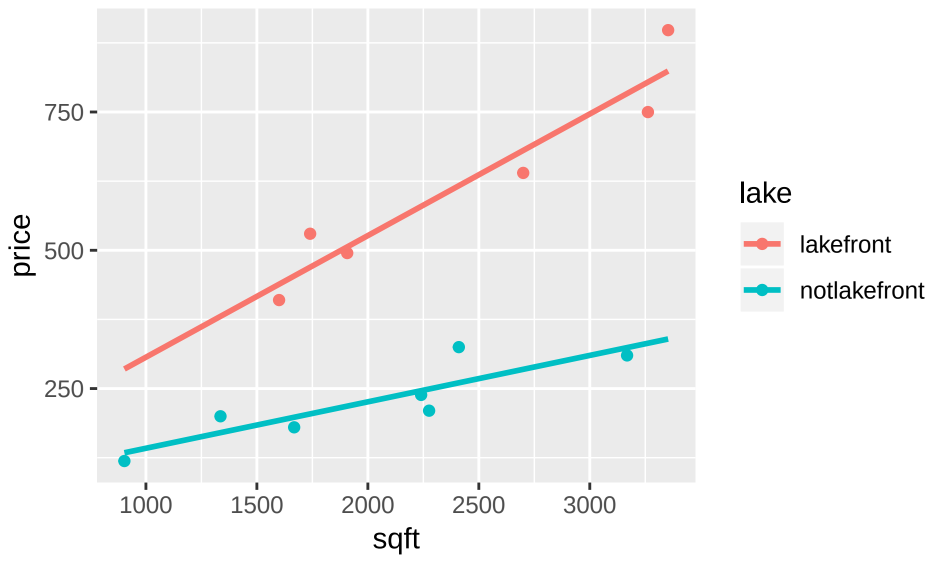
Admin

Recall last class we built a model with both quantitative and categorical explanatory variables. If we have two categories we can write our models as:

For this reason, we can think about these models as having seperate intercepts but the same slope. For our home prices, what we are saying with our models is that the relationship between square footage and price is the same for both lake front and not lake front houses, but the baseline cost differs. To put itn another way, if we have two houses, one on the lakefront and one not on the lakefront, the difference between the two prices is always expected to be:

Therefore, if we took, say a 1500 square foot house on the lakefront and a 1500 square foot house not on the lakefront the expected difference is the same as the difference betweena 3000 square foot house on the lakefront and a 3000 square foot house not on the lakefront. Let's look at a picture:

```
house.dat<-read.table("http://www.isi-stats.com/isi2/data/housing.txt",header=T)
house.dat <- house.dat %>% mutate(price=price.1000)%>%select(-price.1000)%>%
  group_by(lake)
house.dat %>% ggplot(aes(x=sqft,y=price,color=lake))+geom_point()+
  stat_smooth(method="lm",se=FALSE,fullrange=T)
```



Here we decided to draw seperate regression lines for each group (Note the `group_by` command prior to plotting). What can we say about the difference of the differences?

This suggests the possibilities of an interaction model. We can write the model as:

In R we fit it as:

```
contrasts(house.dat$lake)=contr.sum
inter.lm<-lm(price~sqft+lake+sqft:lake,data=house.dat)
summary(inter.lm)
```

```
##
## Call:
## lm(formula = price ~ sqft + lake + sqft:lake, data = house.dat)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -54.16 -28.60 -14.15  29.64  73.93
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 72.43889   45.66280   1.586  0.14711
## sqft         0.15192    0.01947   7.801  2.7e-05 ***
## lake1       14.32549   45.66280   0.314  0.76088
## sqft:lake1   0.06798    0.01947   3.491  0.00682 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 49.48 on 9 degrees of freedom
## Multiple R-squared:  0.9684, Adjusted R-squared:  0.9578
## F-statistic: 91.84 on 3 and 9 DF,  p-value: 4.547e-07
```

Let's use this to write out the fitted model for lakefront:

Fitted model for not lakefront:

Note we also can write the model as:

Which can be fit as:

```
contrasts(house.dat$lake)=contr.treatment
inter.lm<-lm(price~sqft+lake+sqft:lake,data=house.dat)
summary(inter.lm)
```

```
##
## Call:
## lm(formula = price ~ sqft + lake + sqft:lake, data = house.dat)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -54.16 -28.60 -14.15  29.64  73.93
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  86.76438   71.60612   1.212  0.25648
## sqft          0.21990    0.02831   7.769  2.8e-05 ***
## lake2       -28.65098   91.32560  -0.314  0.76088
## sqft:lake2   -0.13595    0.03895  -3.491  0.00682 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 49.48 on 9 degrees of freedom
## Multiple R-squared:  0.9684, Adjusted R-squared:  0.9578
## F-statistic: 91.84 on 3 and 9 DF,  p-value: 4.547e-07
```

Verify that these outcomes yield the same fitted model.

Note that for either output we get an F-statistic of 91.84 on 3 and 9 DF. This is testing:

This probably isn't the test we want in this case. Note that we also have P values associated with sqft, lake2, and sqft:lake2. These are testing:

Note that this is very similar to using Type III sums of squares. Alternatively we can use the F statistic to test the effects using (Note: page 329 is incorrect):

```r
library(car)
```

```
## Warning: package 'car' was built under R version 3.5.1
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

```
## The following object is masked from 'package:purrr':
##
##     some
```

```r
Anova(inter.lm,type=3)
```

```
## Anova Table (Type III tests)
##
## Response: price
##             Sum Sq Df F value    Pr(>F)
## (Intercept)   3594  1  1.4682  0.256483
## sqft        147747  1 60.3508 2.796e-05 ***
## lake           241  1  0.0984  0.760881
## sqft:lake    29829  1 12.1844  0.006824 **
## Residuals    22033  9
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Let's tear this apart a bit. One question you might have is, if we are doing this analysis and end up with different slopes and different intercepts, why not just split our data into two and fit two different regression models?

Certainly we could do:

```r
lakehouses<-house.dat %>% filter(lake=="lakefront")
lake.lm<-lm(price~sqft,data=lakehouses)
summary(lake.lm)
```

```
##
## Call:
## lm(formula = price ~ sqft, data = lakehouses)
##
## Residuals:
##      1      2      3      4      5      6
## -40.58  73.93 -28.60  60.52 -11.10 -54.16
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 86.76438   87.58078   0.991  0.37792
## sqft         0.21990    0.03462   6.352  0.00315 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 60.52 on 4 degrees of freedom
## Multiple R-squared:  0.9098, Adjusted R-squared:  0.8872
## F-statistic: 40.34 on 1 and 4 DF,  p-value: 0.003148
```

Which yields a fitted model of:

And we could do:

```r
nonlakehouses<-house.dat %>% filter(lake!="lakefront")
nonlake.lm<-lm(price~sqft,data=nonlakehouses)
summary(nonlake.lm)
```

```
##
## Call:
## lm(formula = price ~ sqft, data = nonlakehouses)
##
## Residuals:
##       1       2       3       4       5       6       7
##  29.637  64.480 -14.915 -14.149 -18.233 -39.171  -7.649
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 58.11341   44.02459    1.32  0.24403
## sqft         0.08394    0.02078    4.04  0.00992 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 38.43 on 5 degrees of freedom
## Multiple R-squared:  0.7655, Adjusted R-squared:  0.7186
## F-statistic: 16.32 on 1 and 5 DF,  p-value: 0.009923
```

Why might we not want to do this?

We still need to check our assumptions, what plots would we want to examine?

Note that our book makes a statement that may get hidden, but it is actually quite powerful and gets misinterpreted quite a bit. Note that our confidence intervals for our regression coefficients can be found from:

```r
confint(inter.lm)
```

```
##                     2.5 %      97.5 %
## (Intercept)  -75.2199149 248.7486787
## sqft           0.1558632   0.2839272
## lake2       -235.2438250 177.9418747
## sqft:lake2    -0.2240562  -0.0478453
```

These intervals are NOT confidence intervals for $\hat{y}$. Let's look at the regression model we are fitting:

If we want to find a Confidence Interval for $\hat{y}$ what would need:

Let's take a 2000 sq. foot house that's not on the lakefront. In order to find the variance of our prediction we need the variance of $\hat{\beta}_0$, the variance of $\hat{\beta}_1$ and the covariance between these two (recall the variance of a sum is the sum of the covariances). In R we can get the covariance matrix from `vcov(inter.lm)`. So we note that the variance here is:

```
pred.var=5127.43+2700^2*.000801+2*2700*-1.94454
pred.se=sqrt(pred.var)
pred.se
```

```
## [1] 21.59176
```

Using matrix algebra we can compute this as:

For any given observation, we can get the confidence intervals and the SE using:

```
conf.int=predict(inter.lm,data.frame(sqft=2700,lake="lakefront"),se.fit=TRUE,interval="confidence")
```

If we want a prediction interval, we need the variance associated with $\hat{y}$ AND the variance associated with y, which is $\hat{\sigma^2}$. So for instance if we want a prediction of a future lakefront house that's 2700, our variance will be `21.62^2+49.48^2`. So the SE for a prediction will be approx. 54. Our book makes the claim that to form a 95% PI we should use $\hat{y} \pm 2 * \hat{\sigma^2}$ which fails to account for the uncertainty in $\hat{\beta}$ terms. . . .

To find a prediction interval we can use:

```
pred.int=predict(inter.lm,data.frame(sqft=2700,lake="lakefront"),se.fit=TRUE,interval="prediction")
conf.int$fit
```

```
##        fit      lwr      upr
## 1 680.4814 631.5573 729.4055
```

```
pred.int$fit
```

```
##        fit      lwr      upr
## 1 680.4814 558.3277 802.6351
```