Class 8: Linear Regression 2

MGSC 310

Prof. Jonathan Hersh

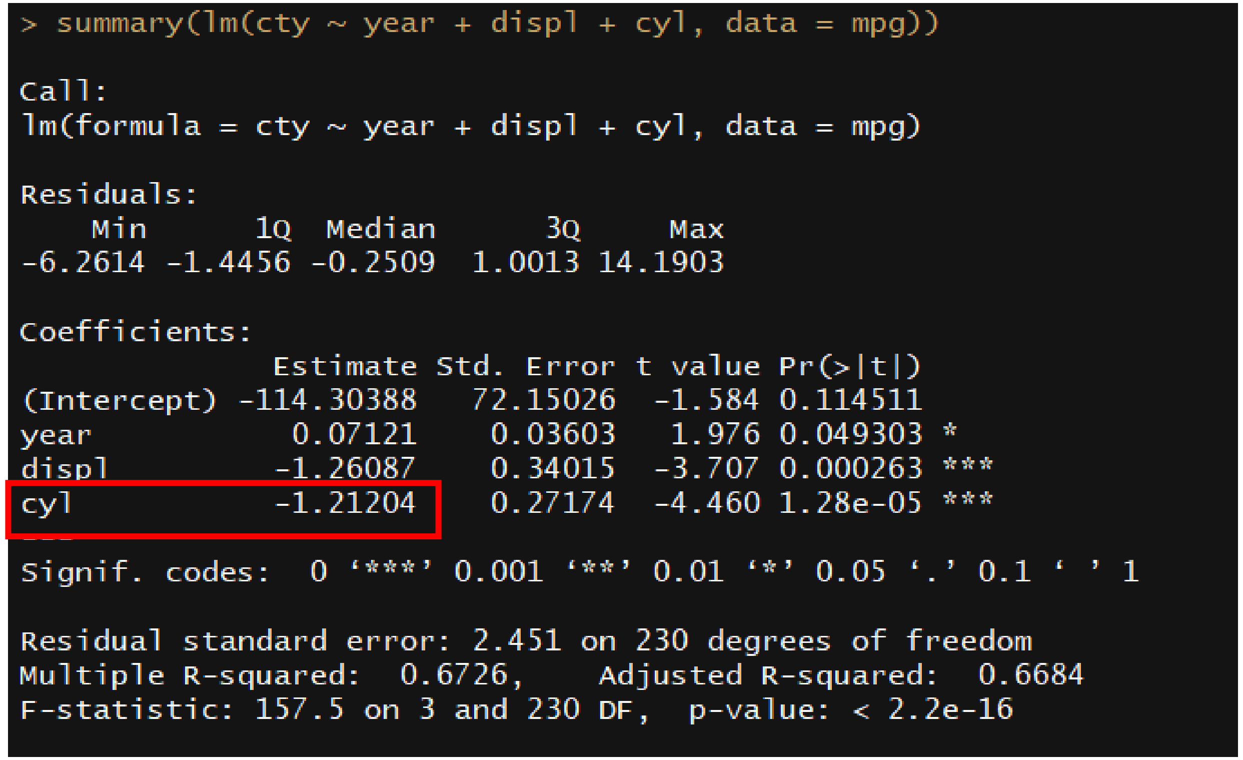
Class 8: Announcements

1. TA Office Hours:
   - Tuesdays: 5:30 – 7
   - Thursdays: 12:30-2
   - Mondays: 5-6:30
2. Quiz 3 posted, due Thursday @ midnight
3. Be sure you are following along with the course reading (ISLR pp 59-80 covered today)
4. Problem Set 2 Posted, Due Sept 29
5. **Problem Set Solutions:**
   - Typically submit these via hard copy b/c of cheating
   - Cannot do this this year ☹
   - For now: TA/Instructor Office Hours will share solutions for specific Qs. Hard copies will be available on campus if you care to pick these up.

Class 8: Outline

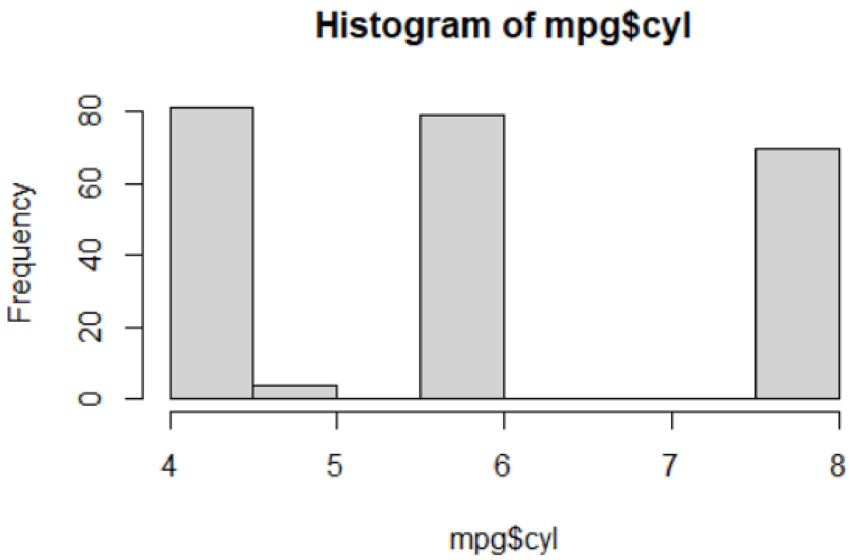
1. Discrete/Qualitative Independent Variables
2. Inference/Hypothesis Testing in Linear Models
3. Model Evaluation
   - Predicted/True Plots, RMSE, and R-Squared
4. Regression Lab 2

Class 7 Lab



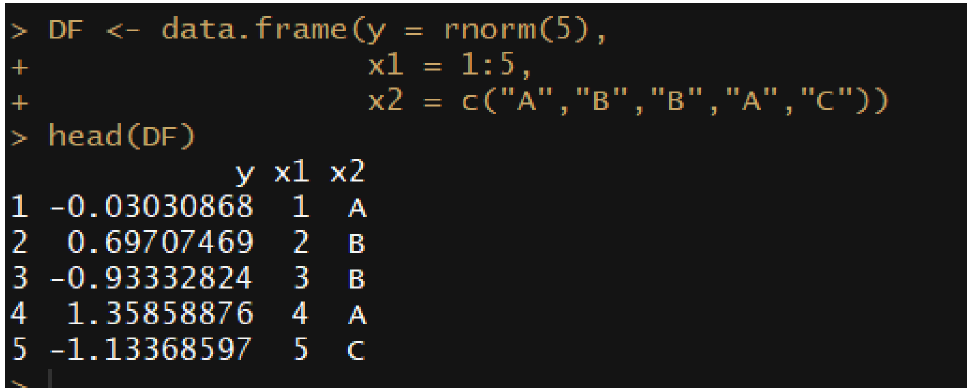
```
> summary(lm(cty ~ year + displ + cyl, data = mpg))

Call:
lm(formula = cty ~ year + displ + cyl, data = mpg)

Residuals:
    Min      1Q  Median      3Q     Max 
-6.2614 -1.4456 -0.2509  1.0013 14.1903 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) -114.30388   72.15026  -1.584 0.114511    
year           0.07121    0.03603   1.976 0.049303 *  
displ         -1.26087    0.34015  -3.707 0.000263 ***
cyl           -1.21204    0.27174  -4.460 1.28e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.451 on 230 degrees of freedom
Multiple R-squared:  0.6726,	Adjusted R-squared:  0.6684 
F-statistic: 157.5 on 3 and 230 DF,  p-value: < 2.2e-16
```

- Increasing engine cylinders by one cylinder decreases city mile per gallon by -1.21, holding year and engine size fixed.

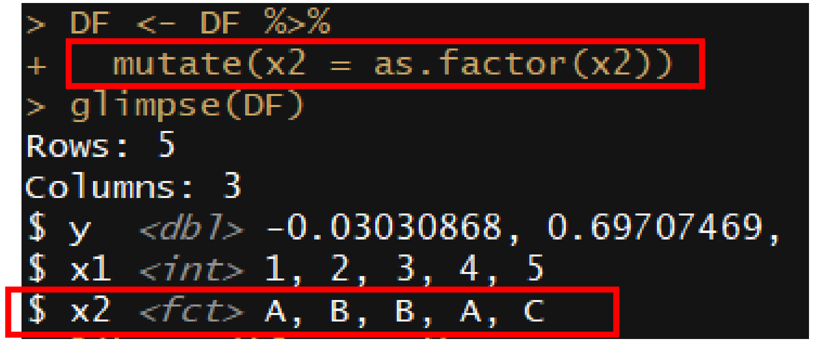


How do we handle qualitative/discrete independent variables in our regression model?

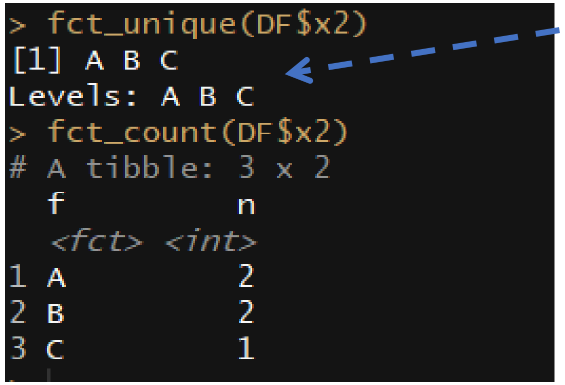
Factors: Like Strings But Better!



```
> DF <- data.frame(y = rnorm(5),
+                  x1 = 1:5,
+                  x2 = c("A","B","B","A","C"))
> head(DF)
            y x1 x2
1 -0.03030868  1  A
2  0.69707469  2  B
3 -0.93332824  3  B
4  1.35858876  4  A
5 -1.13368597  5  C
```



```
> DF <- DF %>%
+   mutate(x2 = as.factor(x2))
> glimpse(DF)
Rows: 5
Columns: 3
$ y  <dbl> -0.03030868, 0.69707469,
$ x1 <int> 1, 2, 3, 4, 5
$ x2 <fct> A, B, B, A, C
```



```
> fct_unique(DF$x2)
[1] A B C
Levels: A B C
> fct_count(DF$x2)
# A tibble: 3 x 2
  f         n
  <fct> <int>
1 A         2
2 B         2
3 C         1
```

- **Factors hold string values efficiently**

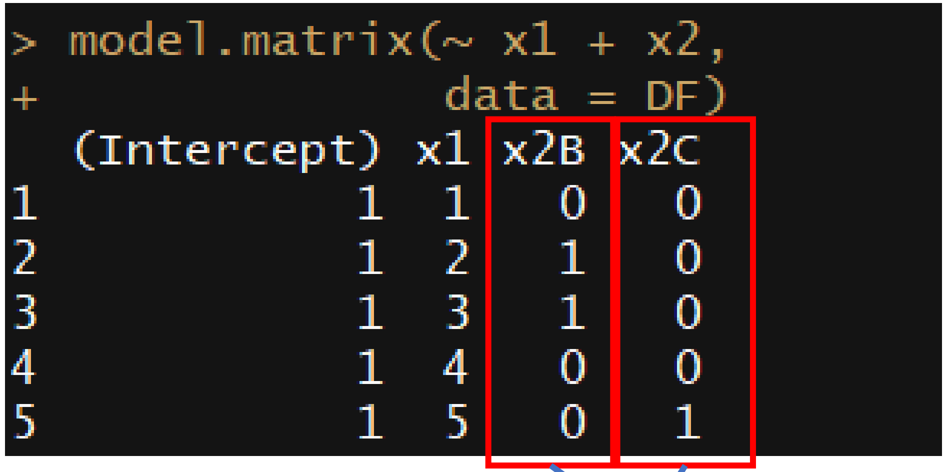
- Instead of holding a character string, it just holds a number and has a key which associates each number with a unique value of the string

- If we convert the character string x2 to a factor, we see A = 1, B = 2, C = 3, etc

- Will say more about working w/ factors but know that the package forcats is your friend

Incorporating Qualitative/Discrete Information Into Regressions

$$y = \beta_0 + \beta_1 \cdot x_1 + \beta_{state} \cdot x_{state}?$$



```
> model.matrix(~ x1 + x2,
+              data = DF)
  (Intercept) x1 x2B x2C
1           1  1   0   0
2           1  2   1   0
3           1  3   1   0
4           1  4   0   0
5           1  5   0   1
```

$$y = \beta_0 + \beta_1 \cdot x_1 + \beta_{stateB} \cdot StateB + \beta_{stateC} \cdot StateC?$$

- **Suppose A = State A, B = State B, etc.**
- How do include state into a regression?

- **model.matrix() function shows how we convert a factor to a regression matrix**
- Each “level” of the factor gets it own column, and a binary indicator of whether that level is true for that observation

- **Columns x2B and x2C are called “dummy” variables**
- Aka “one hot encoding” in machine learning

Why Does Every Factor Level Not Get Its Own Dummy Variable?

Intercept	Y	x1	x2_A	X2_B	X2_C
1	0.4	1	1	0	0
1	-0.5	2	0	1	0
1	-0.3	3	0	1	0
1	0.1	4	1	0	0
1	-0.8	5	0	0	1

x2
"A"
"B"
"B"
"A"
"C"

Why? Because estimates are computed as

$$\beta = (X^T X)^{-1} X^T Y$$

Linear algebra requires $(X^T X)^{-1}$ to be full column rank i.e. each column of X must be linearly independent.

Intercept + X2_A + X_B + X2_C only "span" 3 dimensions

Excluded Base Level of Factor Becomes Base Level for Interpretation

		β_{x1}	β_{x2_B}	β_{x2_C}
Intercept	**Y**	**x1**	**x2_B**	**X2_C**
1	0.4	1	0	0
1	-0.5	2	1	0
1	-0.3	3	1	0
1	0.1	4	0	0
1	-0.8	5	0	1

Interpreting Dummy Variable Coefficients:

- β_{x2_B}: We estimate y will increase by B *if it is of category B* relative to category A
- β_{x2_C}: We estimate y will increase by β_{x2_C} *if it is of category C* relative to category A

Binary/Dummy Variable coefficients can ONLY be interpreted relative to each other

Left out category (i.e. no column) is comparison category

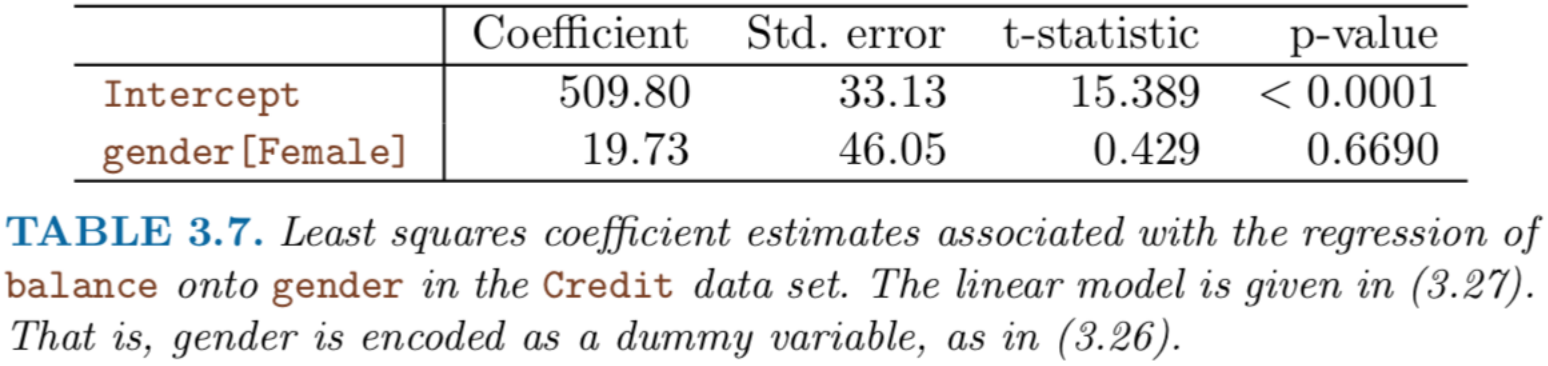
Side Note, This is How Wage and Race Discrimination Regressions Are Performed

$x_i = 1$, if female

$x_i = 0$, if male

$y_i =$ credit card balance

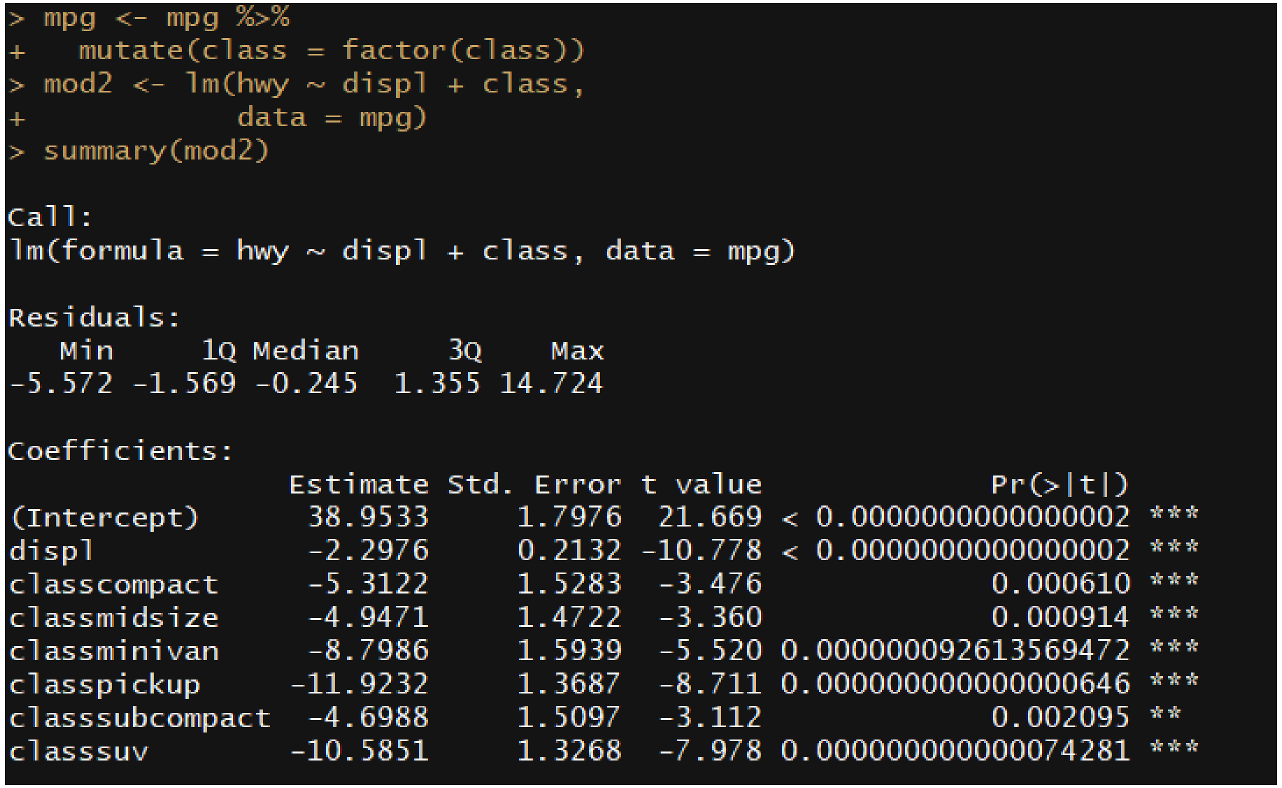
$$y_i = \beta_0 + \beta_1 \cdot x_i + \epsilon_i = \begin{cases} \beta_0 + \beta_1 \cdot x_i + \epsilon_i & i = female \\ \beta_0 + \epsilon_i & i = male \end{cases}$$



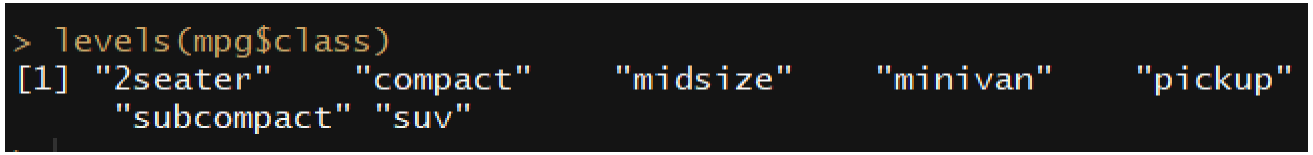
	Coefficient	Std. error	t-statistic	p-value
Intercept	509.80	33.13	15.389	< 0.0001
gender[Female]	19.73	46.05	0.429	0.6690

TABLE 3.7. *Least squares coefficient estimates associated with the regression of* balance *onto* gender *in the* Credit *data set. The linear model is given in (3.27). That is, gender is encoded as a dummy variable, as in (3.26).*

Interpreting Binary/Dummy Coefficients With mpg Dataset



```
> mpg <- mpg %>%
+   mutate(class = factor(class))
> mod2 <- lm(hwy ~ displ + class,
+            data = mpg)
> summary(mod2)

Call:
lm(formula = hwy ~ displ + class, data = mpg)

Residuals:
   Min     1Q Median     3Q    Max 
-5.572 -1.569 -0.245  1.355 14.724 

Coefficients:
                Estimate Std. Error t value             Pr(>|t|)    
(Intercept)      38.9533     1.7976  21.669 < 0.0000000000000002 ***
displ            -2.2976     0.2132 -10.778 < 0.0000000000000002 ***
classcompact     -5.3122     1.5283  -3.476             0.000610 ***
classmidsize     -4.9471     1.4722  -3.360             0.000914 ***
classminivan     -8.7986     1.5939  -5.520 0.000000092613569472 ***
classpickup     -11.9232     1.3687  -8.711 0.000000000000000646 ***
classsubcompact  -4.6988     1.5097  -3.112             0.002095 ** 
classsuv        -10.5851     1.3268  -7.978 0.000000000000074281 ***
```



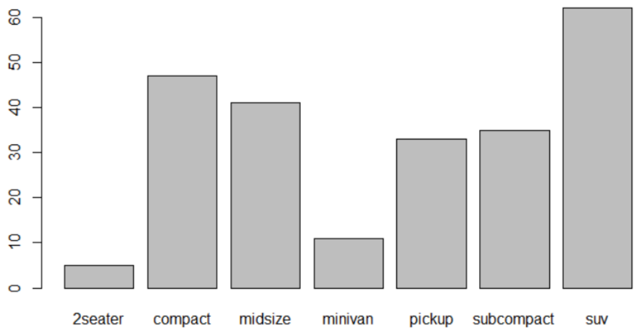
```
> levels(mpg$class)
[1] "2seater"    "compact"    "midsize"    "minivan"    "pickup"    
    "subcompact" "suv"
```

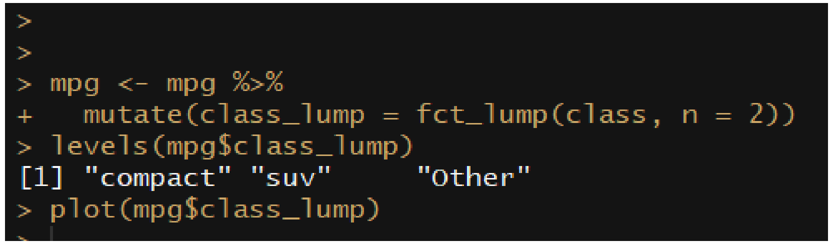
- $\beta_{compact}$: Holding engine size (displacement) fixed we estimated a compact car gets 5.31 *worse highway miles per gallon* relative to the excluded category!

- What is the excluded category?

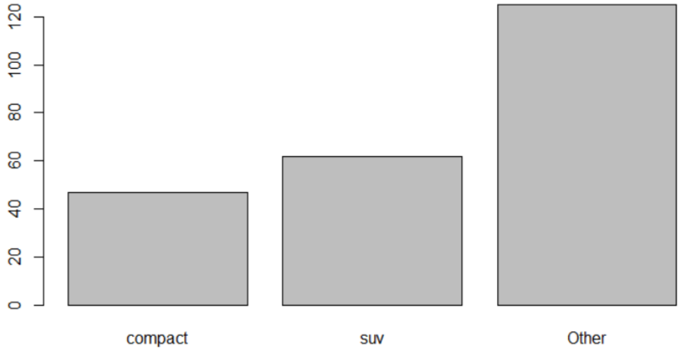
- By default it's the first level of the factor, here "2seater"

Changing Factor Levels with fct_lump



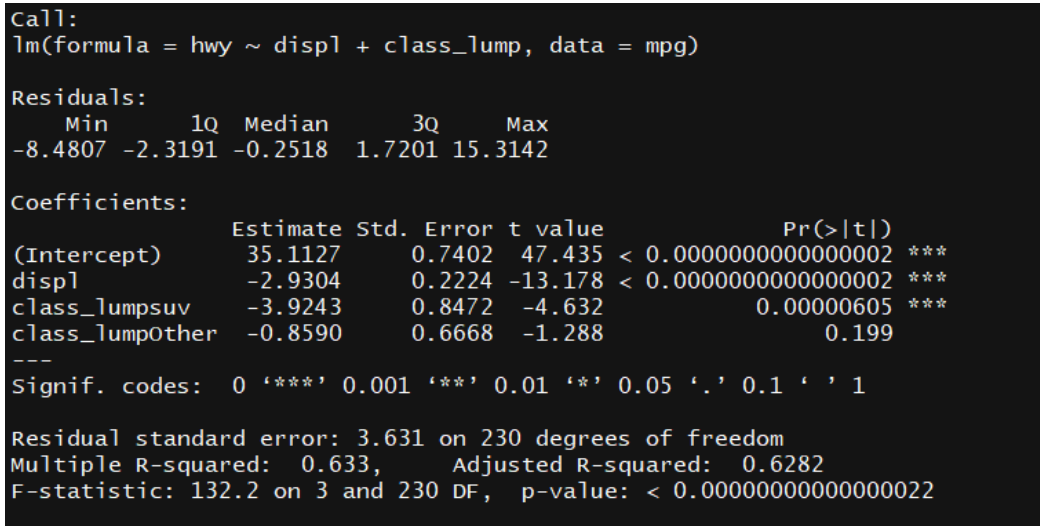


```
>
>
> mpg <- mpg %>%
+   mutate(class_lump = fct_lump(class, n = 2))
> levels(mpg$class_lump)
[1] "compact" "suv"     "Other"
> plot(mpg$class_lump)
>
```



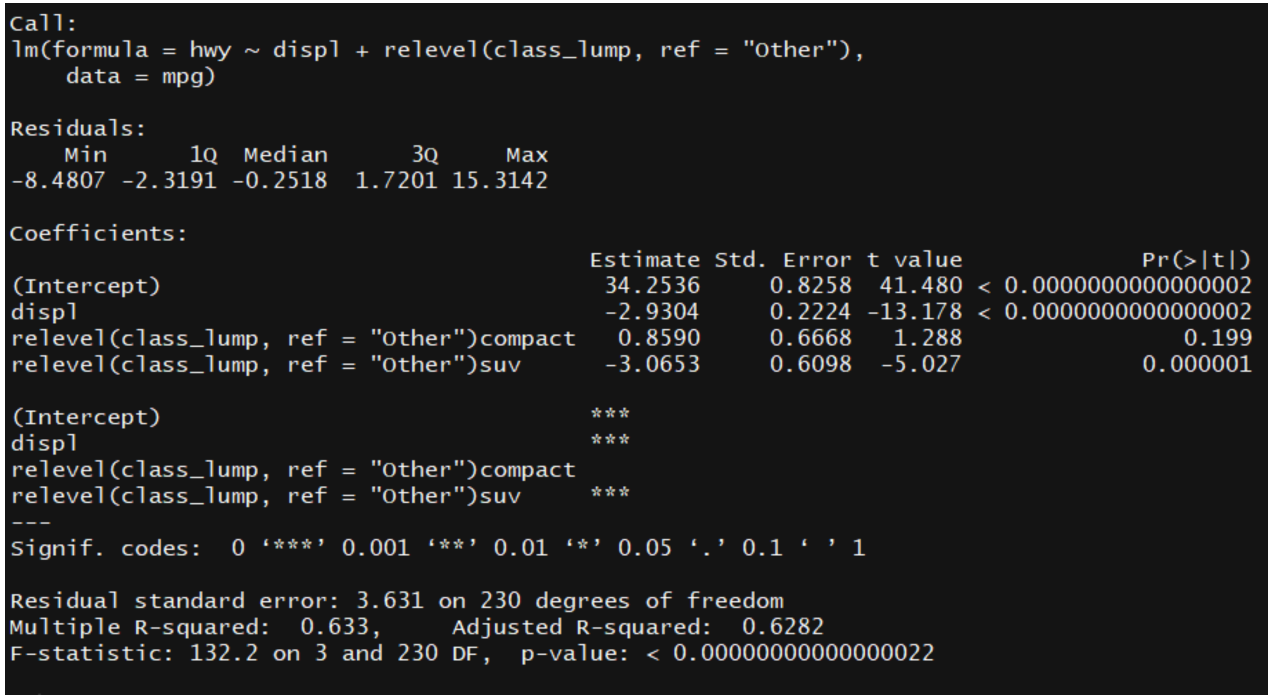
- Suppose we want to change the levels of a factor?
- Many functions in ‘forcats’ to do this but fct_lump is useful.
  - n = 2 specifies how many explicit factors we want
- Here we’ve only given explicit labels to “compact” and “suv”. Every other level is placed into “other” category

Estimating Model With Simplified Factor Levels



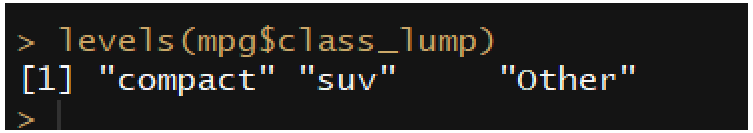
```
Call:
lm(formula = hwy ~ displ + class_lump, data = mpg)

Residuals:
    Min      1Q  Median      3Q     Max 
-8.4807 -2.3191 -0.2518  1.7201 15.3142 

Coefficients:
                Estimate Std. Error t value             Pr(>|t|)    
(Intercept)      35.1127     0.7402  47.435 < 0.0000000000000002 ***
displ            -2.9304     0.2224 -13.178 < 0.0000000000000002 ***
class_lumpsuv    -3.9243     0.8472  -4.632           0.00000605 ***
class_lumpOther  -0.8590     0.6668  -1.288                0.199    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.631 on 230 degrees of freedom
Multiple R-squared:  0.633,	Adjusted R-squared:  0.6282 
F-statistic: 132.2 on 3 and 230 DF,  p-value: < 0.00000000000000022
```

- Change reference category with “relevel()”



```
Call:
lm(formula = hwy ~ displ + relevel(class_lump, ref = "Other"), 
    data = mpg)

Residuals:
    Min      1Q  Median      3Q     Max 
-8.4807 -2.3191 -0.2518  1.7201 15.3142 

Coefficients:
                                          Estimate Std. Error t value             Pr(>|t|)
(Intercept)                                34.2536     0.8258  41.480 < 0.0000000000000002
displ                                      -2.9304     0.2224 -13.178 < 0.0000000000000002
relevel(class_lump, ref = "Other")compact   0.8590     0.6668   1.288                0.199
relevel(class_lump, ref = "Other")suv      -3.0653     0.6098  -5.027             0.000001
                                             
(Intercept)                               ***
displ                                     ***
relevel(class_lump, ref = "Other")compact    
relevel(class_lump, ref = "Other")suv     ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.631 on 230 degrees of freedom
Multiple R-squared:  0.633,	Adjusted R-squared:  0.6282 
F-statistic: 132.2 on 3 and 230 DF,  p-value: < 0.00000000000000022
```

- What is the reference category?



```
> levels(mpg$class_lump)
[1] "compact" "suv"     "Other"  
> 
```

Factor Variable As Switches, Continuous Variables As Sliders

$$hwy_i = \beta_0 + \beta_1 x_{displ} + \epsilon_i$$

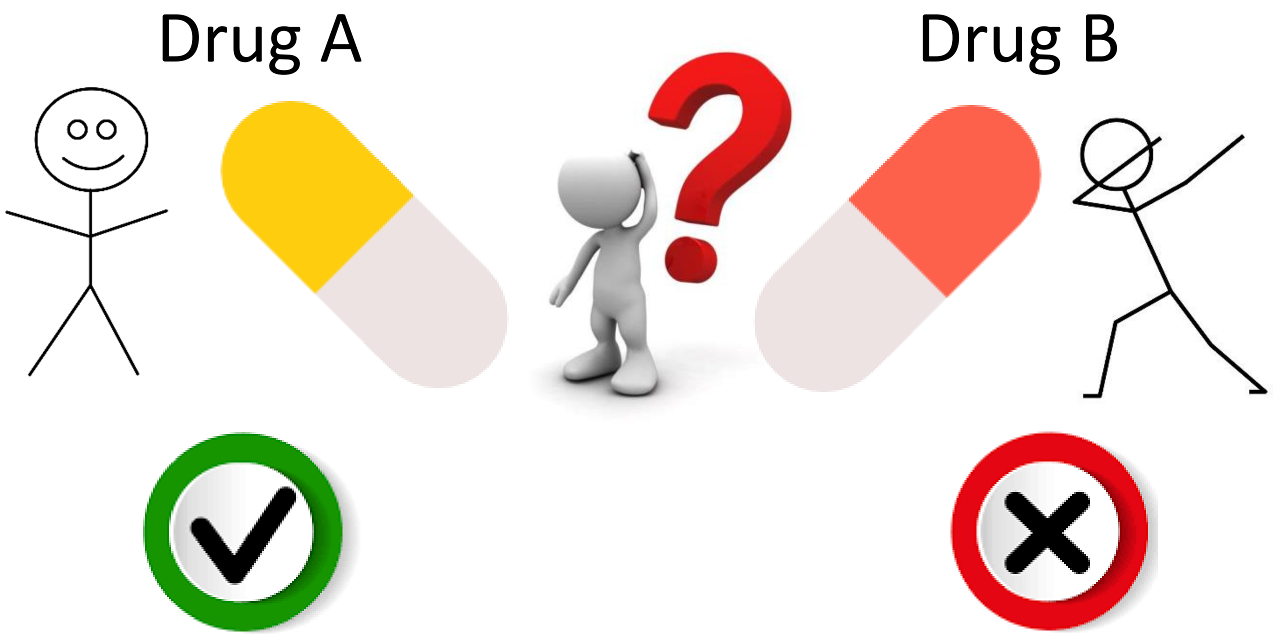
$$\begin{aligned} & hwy_i \\ & = \beta_0 + \beta_1 x_{compact} + \beta_2 \cdot x_{suv} + \epsilon_i \end{aligned}$$

$$\begin{aligned} & hwy_i \\ & = \beta_0 + \beta_1 x_{compact} + \beta_2 \cdot x_{suv} \\ & + \beta_3 \cdot x_{displ} + \epsilon_i \end{aligned}$$

Class 8: Outline

1. Discrete/Qualitative Independent Variables
2. **Inference/Hypothesis Testing in Linear Models**
3. Model Evaluation
   - Predicted/True Plots, RMSE, and R-Squared
4. Regression Lab 2

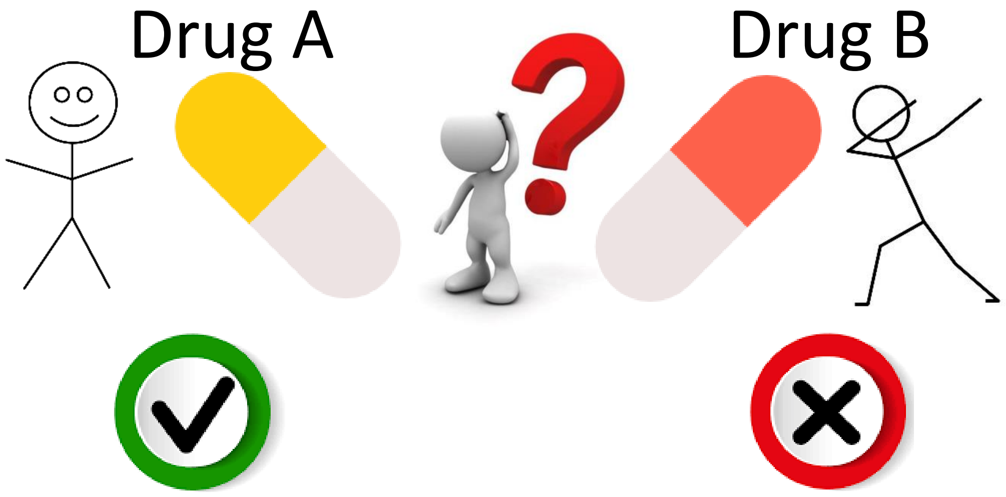
What Are P-Values?



- Suppose we want to know the effectiveness of Drug A vs Drug B
- We can give Drug A to 1 person, and give drug B to 1 person.
- Suppose person A gets better, and person B does not.
- Can we conclude drug A is better than drug B?

Adapted from the excellent StatQuest: https://youtu.be/vemZtEM63GY

What Are P-Values?



- **No! Perhaps…**
  - Person B didn’t follow the instructions
  - Person B has pre-existing conditions
  - Person A is healthier
- Only by repeating the experiment many times can we learn whether Drug A > Drug B
- Ideally we express our confidence as a quantitative number, how likely it is that we find Drug A > Drug B only due to chance?

Repeating Experiment With More Observations

Drug A

Cured!	Not Cured!
73	125

37% Cured!

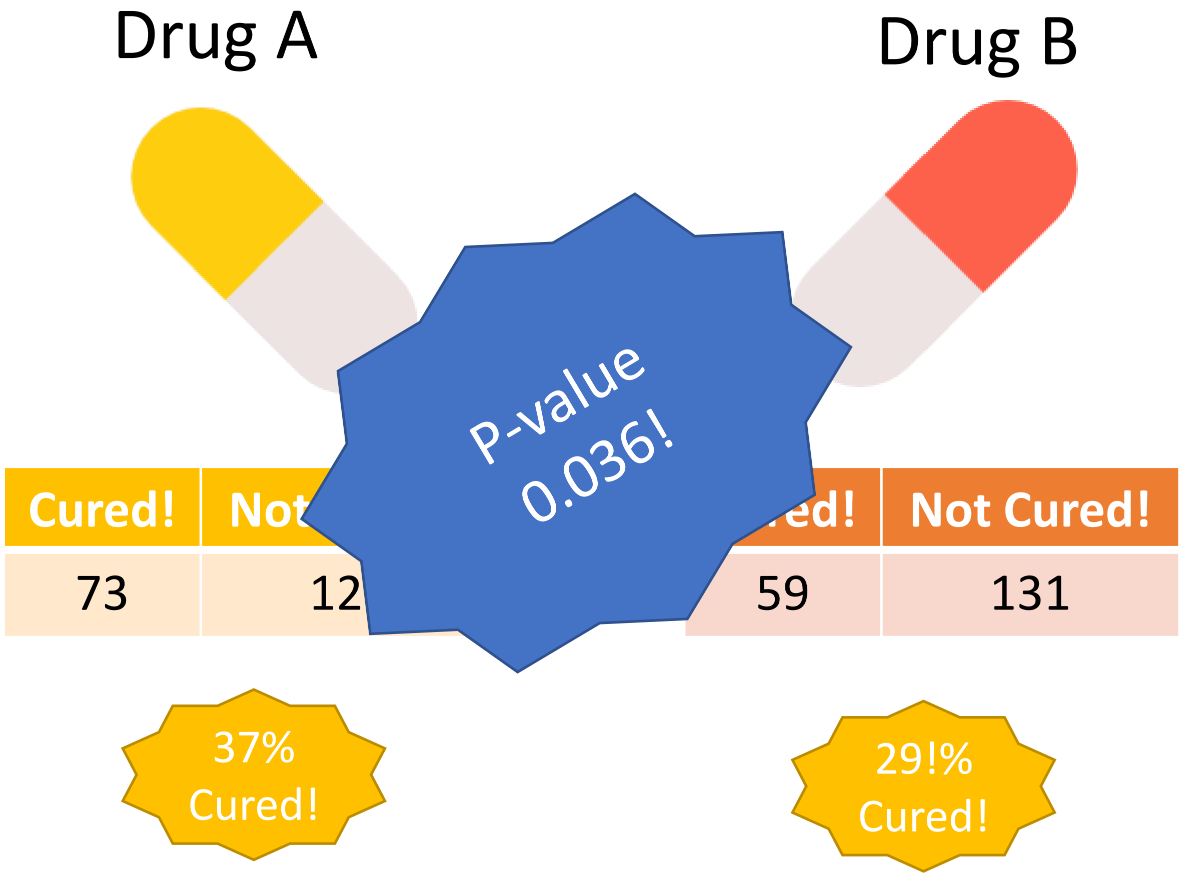
Drug B

Cured!	Not Cured!
59	131

29!% Cured!

- Suppose we repeat the experiment with ~400 patients
- We find that drug A has a cure rate of 37%, drug B a cure rate of 29%.
- But: No study is perfect. There are always random things that could influence drug A vs B
- **The p-value is a number between 0 and 1 that tells us how confident we should be between one hypothesis (H_0 or null) and another (H_1, alternative)**

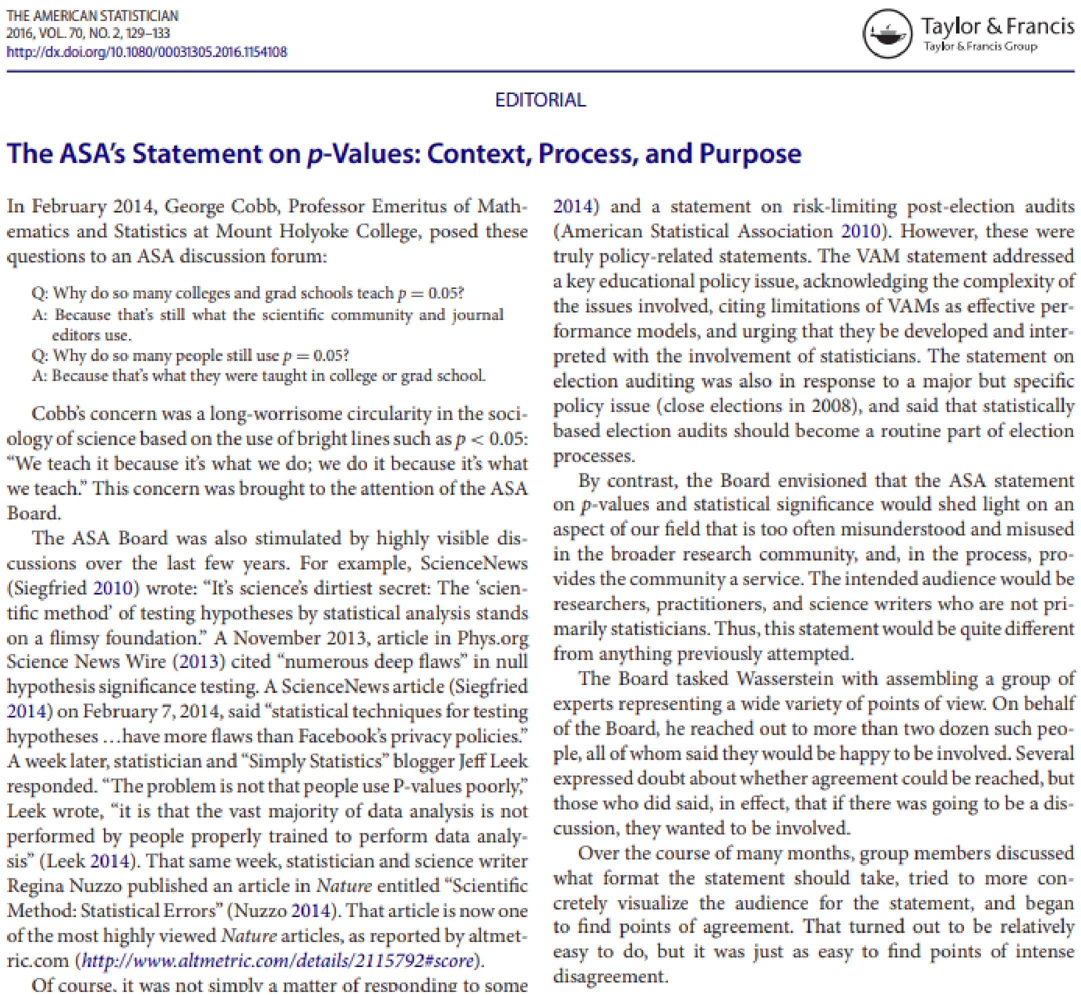
Repeating Experiment With More Observations



- P value close to 0:
  - Result not likely due to chance
- P value close to 1:
  - More likely result is due to chance

- What is “enough evidence”?
  - Alpha = critical value
  - Commonly use ~ 0.05, 0.01, or 0.001
  - i.e. 5%, 1% or 0.1% chance difference due to randomness and there’s no difference

If You Find This Confusing You Are In Good Company



THE AMERICAN STATISTICIAN
2016, VOL. 70, NO. 2, 129–133
http://dx.doi.org/10.1080/00031305.2016.1154108

Taylor & Francis
Taylor & Francis Group

EDITORIAL

The ASA's Statement on *p*-Values: Context, Process, and Purpose

In February 2014, George Cobb, Professor Emeritus of Mathematics and Statistics at Mount Holyoke College, posed these questions to an ASA discussion forum:

> Q: Why do so many colleges and grad schools teach $p = 0.05$?
> A: Because that's still what the scientific community and journal editors use.
> Q: Why do so many people still use $p = 0.05$?
> A: Because that's what they were taught in college or grad school.

Cobb's concern was a long-worrisome circularity in the sociology of science based on the use of bright lines such as $p < 0.05$: "We teach it because it's what we do; we do it because it's what we teach." This concern was brought to the attention of the ASA Board.

The ASA Board was also stimulated by highly visible discussions over the last few years. For example, ScienceNews (Siegfried 2010) wrote: "It's science's dirtiest secret: The 'scientific method' of testing hypotheses by statistical analysis stands on a flimsy foundation." A November 2013, article in Phys.org Science News Wire (2013) cited "numerous deep flaws" in null hypothesis significance testing. A ScienceNews article (Siegfried 2014) on February 7, 2014, said "statistical techniques for testing hypotheses …have more flaws than Facebook's privacy policies." A week later, statistician and "Simply Statistics" blogger Jeff Leek responded. "The problem is not that people use P-values poorly," Leek wrote, "it is that the vast majority of data analysis is not performed by people properly trained to perform data analysis" (Leek 2014). That same week, statistician and science writer Regina Nuzzo published an article in *Nature* entitled "Scientific Method: Statistical Errors" (Nuzzo 2014). That article is now one of the most highly viewed *Nature* articles, as reported by altmetric.com (*http://www.altmetric.com/details/2115792#score*).

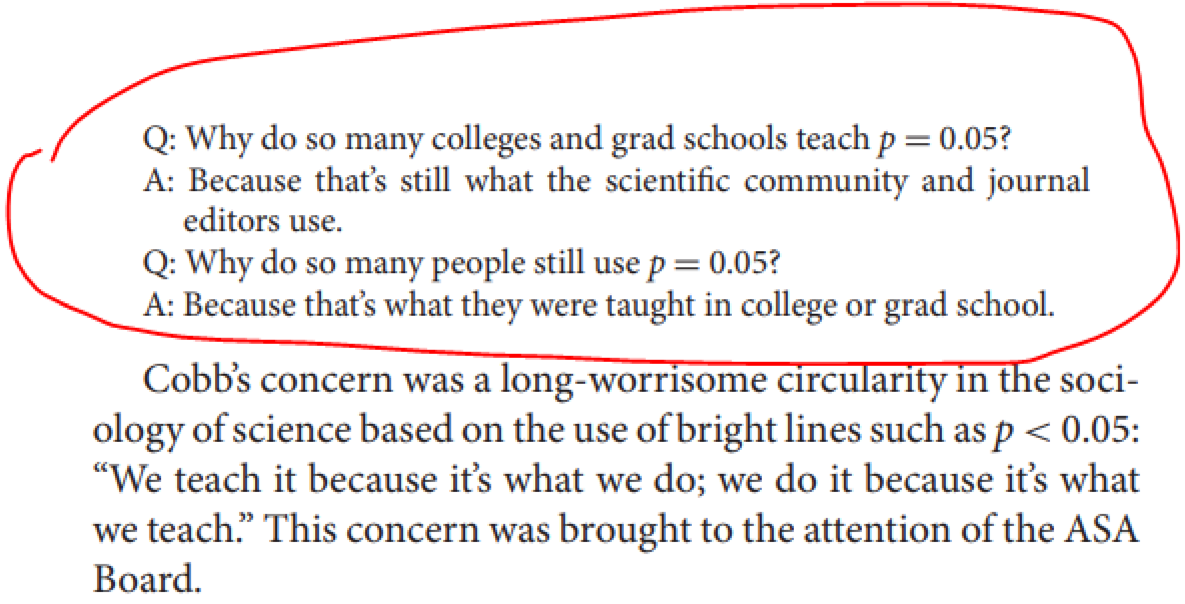
Of course, it was not simply a matter of responding to some

2014) and a statement on risk-limiting post-election audits (American Statistical Association 2010). However, these were truly policy-related statements. The VAM statement addressed a key educational policy issue, acknowledging the complexity of the issues involved, citing limitations of VAMs as effective performance models, and urging that they be developed and interpreted with the involvement of statisticians. The statement on election auditing was also in response to a major but specific policy issue (close elections in 2008), and said that statistically based election audits should become a routine part of election processes.

By contrast, the Board envisioned that the ASA statement on *p*-values and statistical significance would shed light on an aspect of our field that is too often misunderstood and misused in the broader research community, and, in the process, provides the community a service. The intended audience would be researchers, practitioners, and science writers who are not primarily statisticians. Thus, this statement would be quite different from anything previously attempted.

The Board tasked Wasserstein with assembling a group of experts representing a wide variety of points of view. On behalf of the Board, he reached out to more than two dozen such people, all of whom said they would be happy to be involved. Several expressed doubt about whether agreement could be reached, but those who did said, in effect, that if there was going to be a discussion, they wanted to be involved.

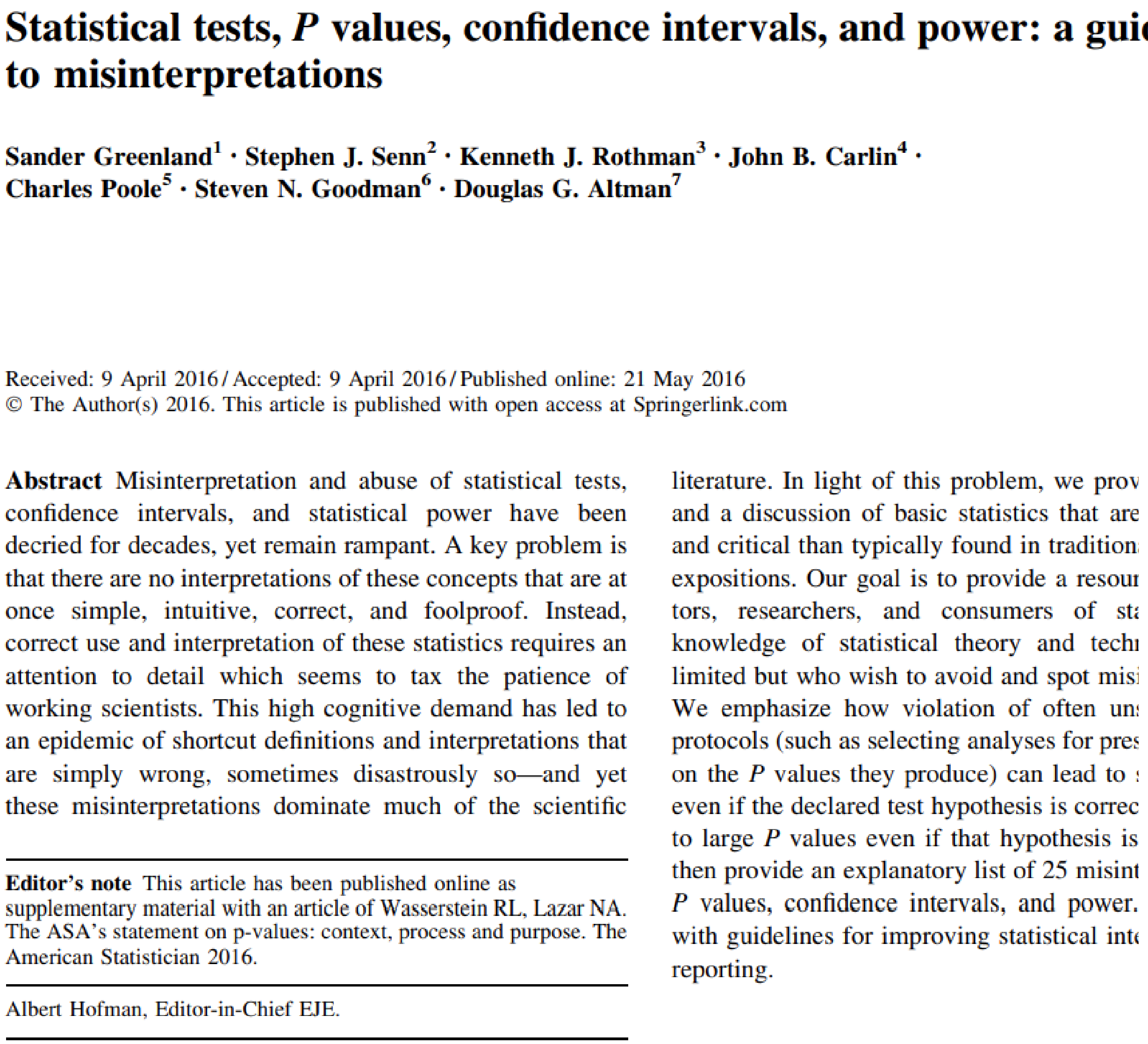
Over the course of many months, group members discussed what format the statement should take, tried to more concretely visualize the audience for the statement, and began to find points of agreement. That turned out to be relatively easy to do, but it was just as easy to find points of intense disagreement.



Q: Why do so many colleges and grad schools teach $p = 0.05$?
A: Because that's still what the scientific community and journal editors use.
Q: Why do so many people still use $p = 0.05$?
A: Because that's what they were taught in college or grad school.

Cobb's concern was a long-worrisome circularity in the sociology of science based on the use of bright lines such as $p < 0.05$: "We teach it because it's what we do; we do it because it's what we teach." This concern was brought to the attention of the ASA Board.

P-Values Commonly Misused or Misunderstood



Statistical tests, *P* values, confidence intervals, and power: a guide to misinterpretations

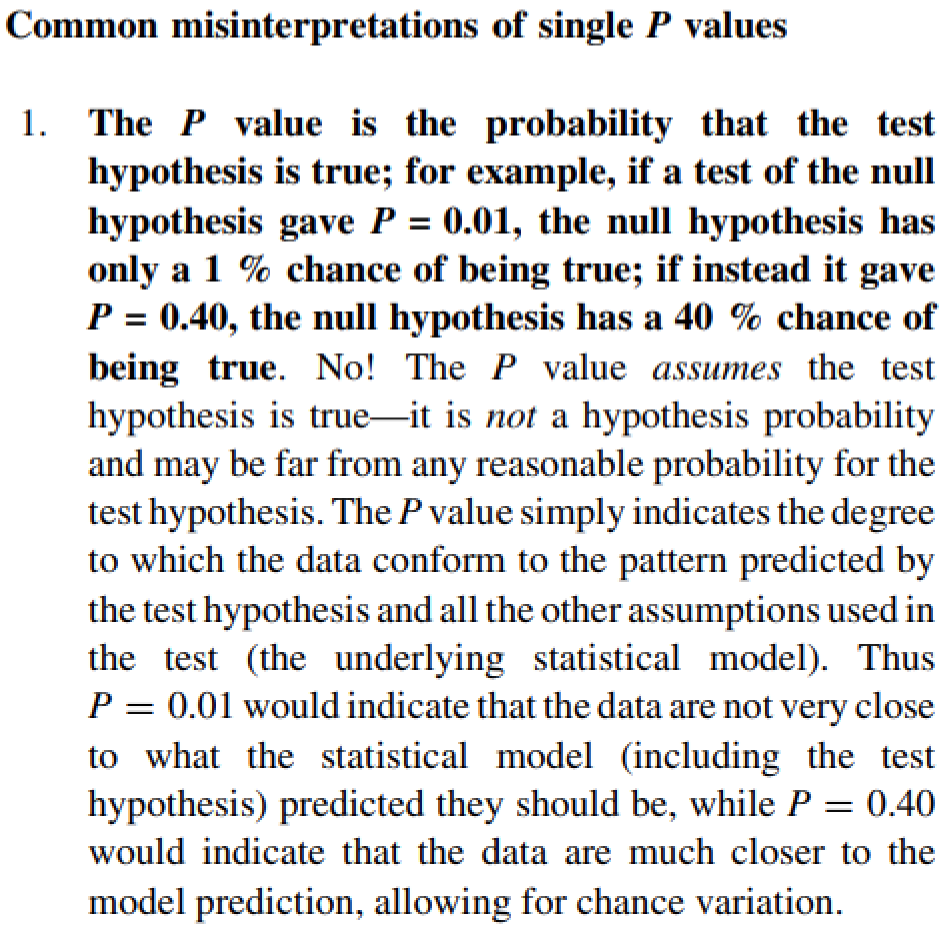
Sander Greenland[1] · Stephen J. Senn[2] · Kenneth J. Rothman[3] · John B. Carlin[4] · Charles Poole[5] · Steven N. Goodman[6] · Douglas G. Altman[7]

Received: 9 April 2016 / Accepted: 9 April 2016 / Published online: 21 May 2016

Abstract Misinterpretation and abuse of statistical tests, confidence intervals, and statistical power have been decried for decades, yet remain rampant. A key problem is that there are no interpretations of these concepts that are at once simple, intuitive, correct, and foolproof. Instead, correct use and interpretation of these statistics requires an attention to detail which seems to tax the patience of working scientists. This high cognitive demand has led to an epidemic of shortcut definitions and interpretations that are simply wrong, sometimes disastrously so—and yet these misinterpretations dominate much of the scientific literature. In light of this problem, we prov and a discussion of basic statistics that are and critical than typically found in tradition expositions. Our goal is to provide a resour tors, researchers, and consumers of sta knowledge of statistical theory and techı limited but who wish to avoid and spot misi We emphasize how violation of often un: protocols (such as selecting analyses for pres on the *P* values they produce) can lead to : even if the declared test hypothesis is correc to large *P* values even if that hypothesis is then provide an explanatory list of 25 misint *P* values, confidence intervals, and power. with guidelines for improving statistical inte reporting.

Editor's note This article has been published online as supplementary material with an article of Wasserstein RL, Lazar NA. The ASA's statement on p-values: context, process and purpose. The American Statistician 2016.

Albert Hofman, Editor-in-Chief EJE.



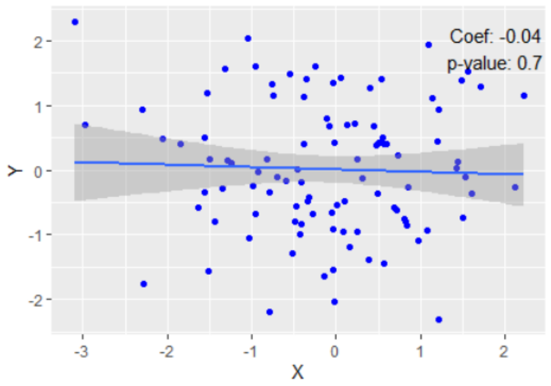
Common misinterpretations of single *P* values

1. **The *P* value is the probability that the test hypothesis is true; for example, if a test of the null hypothesis gave *P* = 0.01, the null hypothesis has only a 1 % chance of being true; if instead it gave *P* = 0.40, the null hypothesis has a 40 % chance of being true**. No! The *P* value *assumes* the test hypothesis is true—it is *not* a hypothesis probability and may be far from any reasonable probability for the test hypothesis. The *P* value simply indicates the degree to which the data conform to the pattern predicted by the test hypothesis and all the other assumptions used in the test (the underlying statistical model). Thus $P = 0.01$ would indicate that the data are not very close to what the statistical model (including the test hypothesis) predicted they should be, while $P = 0.40$ would indicate that the data are much closer to the model prediction, allowing for chance variation.

Hypothesis Test for Coefficients

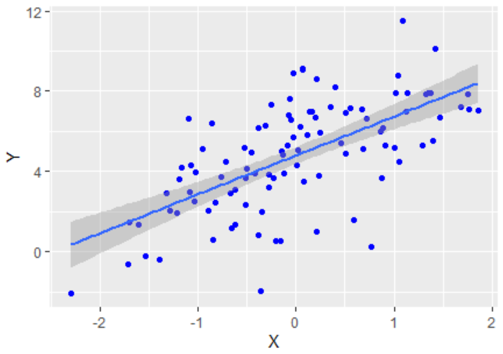
Null Hypothesis (H_0):

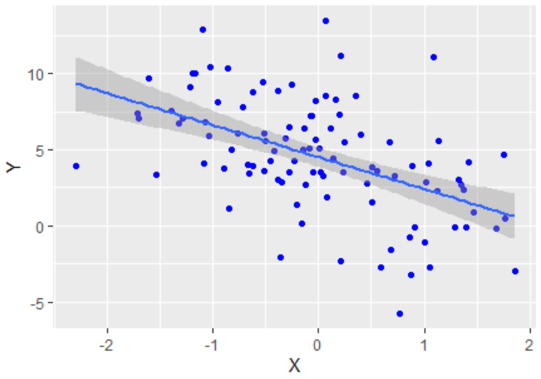
- There is no linear relationship between X and Y
- $\beta = 0$



Alternative Hypothesis (H_1)

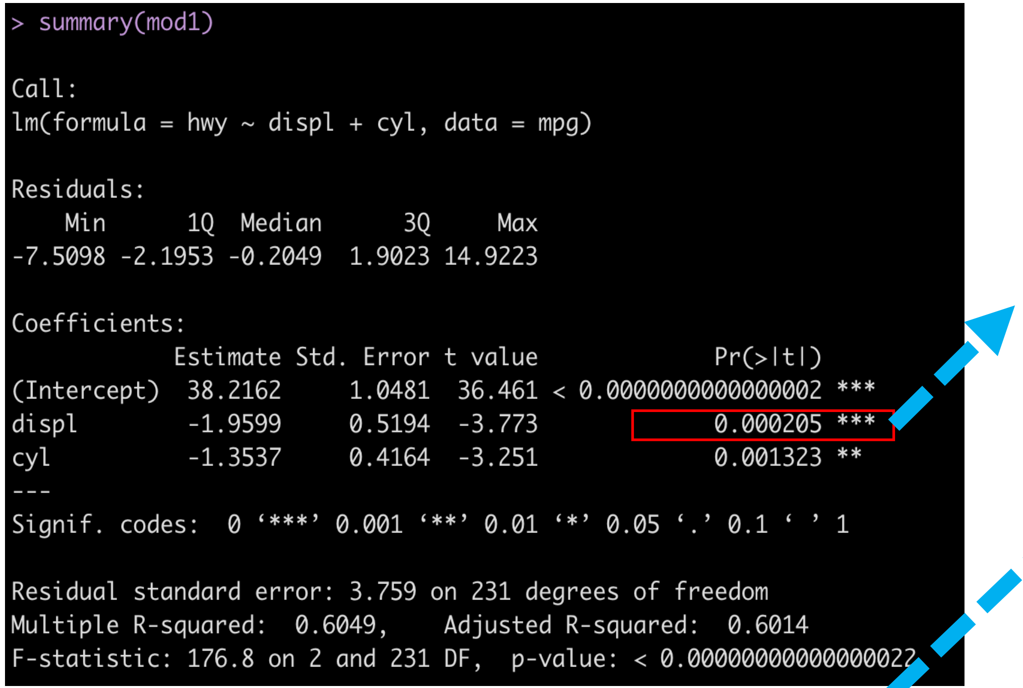
- There is some linear relationship between X and Y





We either **reject the null hypothesis** or **fail to reject the null hypothesis**
Based on a chosen critical value of alpha

What Do p-values For Coefficients Measure?



```
> summary(mod1)

Call:
lm(formula = hwy ~ displ + cyl, data = mpg)

Residuals:
    Min      1Q  Median      3Q     Max
-7.5098 -2.1953 -0.2049  1.9023 14.9223

Coefficients:
            Estimate Std. Error t value             Pr(>|t|)
(Intercept)  38.2162     1.0481  36.461 < 0.0000000000000002 ***
displ        -1.9599     0.5194  -3.773             0.000205 ***
cyl          -1.3537     0.4164  -3.251             0.001323 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.759 on 231 degrees of freedom
Multiple R-squared:  0.6049,	Adjusted R-squared:  0.6014
F-statistic: 176.8 on 2 and 231 DF,  p-value: < 0.00000000000000022
```

P-value gives the probability that, if the null were true, we would receive a result as extreme as this.

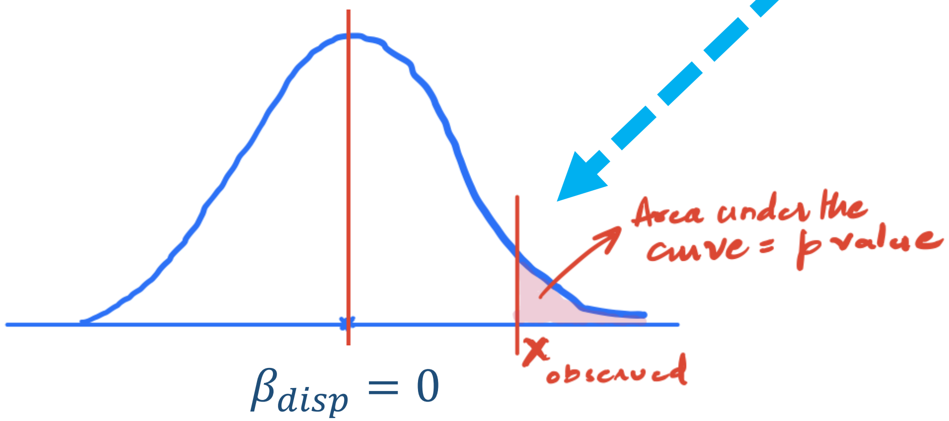
P-value for β_{disp} of 0.00205 say – assuming the null hypothesis of $\beta_{disp} = 0$ (flat slope) is actually true – we would see a coefficient as extreme as $\beta_{disp} = -1.95$ 0.02% of the time.

We either **reject the null hypothesis** or **fail to reject the null hypothesis**

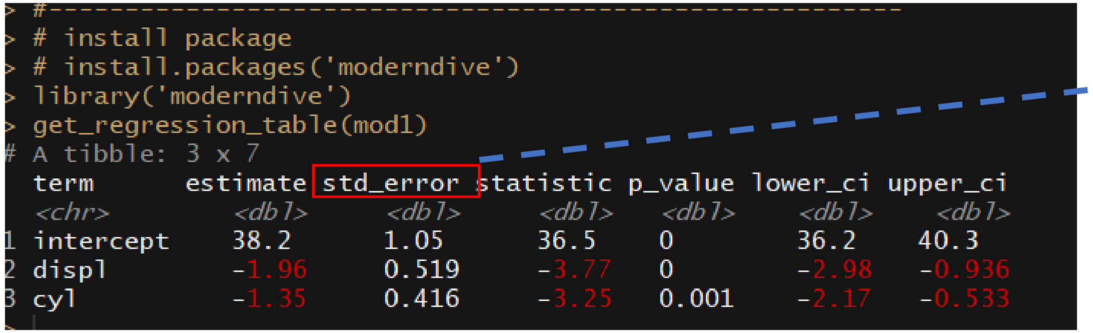
Based on a chosen critical value of alpha **(e.g. alpha = 0.05)**

~= incorrectly reject the null hypothesis 5% of the time even though null is true

This approach performs poorly in high dimensions (100s or 1000s of variables!) for reasons I hope become clear soon.

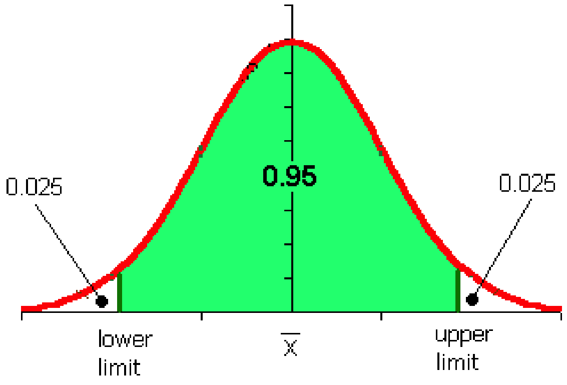


What About Standard Error, T-Statistic and Confidence Interval?



```
> #------------------------------------------------------
> # install package
> # install.packages('moderndive')
> library('moderndive')
> get_regression_table(mod1)
# A tibble: 3 x 7
  term      estimate std_error statistic p_value lower_ci upper_ci
  <chr>        <dbl>     <dbl>     <dbl>   <dbl>    <dbl>    <dbl>
1 intercept    38.2      1.05      36.5    0        36.2    40.3  
2 displ        -1.96     0.519     -3.77   0        -2.98   -0.936
3 cyl          -1.35     0.416     -3.25   0.001    -2.17   -0.533
```

- **Standard error** tells us the estimated standard deviation of the coefficient (the amount it varies across cases)
- ~= Measure of precision of estimate of coefficient
- Smaller SE relative to coefficient = more precise



$$t = \frac{\hat{\beta}_1 - 0}{SE(\hat{\beta}_1)}$$

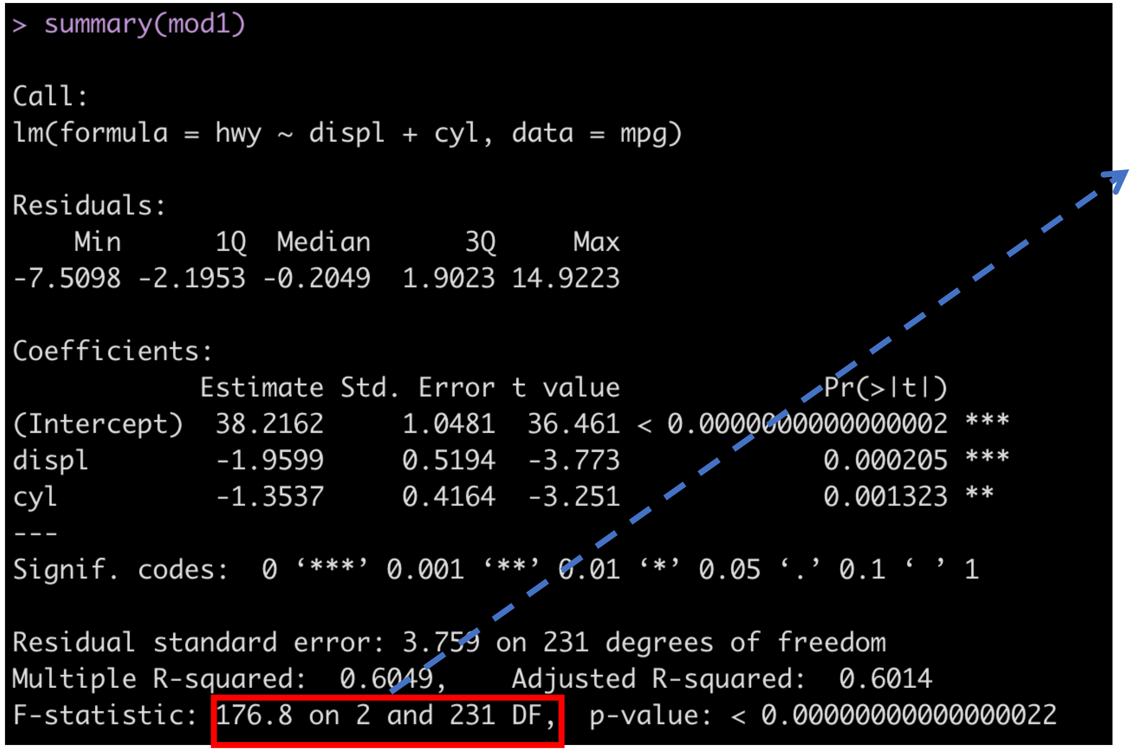
- **Confidence Interval** for $\hat{\beta}$ has a x% probability of containing the true value of β
- E.g. 95% confidence interval contains β with prob 95%

- **T-stat of coefficient** is a transformation of the estimated coefficient divided by the standard error (or precision of estimate)
- Large t-stat in abs value -> big effect size

Class 8: Outline

1. Discrete/Qualitative Independent Variables
2. Inference/Hypothesis Testing in Linear Models
3. **Model Evaluation**
   - Predicted/True Plots, RMSE, and R-Squared
4. Regression Lab 2

Measures of Overall Model Fit: F-Stat and R Squared



```
> summary(mod1)

Call:
lm(formula = hwy ~ displ + cyl, data = mpg)

Residuals:
    Min      1Q  Median      3Q     Max 
-7.5098 -2.1953 -0.2049  1.9023 14.9223 

Coefficients:
            Estimate Std. Error t value             Pr(>|t|)    
(Intercept)  38.2162     1.0481  36.461 < 0.0000000000000002 ***
displ        -1.9599     0.5194  -3.773             0.000205 ***
cyl          -1.3537     0.4164  -3.251             0.001323 ** 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.759 on 231 degrees of freedom
Multiple R-squared:  0.6049,	Adjusted R-squared:  0.6014 
F-statistic: 176.8 on 2 and 231 DF,  p-value: < 0.00000000000000022
```

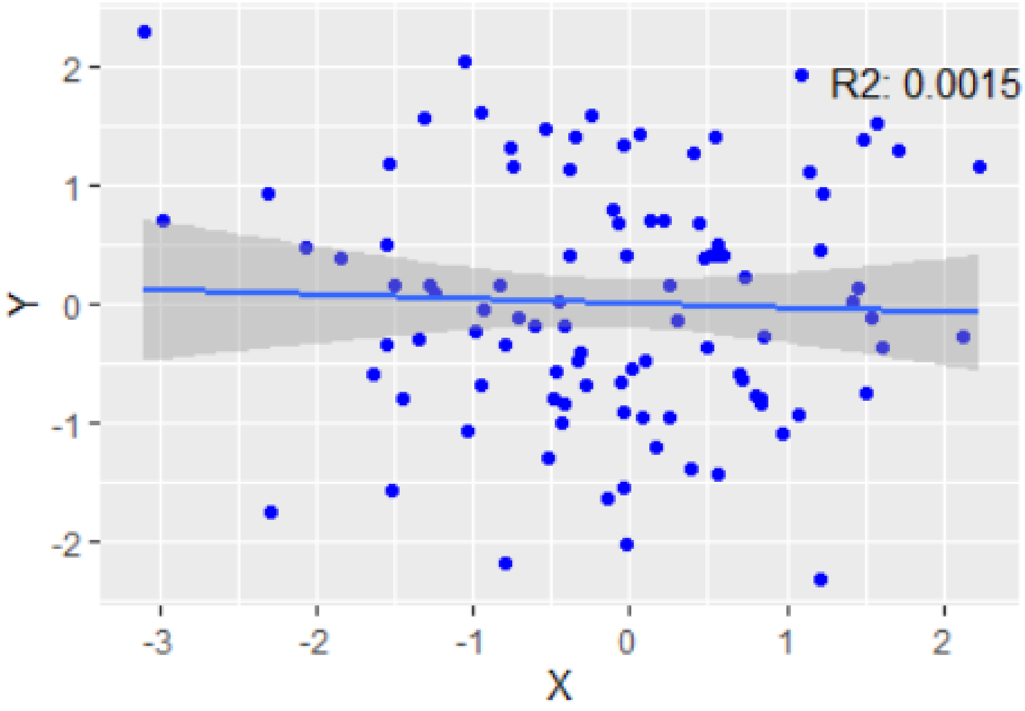
- **F-statistic** tells us whether all of the variables do better than a model with just an intercept
- Null = no effect of all variables except intercept.
- Almost always reject null. Outdated statistic.

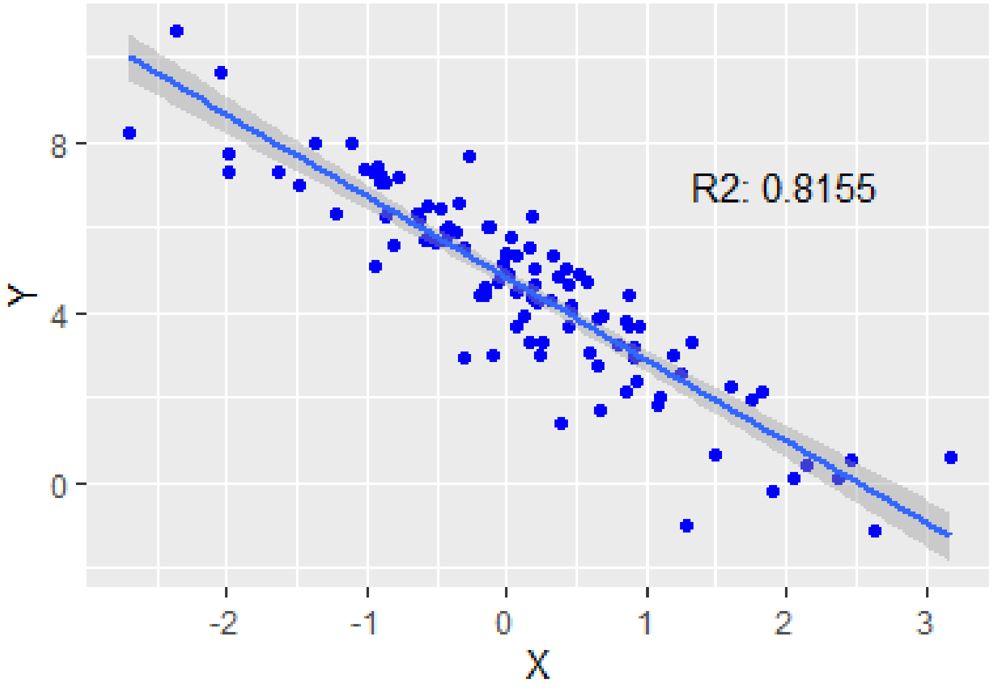
$$R^2 = \frac{TSS - RSS}{TSS} = \frac{TSS}{TSS} - \frac{RSS}{TSS} = 1 - \frac{RSS}{TSS}$$

$$RSS = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \qquad TSS = \sum_{i=1}^{n} (y_i - \bar{y})^2$$

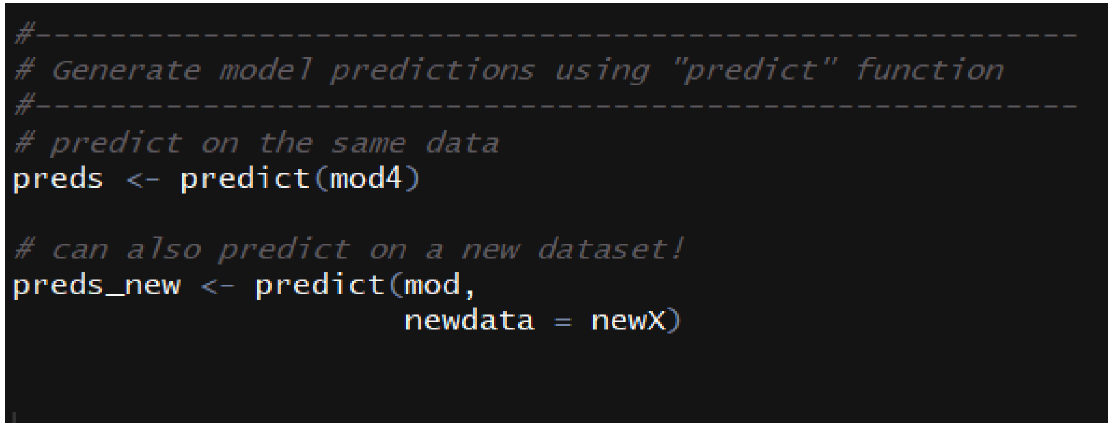
- R^2 **or “Coefficient of Determination”**
- Measures fraction model explains of variation in outcome (y)
- $R^2 \in [0,1]$.
  - 1 = Explain all variation in y
  - 0 = Explain none of the variation
- Higher R^2 better prediction model
- Use adjusted (adjusts for extra variables)

High Versus Low R2

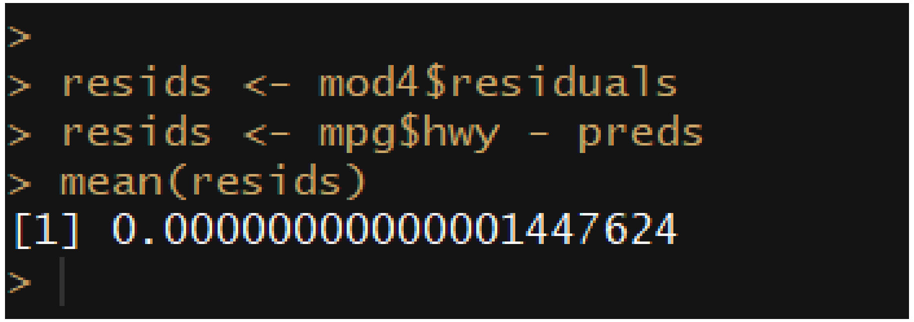




Predict() Function to Generate Model Predictions



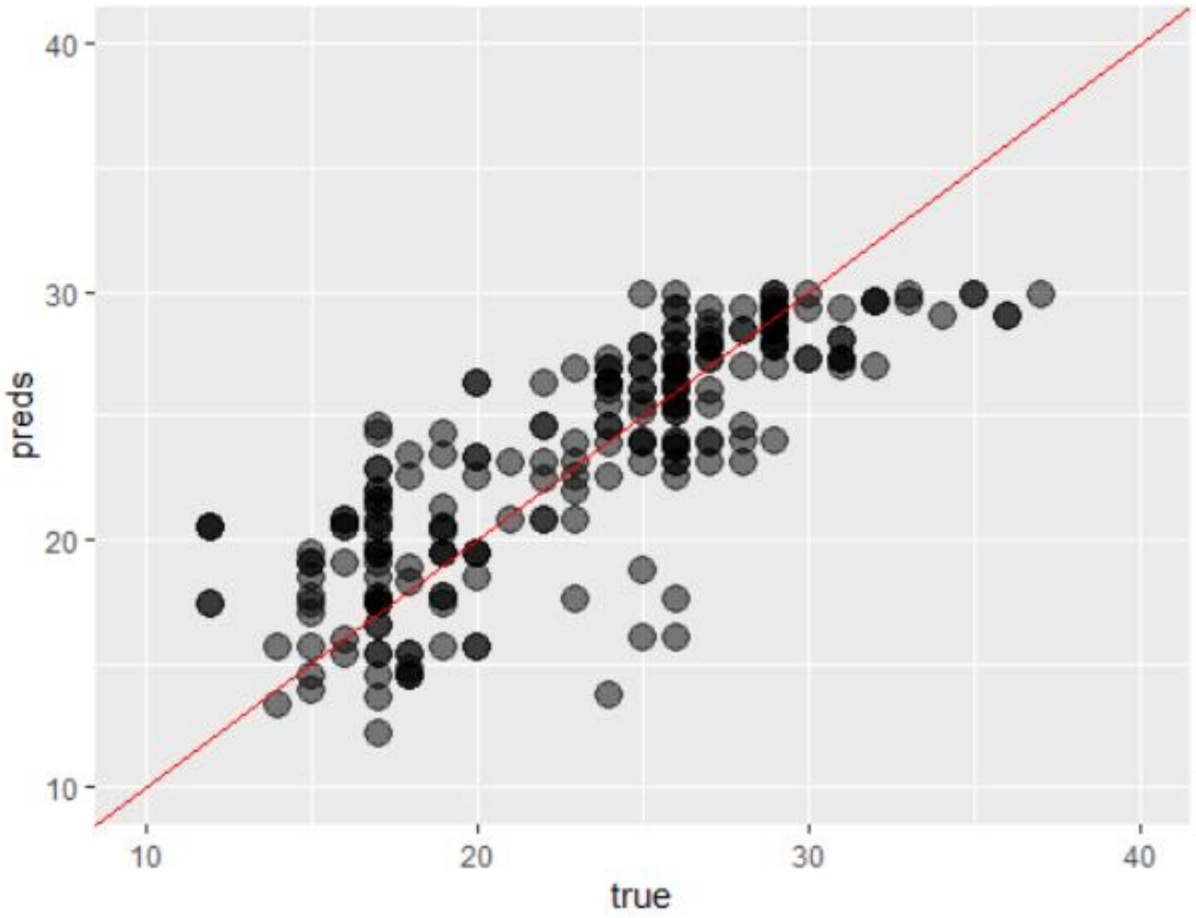
```
#--------------------------------------------------------
# Generate model predictions using "predict" function
#--------------------------------------------------------
# predict on the same data
preds <- predict(mod4)

# can also predict on a new dataset!
preds_new <- predict(mod,
                     newdata = newX)
```



```
>
> resids <- mod4$residuals
> resids <- mpg$hwy - preds
> mean(resids)
[1] 0.00000000000001447624
> 
```

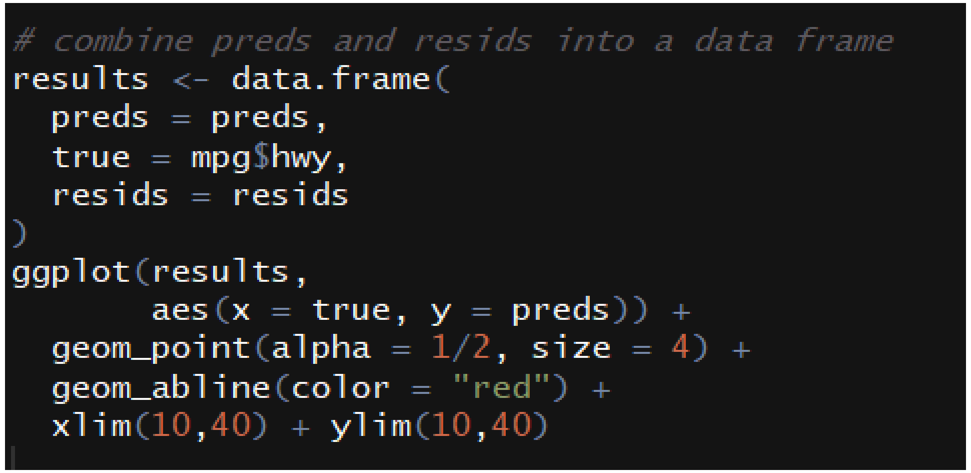
- Use the predict() function to generate model predictions using a trained/estimated model
- Note, we can also give it a new dataset (same Xs) with which to generate new predictions
- To generate residuals (true – predicted or $\hat{\epsilon}_i = y_i - \hat{y}_i$) some functions provide these for us, but we can calculate them ourselves
- Residuals (in-sample or on training set) are mean zero on average

Predicted True Plots



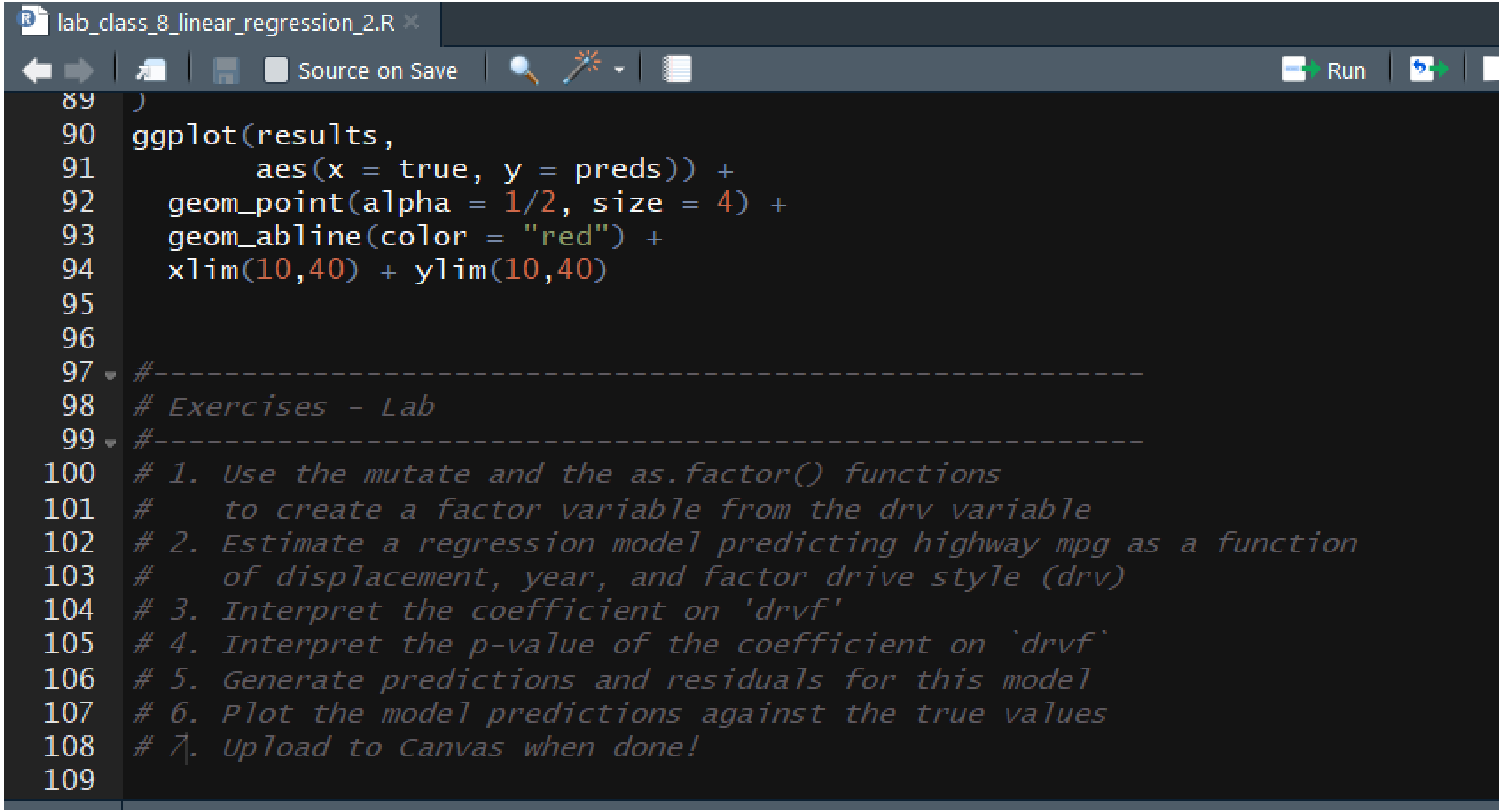
- Generally it’s a good idea to plot your predictions against the actual values to see how your model performs

- Red 45 degree line = if prediction were perfect



```
# combine preds and resids into a data frame
results <- data.frame(
  preds = preds,
  true = mpg$hwy,
  resids = resids
)
ggplot(results,
       aes(x = true, y = preds)) +
  geom_point(alpha = 1/2, size = 4) +
  geom_abline(color = "red") +
  xlim(10,40) + ylim(10,40)
```

Lab Time!



```
)
ggplot(results,
       aes(x = true, y = preds)) +
  geom_point(alpha = 1/2, size = 4) +
  geom_abline(color = "red") +
  xlim(10,40) + ylim(10,40)


#--------------------------------------------------------
# Exercises - Lab
#--------------------------------------------------------
# 1. Use the mutate and the as.factor() functions
#    to create a factor variable from the drv variable
# 2. Estimate a regression model predicting highway mpg as a function
#    of displacement, year, and factor drive style (drv)
# 3. Interpret the coefficient on 'drvf'
# 4. Interpret the p-value of the coefficient on `drvf`
# 5. Generate predictions and residuals for this model
# 6. Plot the model predictions against the true values
# 7. Upload to Canvas when done!

```

Class 8 Summary

- Factors are like strings, but better. They store strings as numbers, and contain a hash table/lookup code for each unique string
- We interpret binary dummy variables relative to each other, and specifically relative to the **left out category** or the **reference category** of the factor (e.g. first factor level for the factor)
- P-values tell us the likelihood we would see a result as extreme as the one seen assuming the null is true
  - If the null is not true all bets are off!
- Hypothesis testing for coefficients tests whether beta = 0 i.e. no slope!

- Standard Errors measures precision of our estimated coefficients
- R2 tells us how much variation in the outcome variable (y) we capture with our model!
- Use the predict() function to generate predictions
- Predicted/True plots show us how well our model does against the true values