STOR 320 Modeling II

Lecture 25

Yao Li

Department of Statistics and Operations Research

UNC Chapel Hill

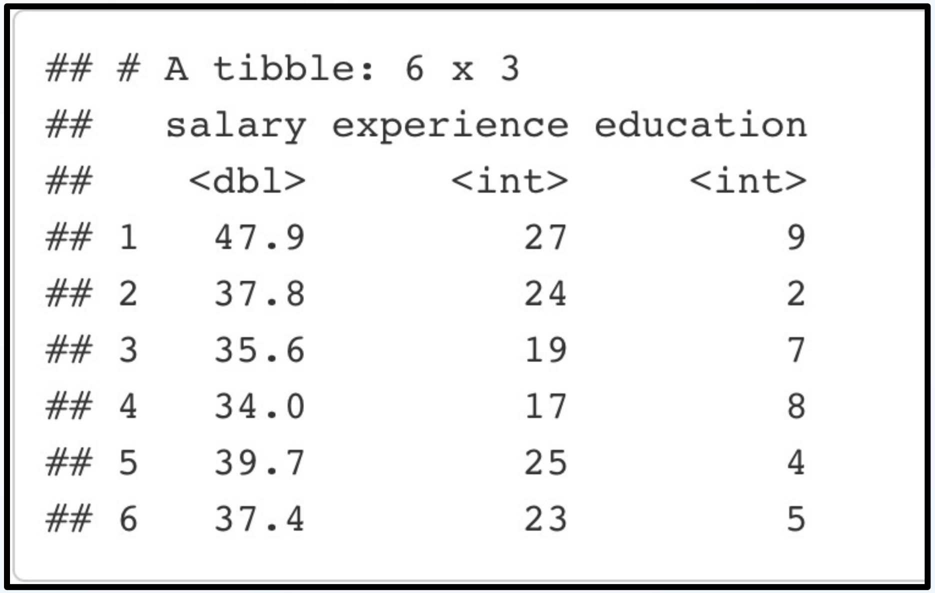
Example

- Modeling Real Experimental Data
  - Question: What Factors Improve Hourly Wage?
    - Hypothesis 1: Experience

    - Hypothesis 2: Education

Example

- Modeling Real Experimental Data
  - Data From 10,000 Individuals
    - X_1= Experience (# of Years)
    - X_2= Education (# of Years)
    - Y= Salary (dollars/hour)
    - Preview of Data:



```
## # A tibble: 6 x 3
##   salary experience education
##    <dbl>      <int>     <int>
## 1   47.9         27         9
## 2   37.8         24         2
## 3   35.6         19         7
## 4   34.0         17         8
## 5   39.7         25         4
## 6   37.4         23         5
```

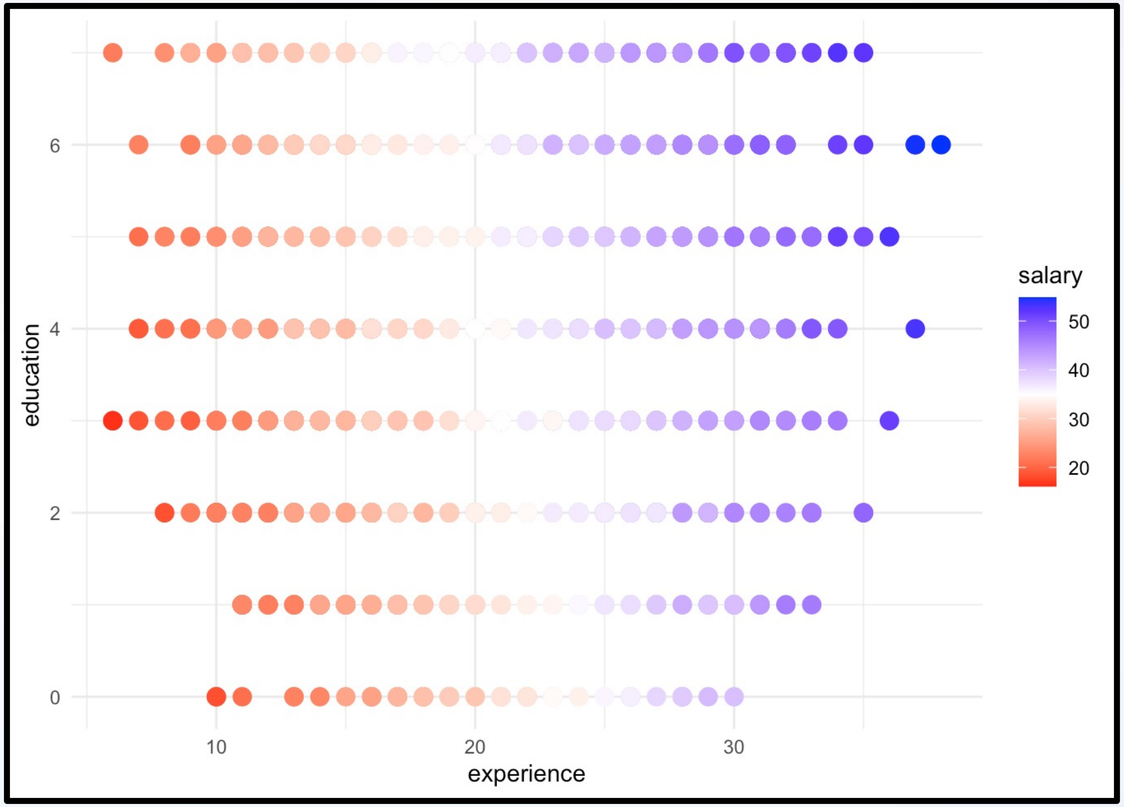
MODEL 2

- MODEL 2

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon$$

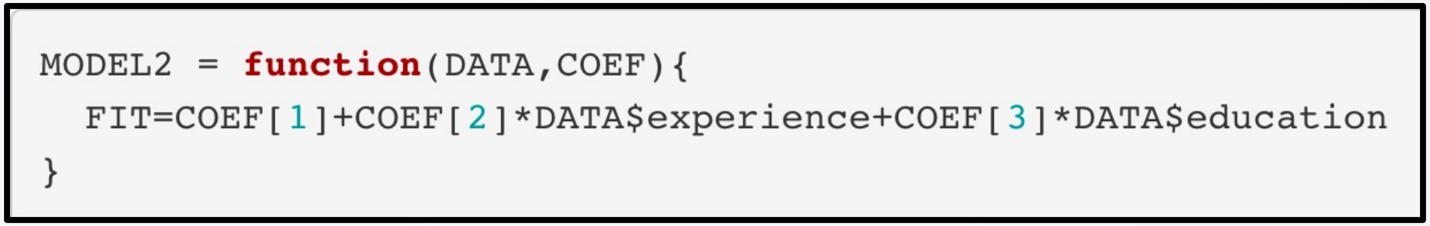
$$E(Y) = \beta_0 + \beta_1 X_1 + \beta_2 X_2$$

- Visualization of Relationship



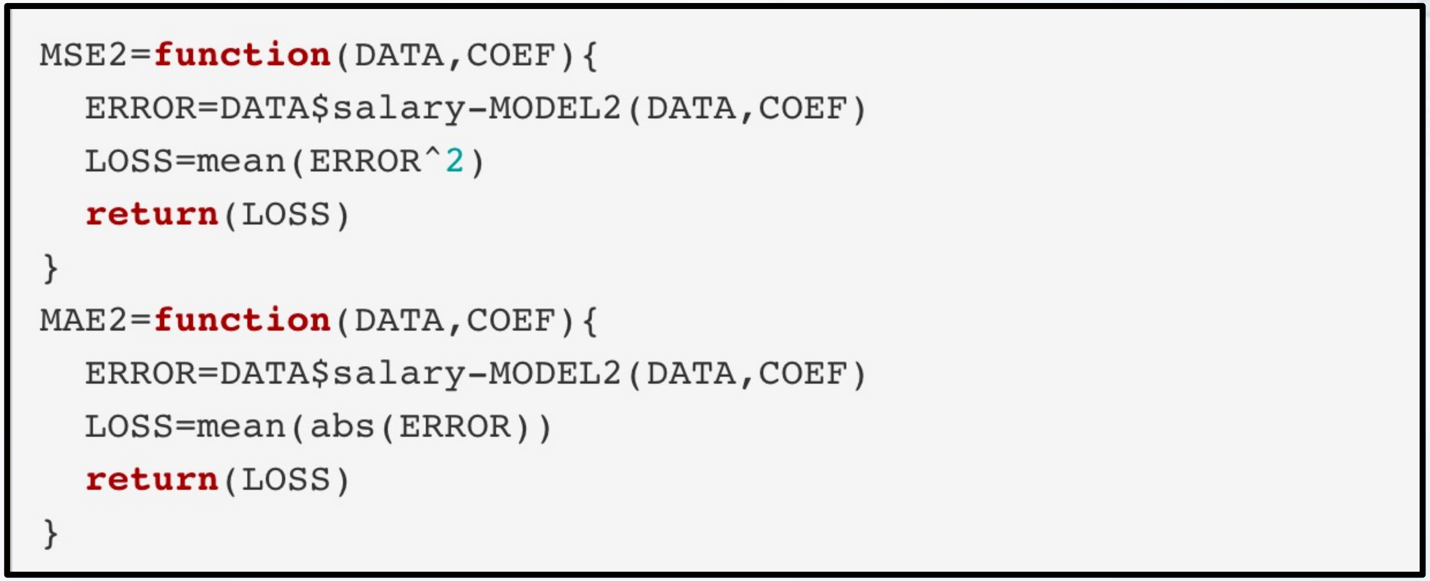
MODEL 2

- Function to Get Fitted Values



```
MODEL2 = function(DATA,COEF){
  FIT=COEF[1]+COEF[2]*DATA$experience+COEF[3]*DATA$education
}
```

- Functions to Evaluate Model



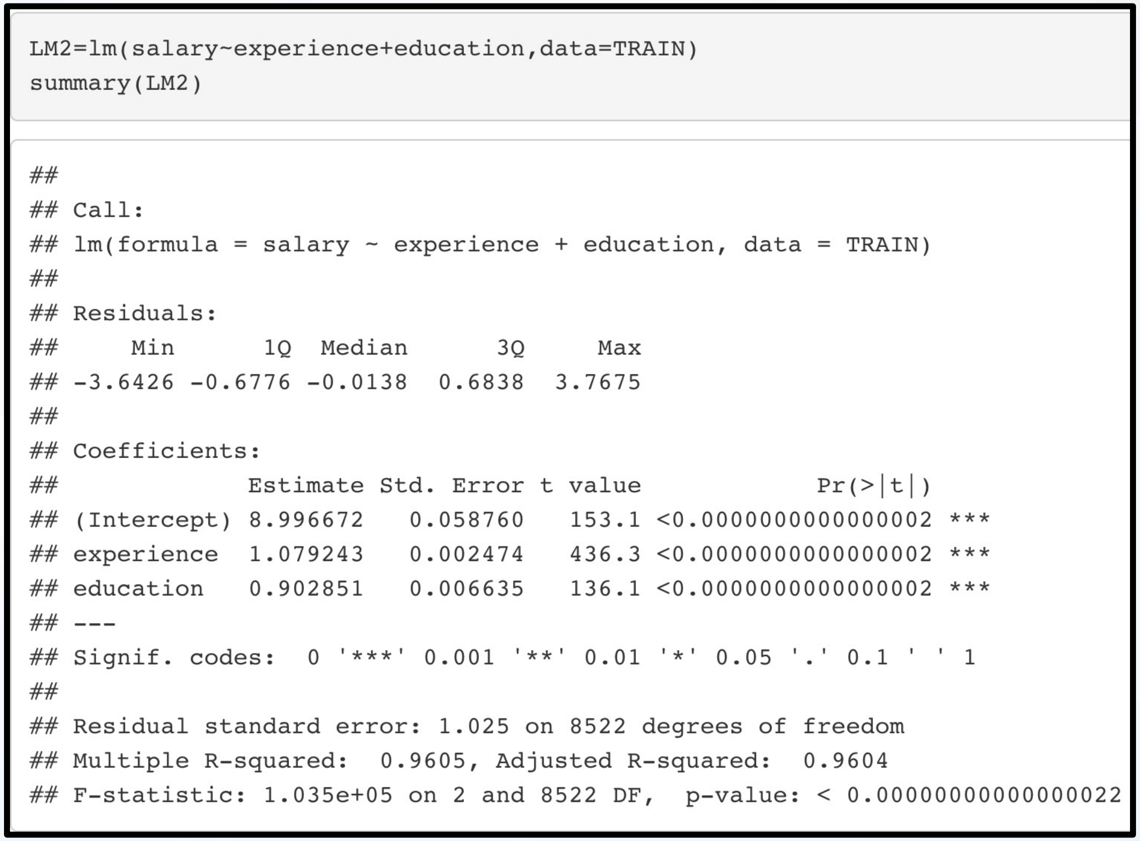
```
MSE2=function(DATA,COEF){
  ERROR=DATA$salary-MODEL2(DATA,COEF)
  LOSS=mean(ERROR^2)
  return(LOSS)
}
MAE2=function(DATA,COEF){
  ERROR=DATA$salary-MODEL2(DATA,COEF)
  LOSS=mean(abs(ERROR))
  return(LOSS)
}
```

Multiple Regression

- Use lm() with summary()
- Final MODEL 2

$$Y = 9 + 1.08X_1 + 0.9X_2 + \varepsilon$$

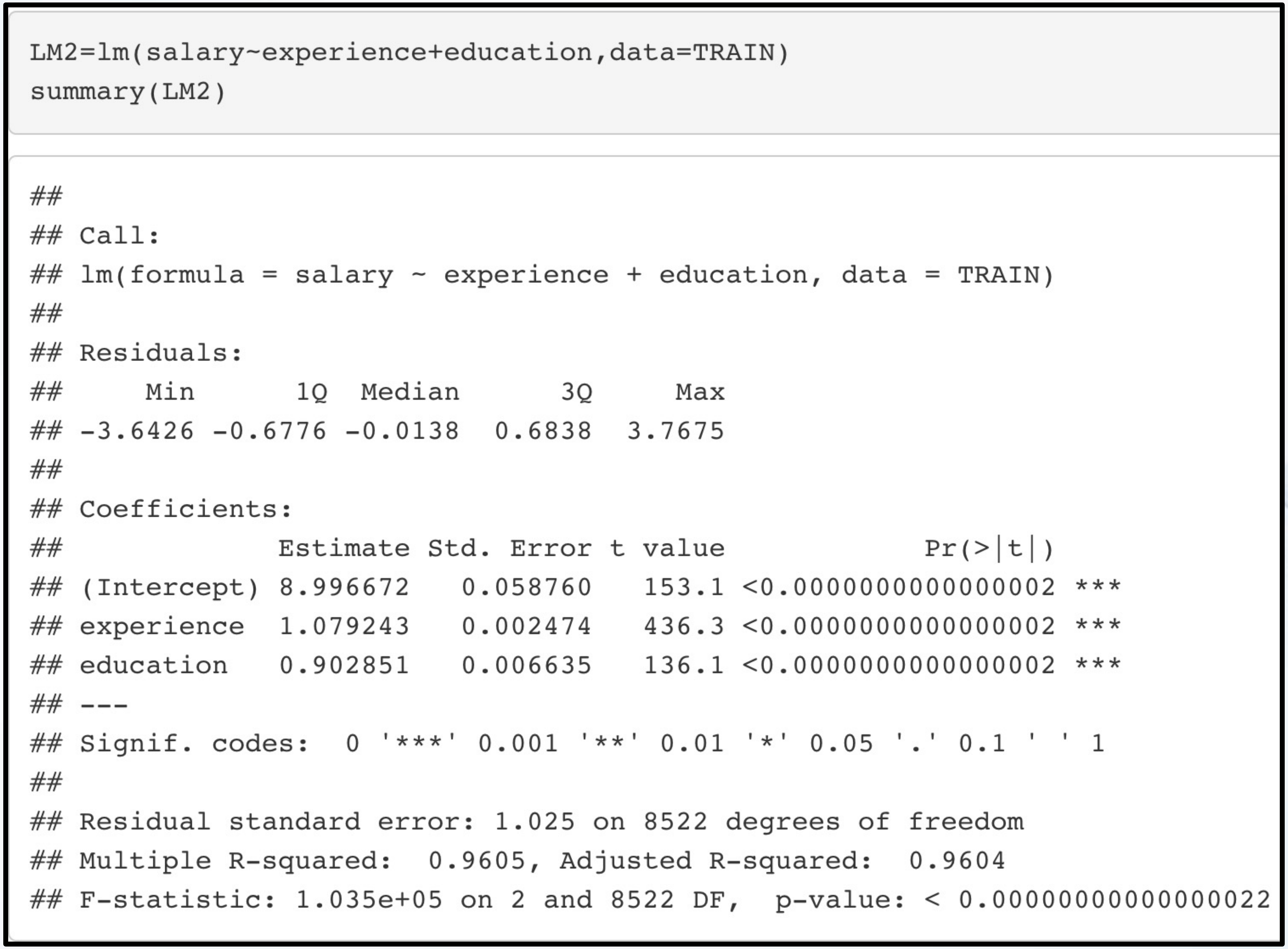
$$E(Y) = 9 + 1.08X_1 + 0.9X_2$$



```
LM2=lm(salary~experience+education,data=TRAIN)
summary(LM2)
```

```
##
## Call:
## lm(formula = salary ~ experience + education, data = TRAIN)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.6426 -0.6776 -0.0138  0.6838  3.7675
##
## Coefficients:
##             Estimate Std. Error t value            Pr(>|t|)
## (Intercept) 8.996672   0.058760   153.1 <0.0000000000000002 ***
## experience  1.079243   0.002474   436.3 <0.0000000000000002 ***
## education   0.902851   0.006635   136.1 <0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.025 on 8522 degrees of freedom
## Multiple R-squared:  0.9605, Adjusted R-squared:  0.9604
## F-statistic: 1.035e+05 on 2 and 8522 DF,  p-value: < 0.00000000000000022
```

Model Summary

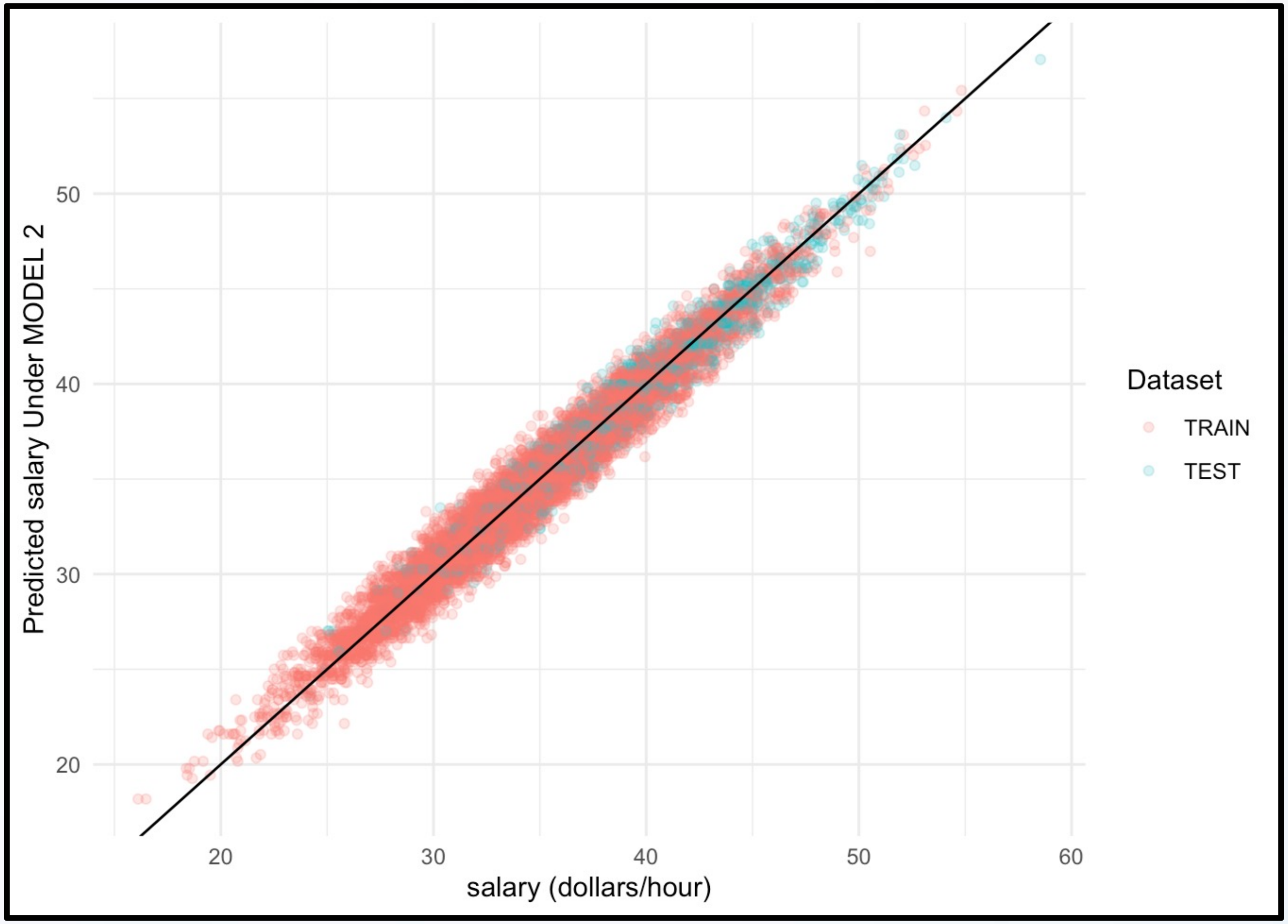


```
LM2=lm(salary~experience+education,data=TRAIN)
summary(LM2)
```

```
##
## Call:
## lm(formula = salary ~ experience + education, data = TRAIN)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.6426 -0.6776 -0.0138  0.6838  3.7675
##
## Coefficients:
##             Estimate Std. Error t value            Pr(>|t|)
## (Intercept) 8.996672   0.058760   153.1 <0.0000000000000002 ***
## experience  1.079243   0.002474   436.3 <0.0000000000000002 ***
## education   0.902851   0.006635   136.1 <0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.025 on 8522 degrees of freedom
## Multiple R-squared:  0.9605, Adjusted R-squared:  0.9604
## F-statistic: 1.035e+05 on 2 and 8522 DF,  p-value: < 0.00000000000000022
```

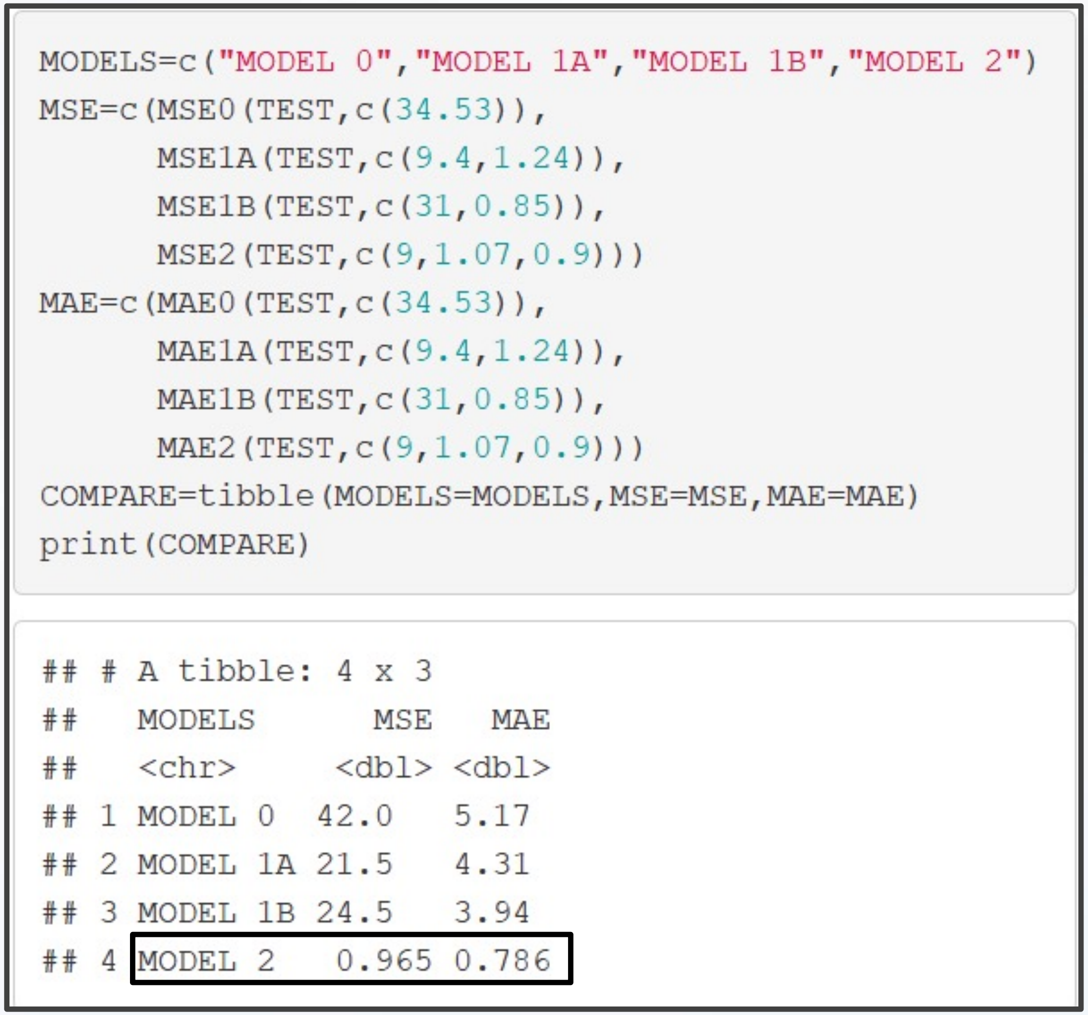
Visualization

- Comparing Predicted Values to Actual Values for MODEL 2



Model Evaluation

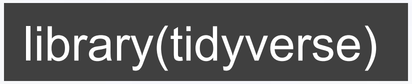
- Out-of-Sample Evaluation

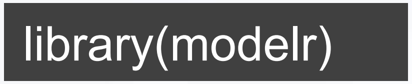


```
MODELS=c("MODEL 0","MODEL 1A","MODEL 1B","MODEL 2")
MSE=c(MSE0(TEST,c(34.53)),
      MSE1A(TEST,c(9.4,1.24)),
      MSE1B(TEST,c(31,0.85)),
      MSE2(TEST,c(9,1.07,0.9)))
MAE=c(MAE0(TEST,c(34.53)),
      MAE1A(TEST,c(9.4,1.24)),
      MAE1B(TEST,c(31,0.85)),
      MAE2(TEST,c(9,1.07,0.9)))
COMPARE=tibble(MODELS=MODELS,MSE=MSE,MAE=MAE)
print(COMPARE)
```

```
## # A tibble: 4 x 3
##   MODELS      MSE   MAE
##   <chr>     <dbl> <dbl>
## 1 MODEL 0  42.0   5.17
## 2 MODEL 1A 21.5   4.31
## 3 MODEL 1B 24.5   3.94
## 4 MODEL 2   0.965 0.786
```

Tutorial 11

- Instructions
  - Download Tutorial Zip
  - Unzip Folder
  - Required Packages
    - 

      ```
      library(tidyverse)
      ```
    - 

      ```
      library(modelr)
      ```
  - Open .Rmd File and Knit

- Daily Spanish River Data
  - W = Max Water Temperature
  - A = Max Air Temperature
  - L = River Identifier (31 Rivers)

Introduction

- Questions About RMarkdown
  - What Does the Following Code Do When Knitted?

```
`r length(unique(DATA$L))`
```

  - What Does the following Code Chunk Option Do When Knitted?

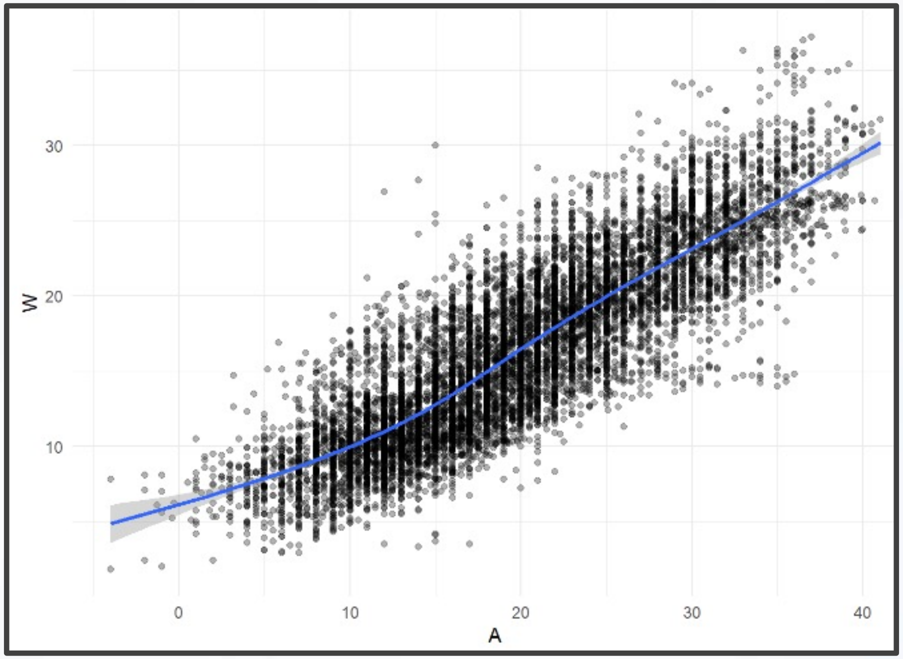
```
echo=F
```

Introduction

- Goal: Build a Model to Predict Max Water Temp Given Max Air Temp
  - What Do You Know About the Relationship of These Variables?
  - Who Would Care About this Relationship?
  - Why Would Someone Want to Predict the Max Water Temp?
  - Why Would this Model Be Useful?

Part 1: Examining the Relationship

- Run Chunk 1
  - What Do You Notice About the Overall Relationship?

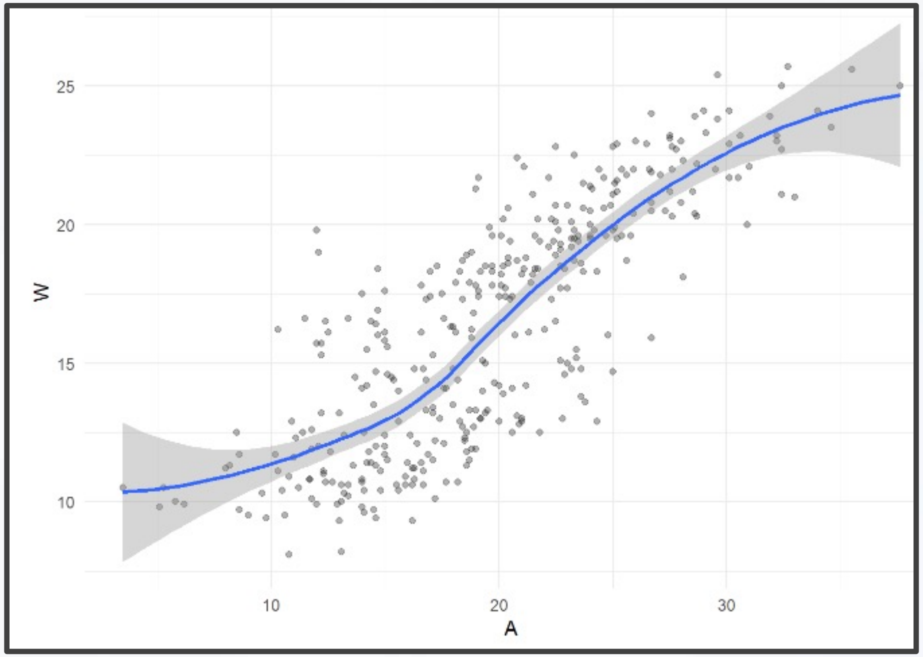


  - Do You Think This Relationship is the Same for All Locations?
  - Why?

message=F

Part 1: Examining the Relationship

- Run Chunk 2
  - Location is a Numeric Variable
  - What Do You Notice About the Relationship for L==103?



  - What do You Notice Now?

Part 1: Examining the Relationship

- Chunk 2 Modified
  - Modify Chunk 2 to Create a Function Called WAPlot.func With 1 Argument Location
  - Function Usage: You Specify the Location as an Integer and the Function Outputs a Figure of the Relationship
  - Use Your Function For Three Different Locations
  - Knit the Document to Observe and Compare

Part 1: Examining the Relationship

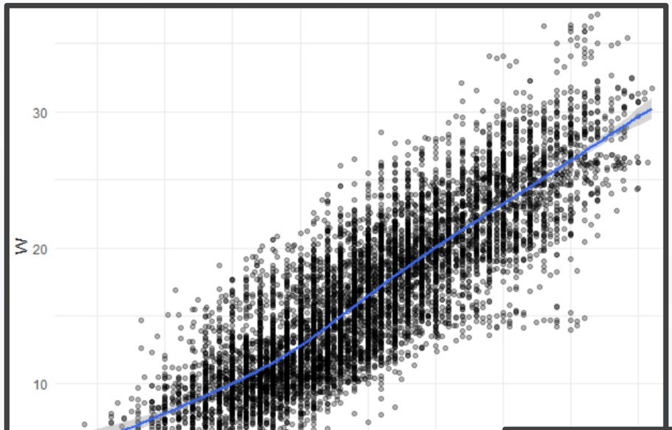
- Chunk 2 Discussion
  - What are the Differences in the Relationship Between W and A for the Various Locations?
  - Why do You Think These Differences Exist?
  - How do You Suggest We Handle the Differences?

Part 1: Examining the Relationship

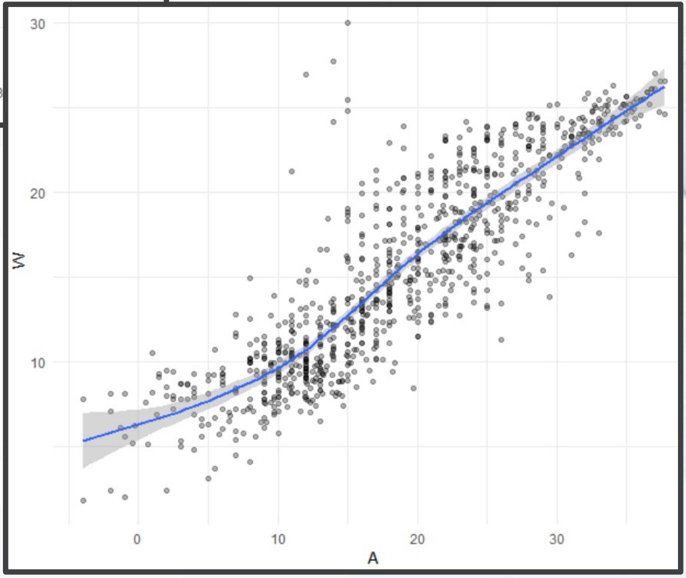
- Chunk 3
  - Randomly Samples 3 Locations
  - Plant Your Seed and Run Code
  - Usage:
    - anti_join()
    - semi_join()
  - Why Don’t We Handpick the Three Locations?

Part 1: Examining the Relationship

- Run Chunk 4
  - Train Plot



  - Test Plot



Part 2: Linear Model

- Linear Model
  - Simplest Relationship that is Easily Explained
  - For every 1 Degree Change in A, W changes by *b* Degrees
  - When A=0 Degrees, the Expected Water Temperature is *a* Degrees

Part 2: Linear Model

- Run Chunk 1
  - Fits Linear Model to Train Data
  - What is Your Intercept?
  - What is Your Slope?

- Run Chunk 2
  - Saves Predictions to Train/Test

```
add_predictions(MODEL,var=“NAME”)
```

- Run Chunk 3
  - Saves Residuals to Train/Test

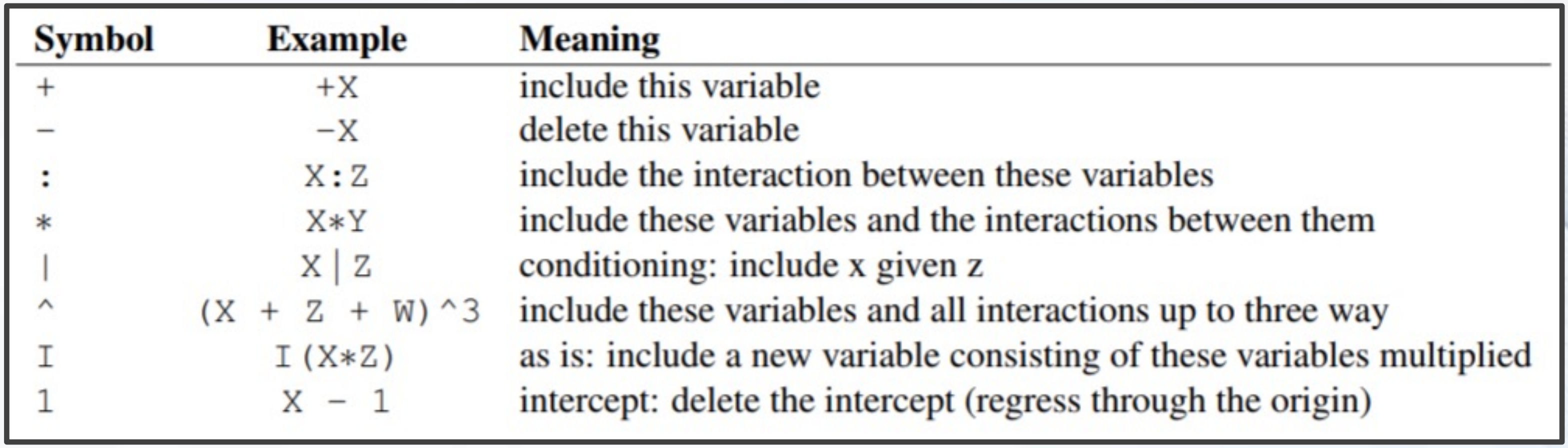
```
add_residuals(MODEL,var=“NAME”)
```

Part 3: Polynomial Model

- Polynomial Model
  - “Feature Engineering”
  - Generalized Additive Model
  - Geom_smooth() Fits a GAM when Fitting a Curve
  - Useful for Approximating Nonlinear Relationships
  - Dependent on Degree “k”
  - Goal: Choose Best “k”

Part 3: Polynomial Model

- Formula Object in R
  - Special Notation
  - Helpful Table:



Symbol	Example	Meaning
+	+X	include this variable
−	−X	delete this variable
:	X:Z	include the interaction between these variables
*	X*Y	include these variables and the interactions between them
|	X | Z	conditioning: include x given z
^	(X + Z + W)^3	include these variables and all interactions up to three way
I	I(X*Z)	as is: include a new variable consisting of these variables multiplied
1	X - 1	intercept: delete the intercept (regress through the origin)

  - We will Use the I() Function to Create New Variables Based Off Variables We Have

Part 3: Polynomial Model

- Run Chunk 1

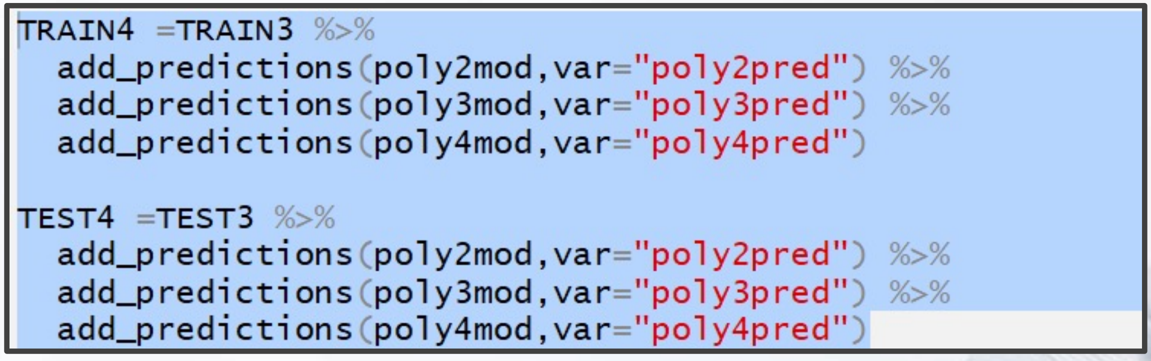
    - Fits 2nd Degree Polynomial
    - Fits 3rd Degree Polynomial
    - Fits 4th Degree Polynomial

- Run Chunk 2

    - Obtains Predictions Under the Different Polynomial Models

Part 3: Polynomial Model

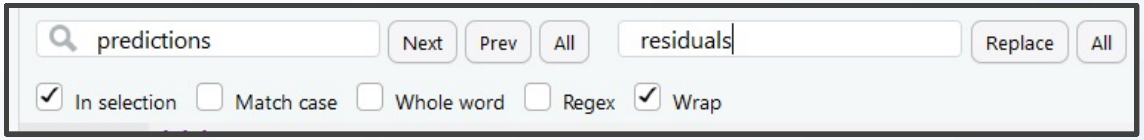
- Chunk 3
  - Code Needs Modification
  - Highlight Code



```
TRAIN4 =TRAIN3 %>%
  add_predictions(poly2mod,var="poly2pred") %>%
  add_predictions(poly3mod,var="poly3pred") %>%
  add_predictions(poly4mod,var="poly4pred")

TEST4 =TEST3 %>%
  add_predictions(poly2mod,var="poly2pred") %>%
  add_predictions(poly3mod,var="poly3pred") %>%
  add_predictions(poly4mod,var="poly4pred")
```

  - TRAIN3 -> TRAIN4 and etc.
  - Use Ctrl+F (Find and Replace)
    - ‘predictions’ -> ‘residuals’
    - ‘pred’ -> ‘res’



predictions residuals

☑ In selection □ Match case □ Whole word □ Regex ☑ Wrap

  - Run Chunk 3 After Modifying

Intermission

- Run Code Chunk

  - save.image() = Used to Save Workspace into .Rdata File

  - load() = Used to Load Workspace from .Rdata File

  - .Rdata = File Extension of R Workspace File (All Objects in Global Environment)