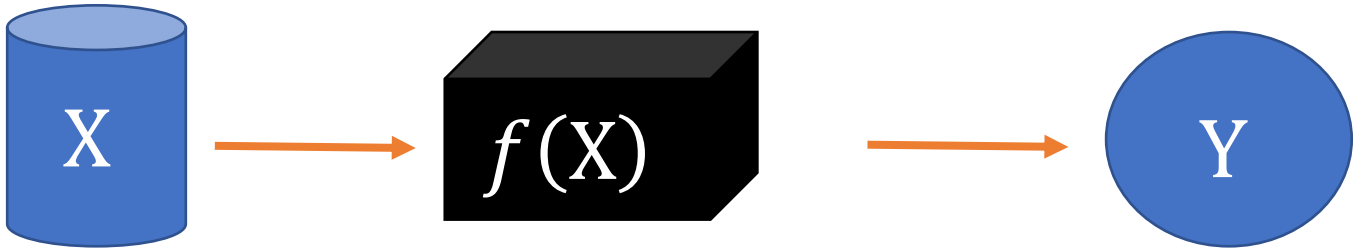
# 04. Linear Regression, Ridge and ElasticNet

Jonathan Hersh (Chapman University Argyros School of Business)

# Recipes for learning $f(\mathrm{X})$: Ordinary Linear Models (OLS) or "Least Squares"

$$\mathrm{Y} = f(\mathrm{X}) + \epsilon$$

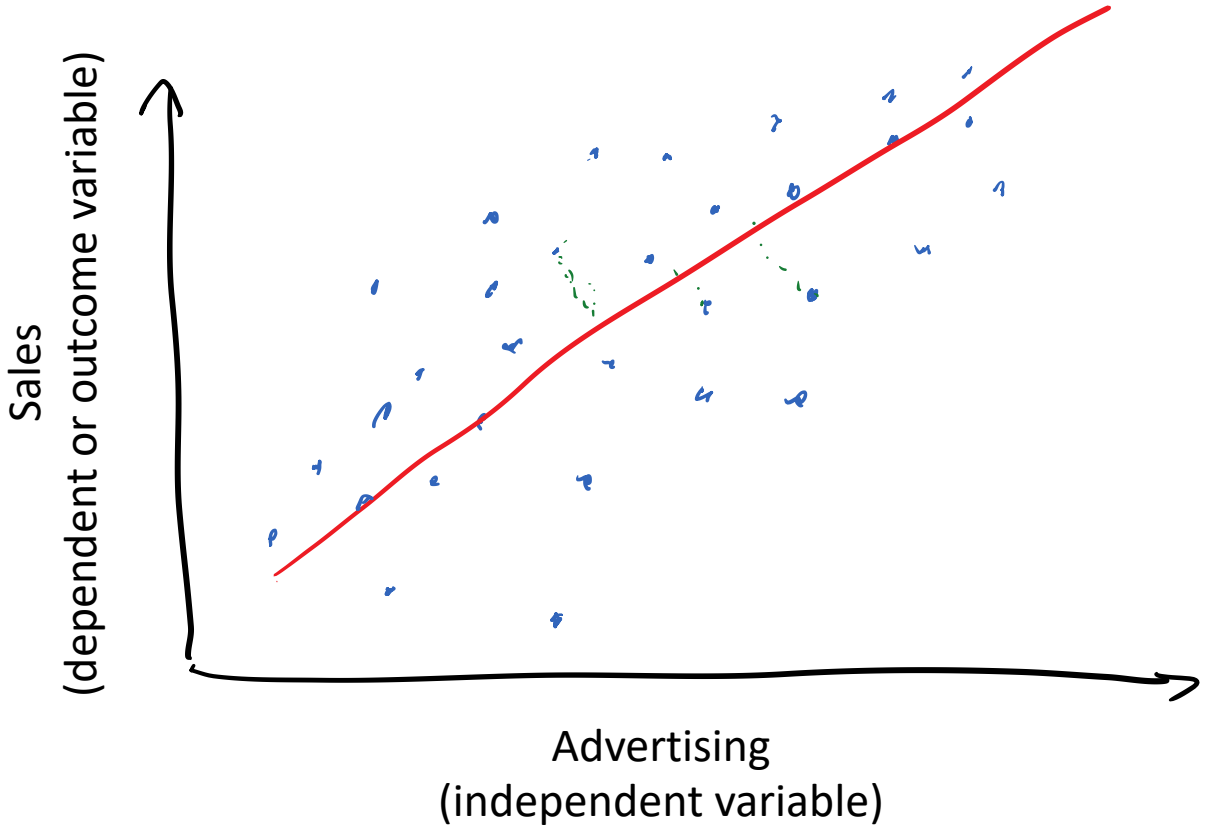



***Ordinary Linear Models***

$$f(X) = \beta_1 \cdot x_1 + \beta_2 \cdot x_2 + \cdots + \beta_3 \cdot x_3$$

OLS: Only allows linear combinations of Xs

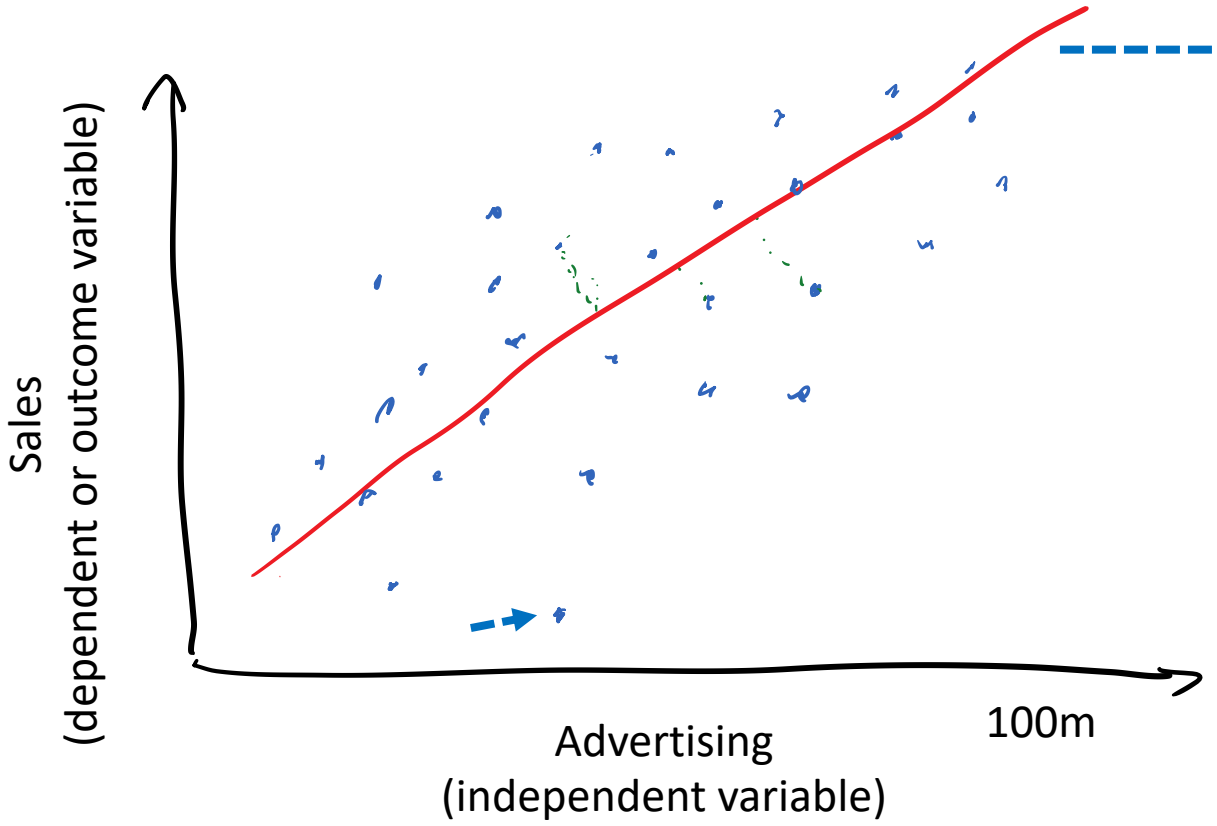
# What is Linear Regression?




**Regression:** statistical process of estimating relationship between an outcome and and one or more predictors or independent variables

**Linear Regression:** restricting relationship between predictors and outcome to be linear

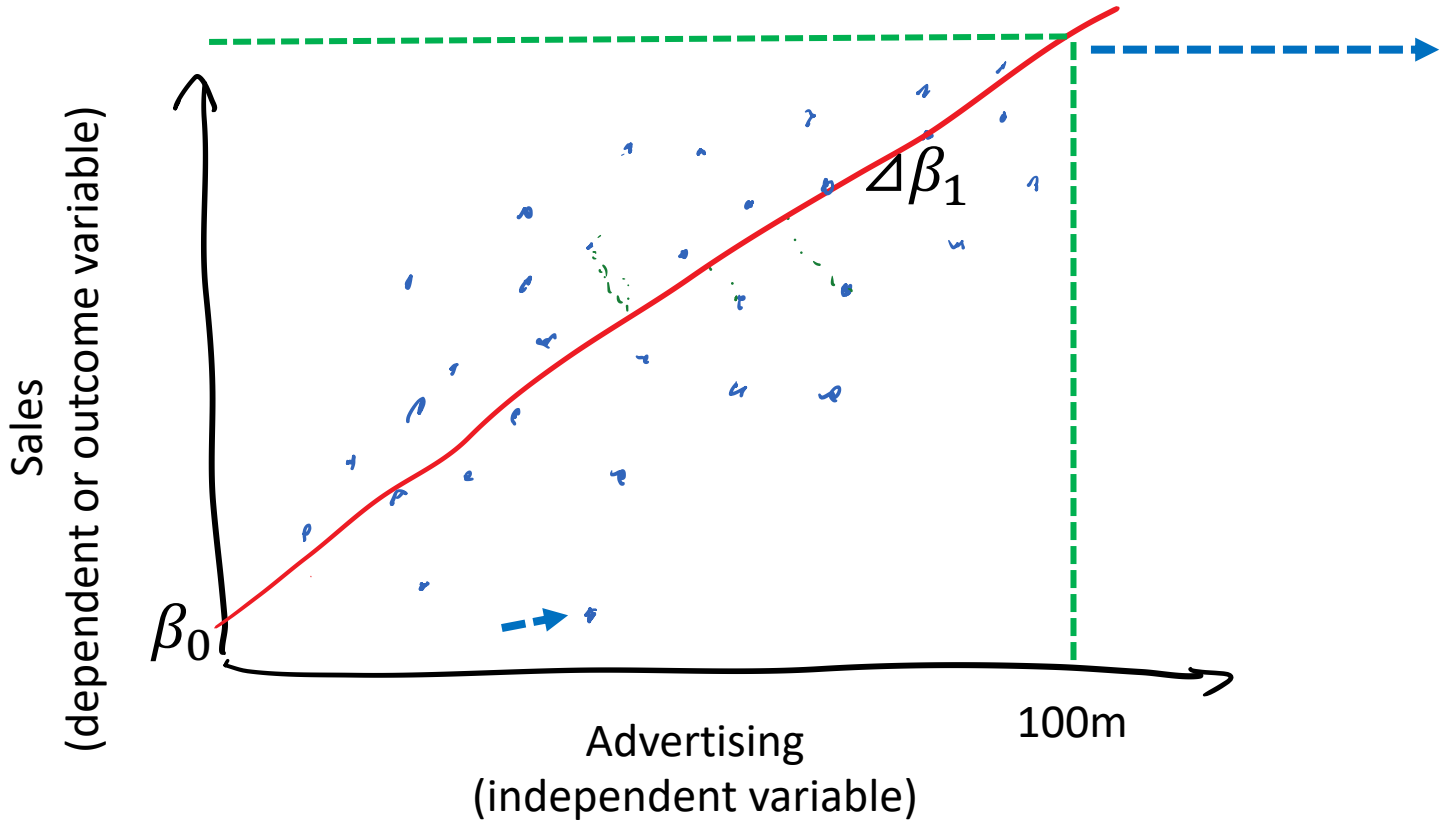
# Linear Regression Equation




$$y_i = \beta_0 + \beta_1 x_{Advertising} + \epsilon_i$$

Red line "explains" the data the best.

# Predictions from Linear Regression




$$y_i = \beta_0 + \beta_1 x_{Advertising} + \epsilon_i$$

Suppose we spend 100m on advertising?
What's our expected sales?

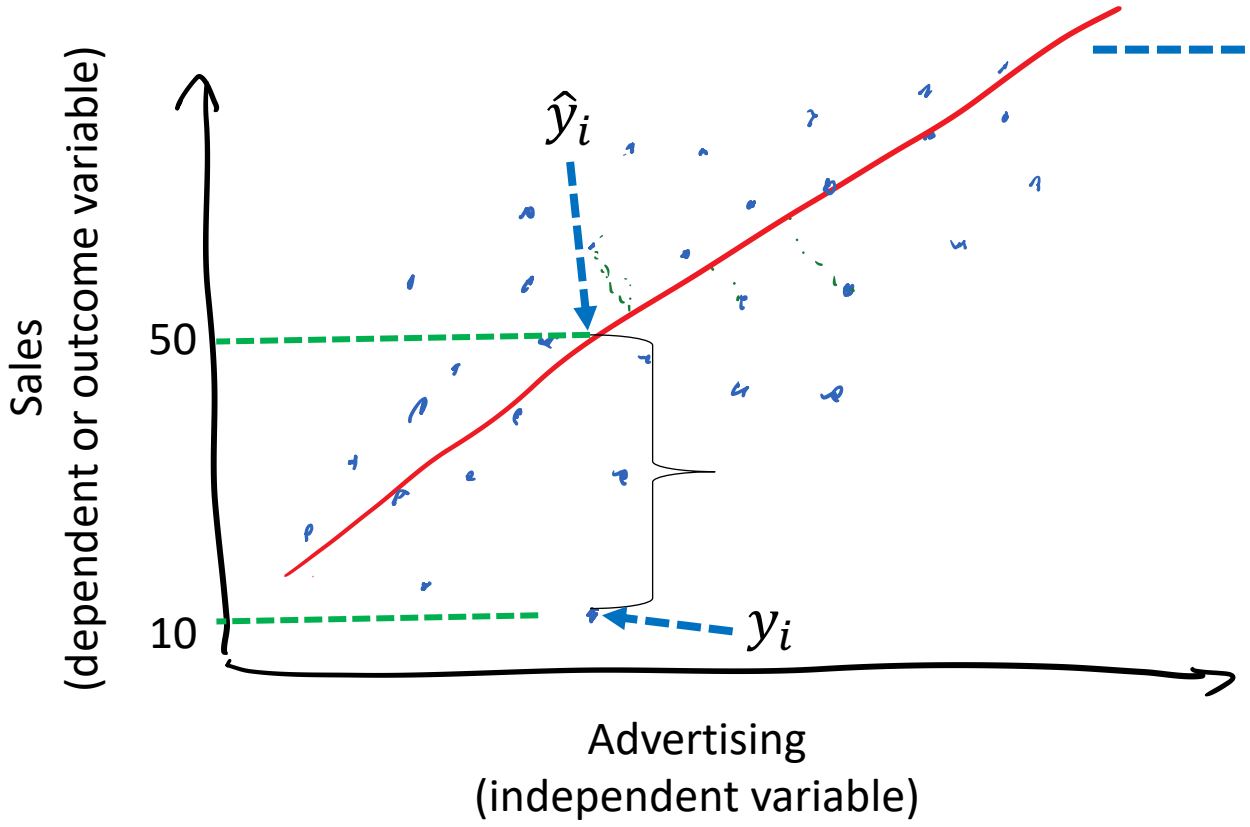
$$? = \widehat{\beta_0} + \widehat{\beta_1} 100m$$

$$? = 10 + 1 * 100$$

$$110 = 10 + 1 * 100$$

"Hat", e.g. $\widehat{\beta_0}$ , means we've estimated this relationship from data.

# Residuals: Measure Difference Between F(x) and Y




$$y_i = \beta_0 + \beta_1 x_{Advertising} + \epsilon_i$$

Errors: $\epsilon_i = y_i - \hat{y}_i$

Error: $\hat{\epsilon}_i = 10 - 50 = -40$

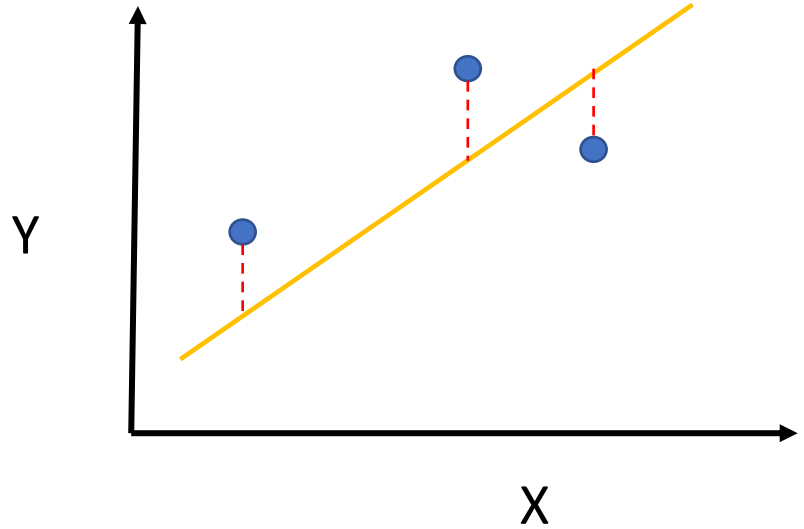
Errors are the difference between what we predict ($\hat{y}_i$) and the actual values ($y_i$).

# How Are Linear Regression Coefficients Chosen?

$$\hat{\beta}\ minimizes: \sum_{i=1}^{N} (y_i - \beta_0 - x_{i1}\beta_1 - x_{i2}\beta_2 - \cdots - x_{ik}\beta_k)^2$$

Sum through all observations

$$\epsilon_i = y_i - \hat{y}$$
$$= \mathrm{y_i} - \beta_0 - x_1\beta_1 - \cdots - x_2\beta_2$$



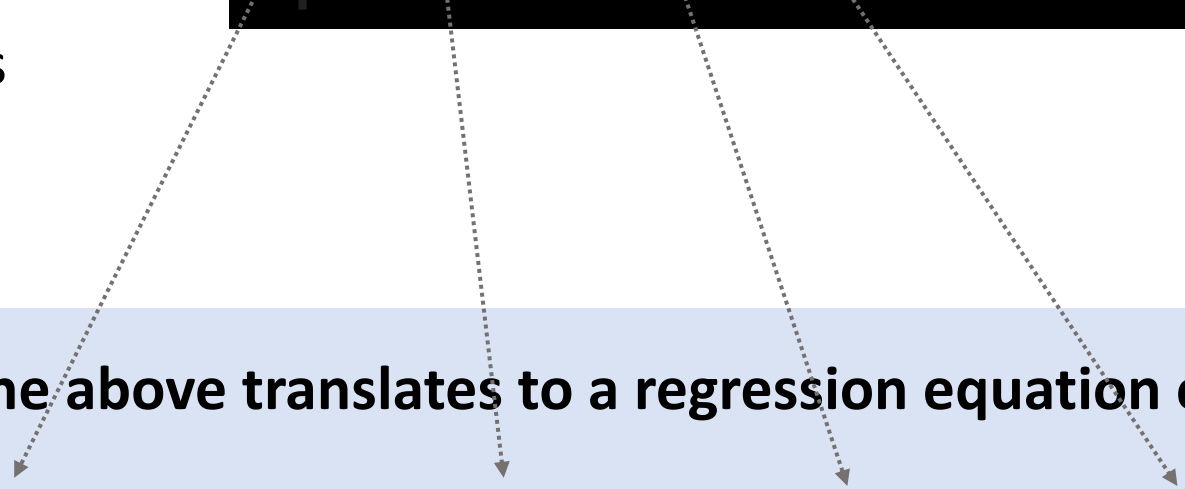

Least Squares Minimizes the **sum of squared residuals**

Visually, the slope ($\beta_1$) minimizes the difference between the points and the yellow line (red lines)

# Model Formulas in R

- Formulas in R start with the dependent variable on the left hand side (LHS)
- Followed by ”~” tilde
- Then all dependent variables separated by plus signs

```
>
>
>
> data(mpg)
> hwy ~ year + displ + cyl
hwy ~ year + displ + cyl
>
```



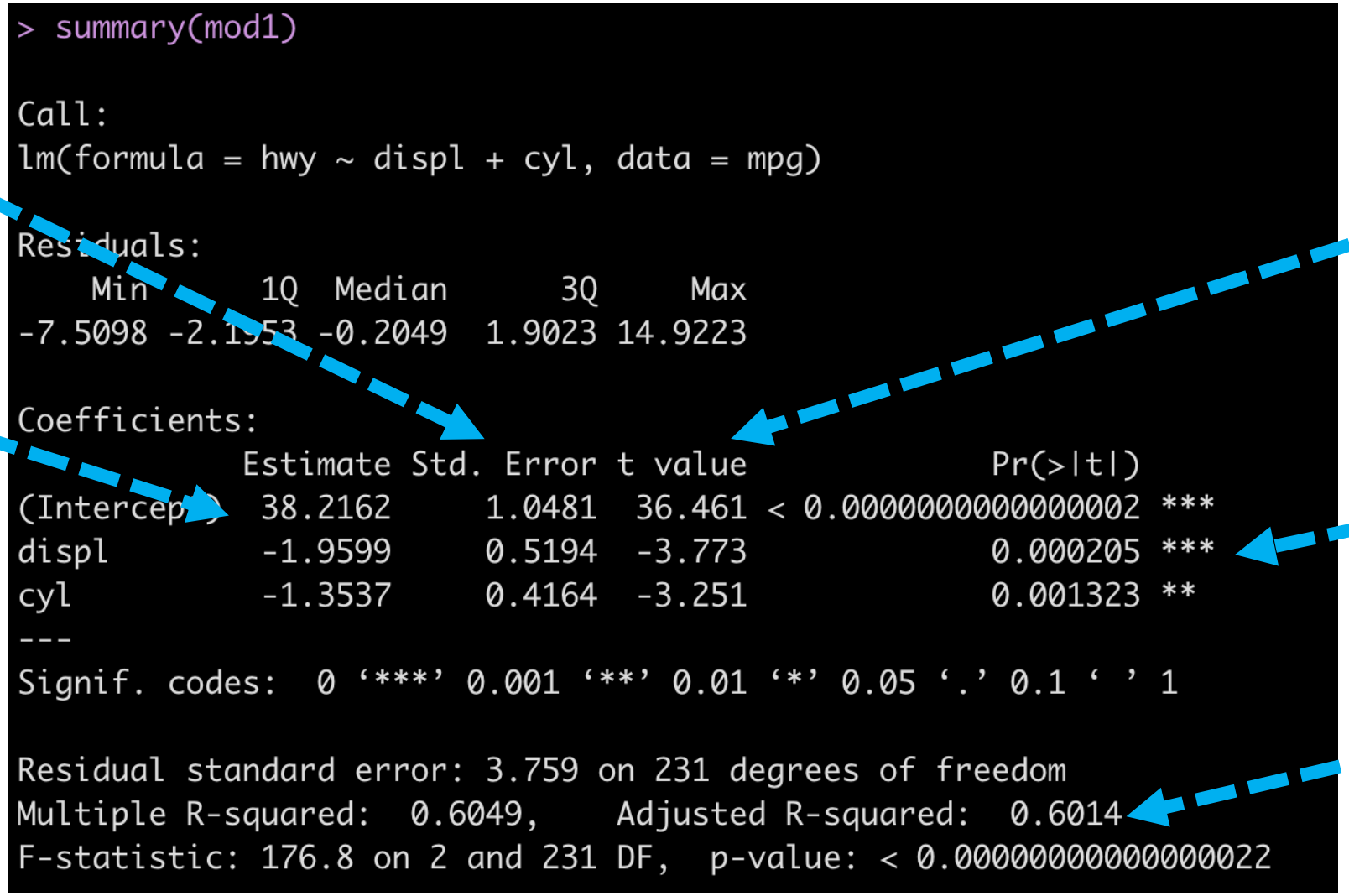
- **The above translates to a regression equation of:**
- $\boldsymbol{hwy = \beta_0 + \beta_1 \cdot year + \beta_2 \cdot displ + \beta_3 \cdot cyl}$

# Estimating Linear Models Using lm()

- Estimate a linear model using the 'lm()' function in R
- We must pass the dataset on which to estimate our model
- Then we store the regression model as 'mod1' (or whatever name you like
- Summary() outputs a summary of the estimated model

```
# estimate a linear model with displacement, and
# cycl on the RHS, and hwy as the
# development variable (LHS)
# Use the 'mpg' dataframe to estimate the model
# and store the regression equation as 'mod1'
mod1 <- lm(hwy ~ displ + cyl,
           data = mpg)

# print out a summary of the linear model
summary(mod1)

# or just view the whole "list" object of
# the model results
str(mod1)
```

# Viewing Regression Output Using "Summary"



```
> 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
```

Coefficient standard errors

Coefficient T-Statistic

Estimated Coefficients or "betas"

P-values for coefficients

Independent (X) variables

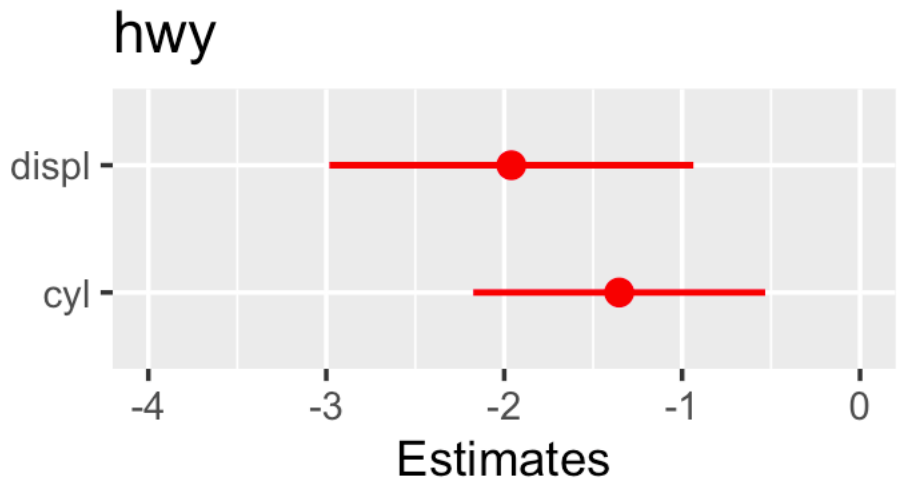
$R^2$, or "coefficient of determination" (model fit)

# Making “Pretty” Version of Regression Output Table

```
# install.packages('sjPlot')
library('sjPlot')
# output a prettier table of results
# looks very nice in RMarkdown!
tab_model(mod1)

# output a plot of regression coefficients
plot_model(mod1)

# output a table of nice coefficients
tidy(mod1)
```

```
# A tibble: 3 x 5
  term        estimate std.error statistic  p.value
  <chr>          <dbl>     <dbl>     <dbl>    <dbl>
1 (Intercept)    38.2      1.05      36.5  8.57e-98
2 displ          -1.96     0.519     -3.77 2.05e- 4
3 cyl            -1.35     0.416     -3.25 1.32e- 3
>
```

| | hwy | | |
|---|---|---|---|
| *Predictors* | *Estimates* | *CI* | *p* |
| (Intercept) | 38.22 | 36.15 – 40.28 | **<0.001** |
| displ | -1.96 | -2.98 – -0.94 | **<0.001** |
| cyl | -1.35 | -2.17 – -0.53 | **0.001** |
| Observations | 234 | | |
| $R^2$ / $R^2$ adjusted | 0.605 / 0.601 | | |

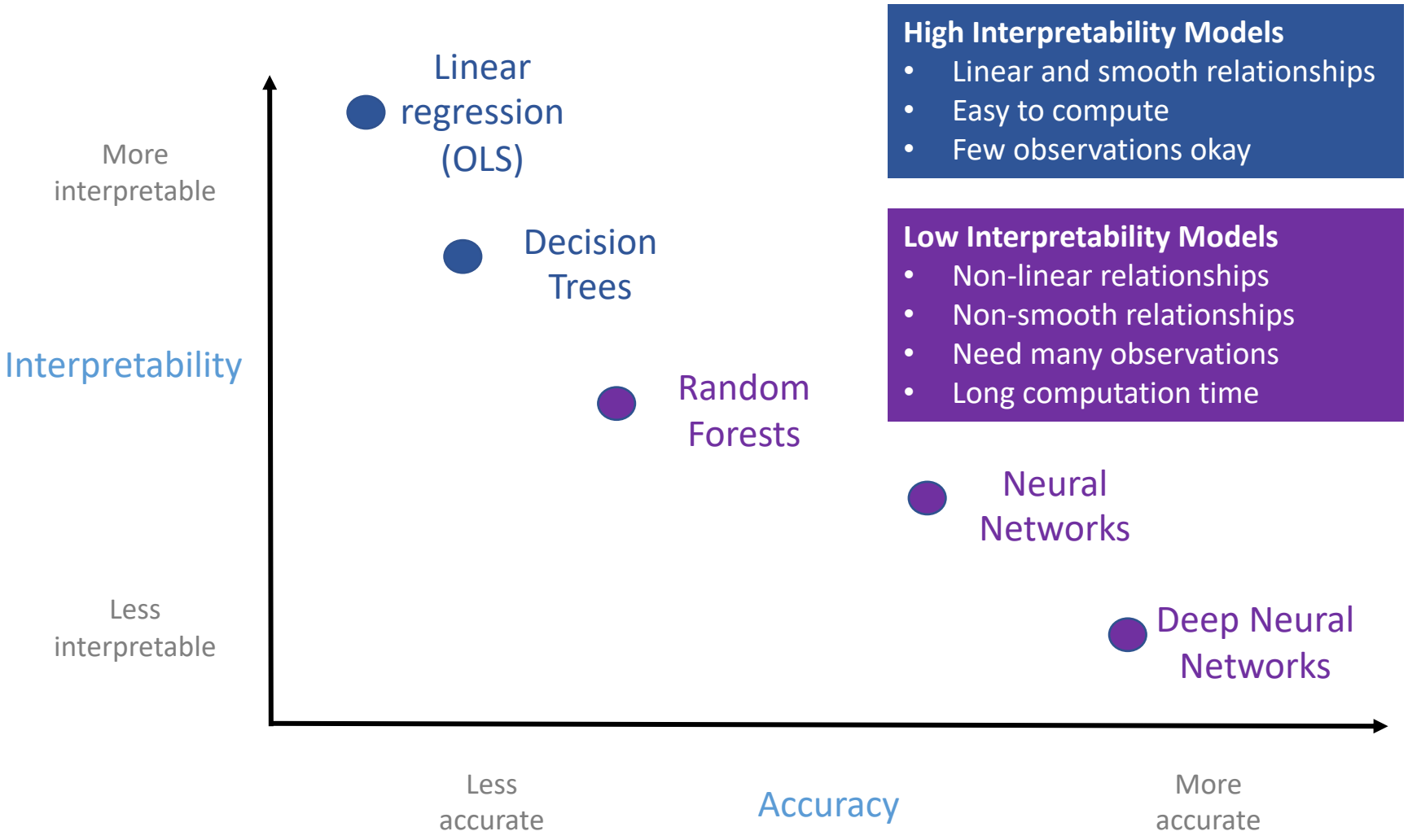


# Linear Model to Predict Bank Access

```
> bank_mod <- lm(any_bank_account ~ educ_head_of_hh + log_numHHmem + log_numChildren,
+             data = LFS_2019,
+             weight = Weight)
> summary(bank_mod)

Call:
lm(formula = any_bank_account ~ educ_head_of_hh + log_numHHmem +
    log_numChildren, data = LFS_2019, weights = Weight)

Weighted Residuals:
   Min     1Q Median     3Q    Max
-8.963 -3.058  1.403  2.250  6.095

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)     0.485943   0.045782  10.614   <2e-16 ***
educ_head_of_hh 0.020106   0.001873  10.733   <2e-16 ***
log_numHHmem    0.026172   0.037225   0.703    0.482
log_numChildren -0.036650   0.027518  -1.332    0.183
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.086 on 2167 degrees of freedom
Multiple R-squared:  0.05203,   Adjusted R-squared:  0.05072
F-statistic: 39.64 on 3 and 2167 DF,  p-value: < 2.2e-16
```

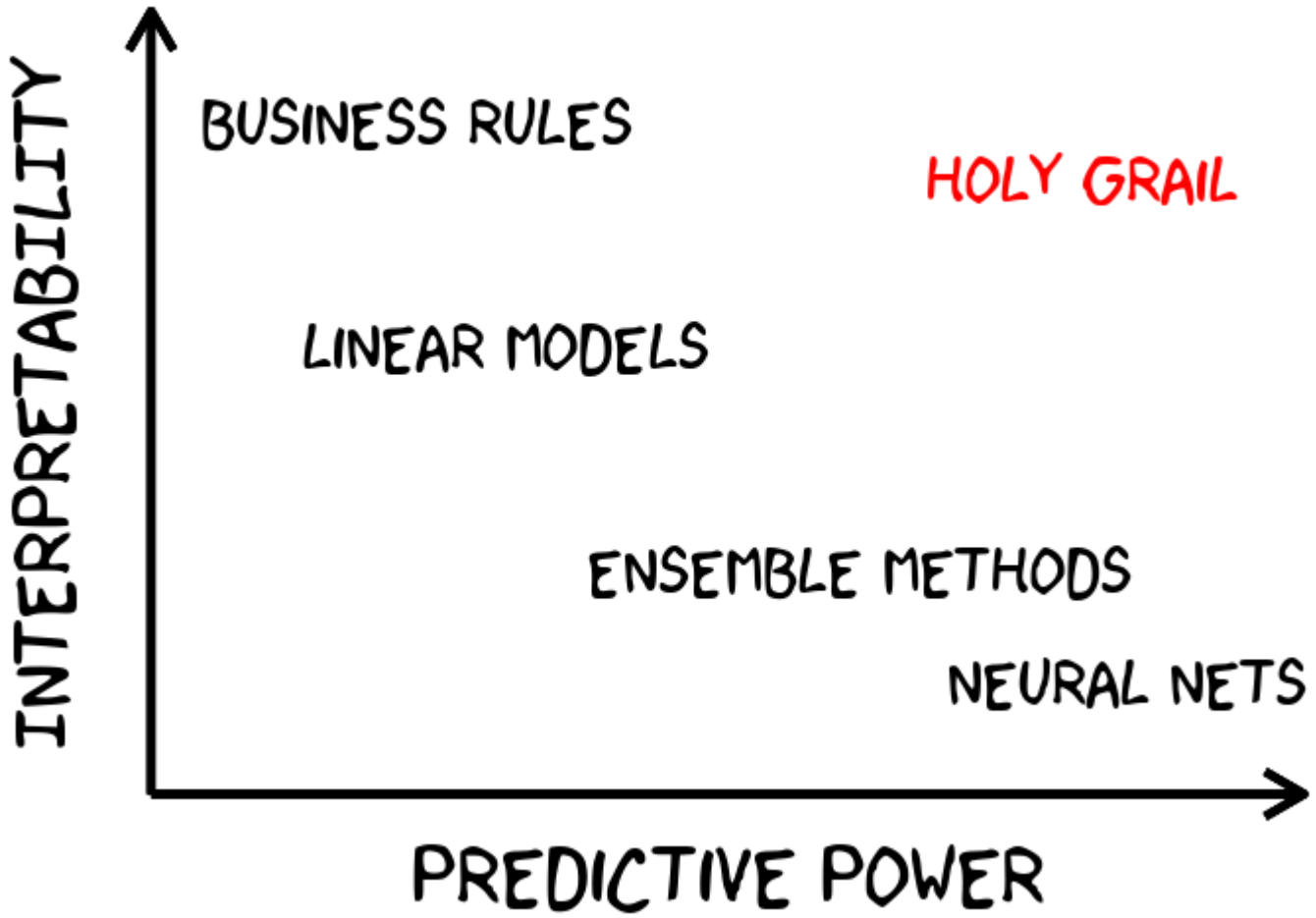
```
#-------------------------------------------------
# Exercises
#-------------------------------------------------

# 1. Estimate a linear model predicting any_bank_account as a function of
#    urban, tenureTypeOwn, floorMatPoor, toiletPoor, elecGrid, bedrooms, aircon
#    cellphones, computers, and numHHmem. Store this as bank_mod2

# 2. Estimate a linear model predicting borrowed_any as a function of the
#    same variables listed. Store this as borrowed_mod

# 3. Run the summary command over both models
```

# What Is Model Interpretability?




- **Model interpretability:**
  - "the degree to which a human can understand the cause of a decision" (Miller, 2017)
- The higher the interpretability, the easier it is for someone to comprehend why a decision has been made

# Of Course We Care About Both!



# Why Do We Care About Model Interpretability?

**1. Strengthen Trust and Transparency**

- People trust things they can understand, and don't trust things they don't (5G)

**2. Explain decisions**

- An interpretable model allows humans to understand the proposed decision, and diagnose and analyzed the solution

**3. Regulatory Requirements**

- Certain regulatory schemes (GDPR, Anti-Discrimination) require transparency.

**4. Improve the models**

- Interpretability ensures the model is right or wrong for the right reasons. Interpretability offers new feature engineering and helps debugging.

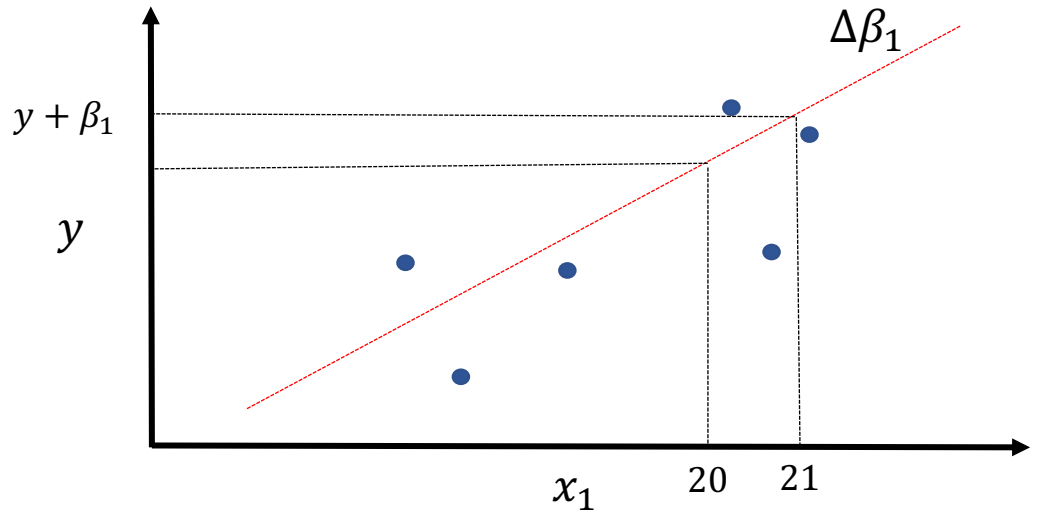
# Interpreting Linear Model Coefficients

- $\boldsymbol{\beta_1}$ mathematically explains how y changes when we increase $x_1$ by one unit
- Suppose we change $x_1$ by one unit of $x_1$. By how much does y change?
- Well, it changes by exactly $\boldsymbol{\beta_1}$

$$y = \beta_0 + \boldsymbol{\beta_1} \cdot x_1 + \cdots + \beta_3 \cdot x_3$$

$$?= \beta_0 + \boldsymbol{\beta_1} \cdot (x_1 + 1) + \cdots + \beta_3 \cdot x_3$$

$$y + \beta_1 = \beta_0 + \boldsymbol{\beta_1} \cdot (x_1 + 1) + \cdots + \beta_3 \cdot x_3$$



# Interpreting Linear Coefficients In Words

- **Communicating effect of coefficient**
  Increasing displacement by **one liter** (communicate units!) **decreases highway mile per gallon** (y variable) by **1.96 miles per gallon** holding fixed everything else
  - **X-variable**
  - **X-variable units**
  - **Direction (pos/neg)**
  - **Y-variable (outcome)**
  - **Estimated coefficient (magnitude)**
  - **Y-units**

```
> 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
```

**DO NOT JUST SAY WHEN X GOES UP Y GOES UP OR DOWN**
**THIS IS OBVIOUS AND YOU WILL GET FIRED**