

Using GitHub and RStudio to Facilitate Authentic Learning Experiences in a Regression Analysis Course

Maria Tackett
Duke University

JSM 2019



maria.tackett@duke.edu



[@MT_statistics](https://twitter.com/MT_statistics)



Introduction

Assignments

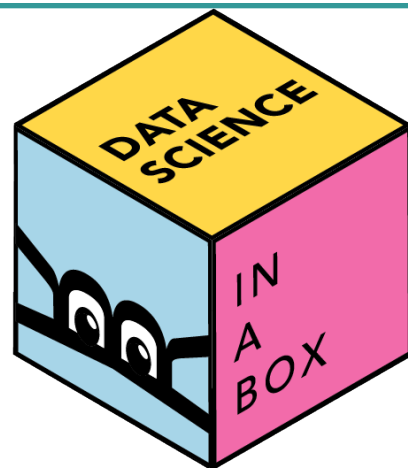
Challenges
+
Learning
Outcomes

Motivation

*“To be prepared for statistics and data science careers, **students need facility with professional statistical analysis software**, the ability to access and wrangle data in various ways, and the ability to perform algorithmic problem solving.”*

2014 ASA Curriculum Guidelines for Undergraduate Programs in Statistical Science

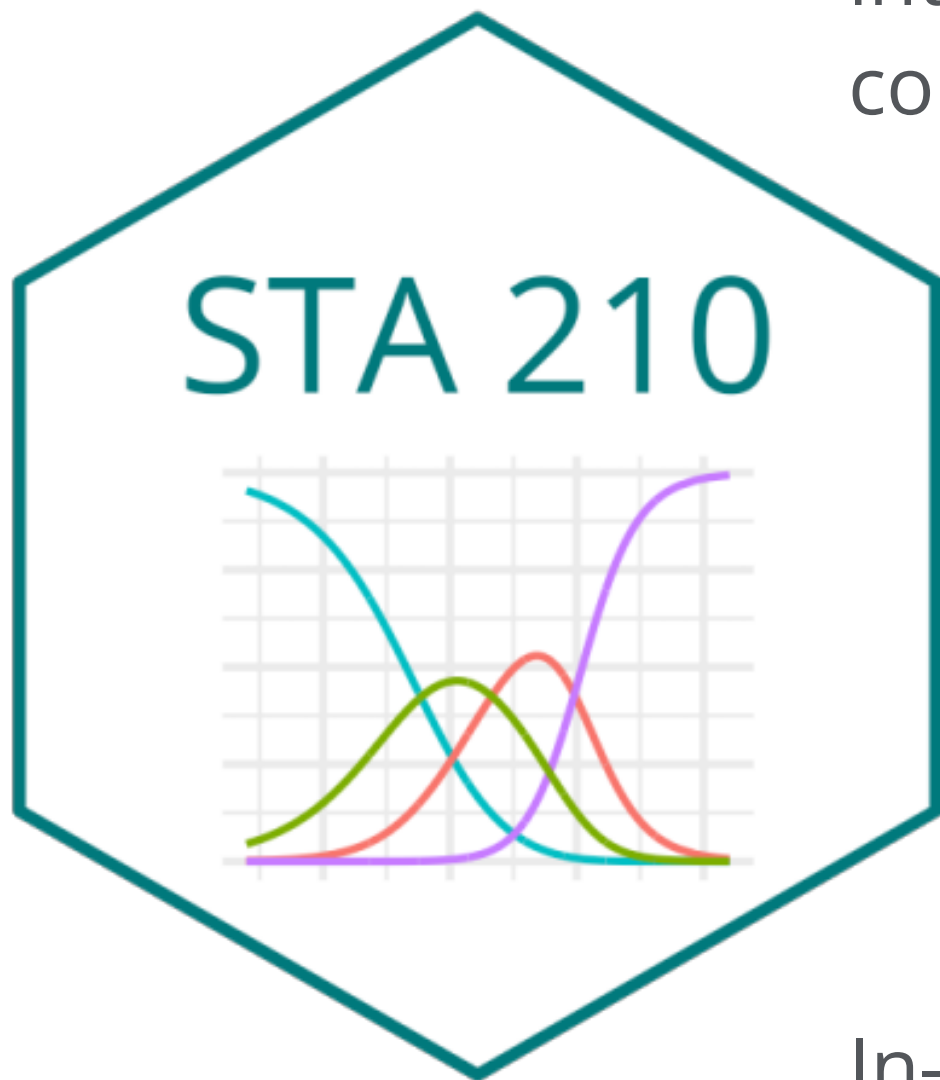
Innovations in
intro statistics
and data science
courses



Goal: Create learning experiences to continue cultivating these skills in intermediate courses

The Course

70 students who have taken introductory statistics or probability course



Applied course focusing on linear regression, analysis of variance and logistic regression

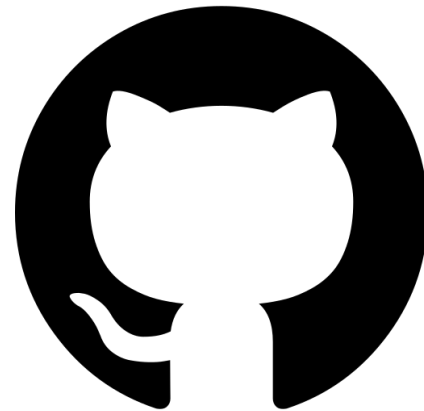
In-class activities + computing labs + homework assignments

Computing Tools



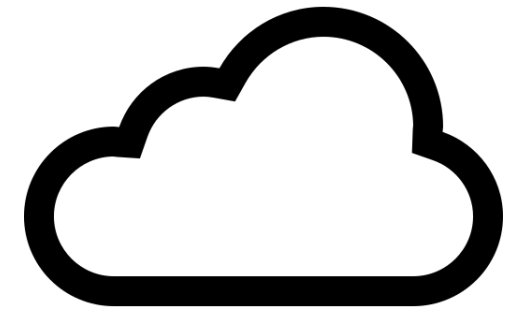
R + R Markdown
for analysis and
write up

Run Git
commands
through Git pane



Assign and submit
assignments

Platform for
collaboration on
group
assignments



RStudio Cloud

Packages installed
+ Git configured
on Day 1

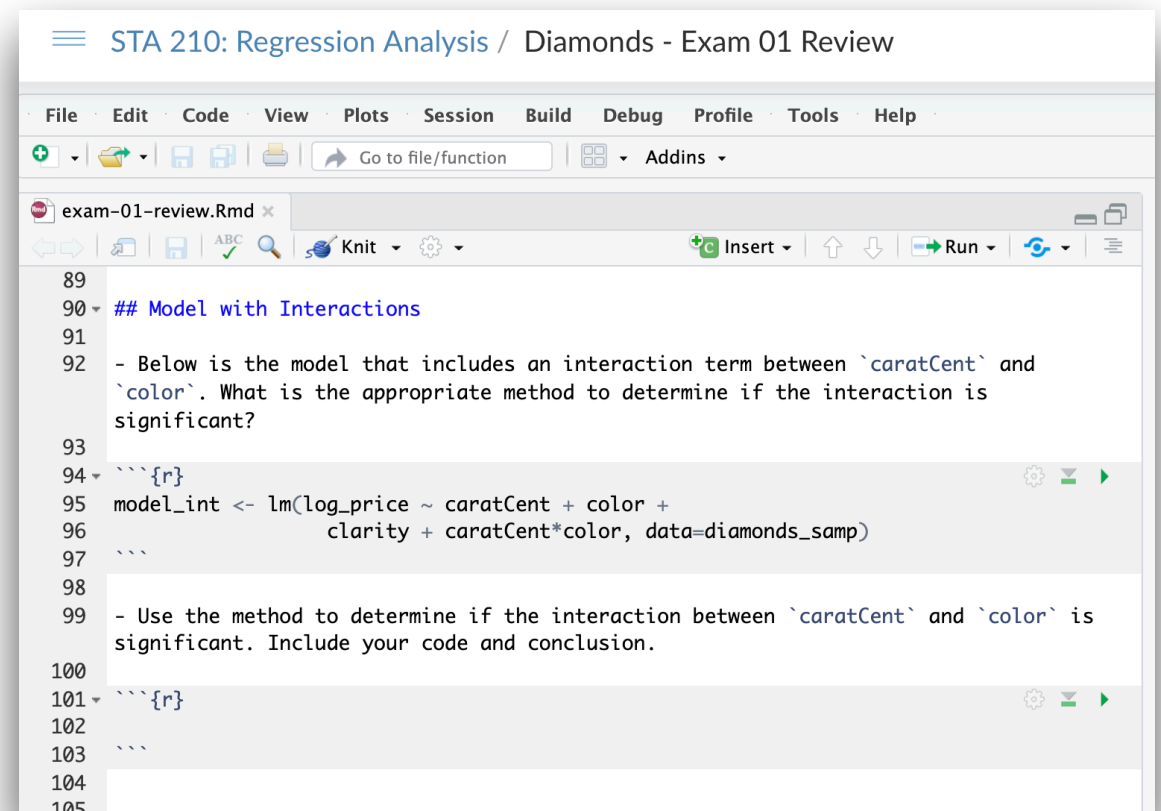
Introduction

Assignments

Challenges
+
Learning
Outcomes

In-Class Activities

1. Create RStudio Cloud project with R Markdown template and data.
2. Students make copy of RStudio Cloud project and answer questions by writing short lines of code and interpreting the results.
3. Display a student's RStudio Cloud project on the classroom screen as the student (or group) shares their work with the class.



The screenshot displays the RStudio Cloud web interface. The browser address bar shows the URL: STA 210: Regression Analysis / Diamonds - Exam 01 Review. The interface includes a menu bar with options: File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. Below the menu is a toolbar with icons for file operations and a search bar labeled 'Go to file/function'. The main editor area shows an R Markdown document titled 'exam-01-review.Rmd'. The document content includes a title '## Model with Interactions' and a paragraph asking for a method to determine if an interaction is significant. The R code block defines a linear model: `model_int <- lm(log_price ~ caratCent + color + clarity + caratCent*color, data=diamonds_samp)`. The document also contains a section for the student to provide their conclusion.

```
89
90 ## Model with Interactions
91
92 - Below is the model that includes an interaction term between `caratCent` and
93   `color`. What is the appropriate method to determine if the interaction is
94   significant?
95
96 ```{r}
97 model_int <- lm(log_price ~ caratCent + color +
98                 clarity + caratCent*color, data=diamonds_samp)
99 ```
100
101 - Use the method to determine if the interaction between `caratCent` and `color` is
102   significant. Include your code and conclusion.
103
104 ```{r}
105 ```
```

Creating Assignments

1. Create starter repo in GitHub.
2. Make a copy of the starter repo for each student (or team) using **ghclass**.
3. Students clone repo into RStudio, write code and narrative in the R Markdown template, and knit to produce a Markdown document.
4. Write feedback to students as an “issue” in the GitHub repo and post grades in course management system.

The top screenshot shows a GitHub repository for 'STA210-Sp19 / hw-01-slr'. It displays the repository's overview, including 12 commits, 1 branch, 0 releases, and 1 contributor. The file list includes `data`, `.gitignore`, `README.md`, `hw-01-slr.Rmd`, `hw-01-slr.Rproj`, `hw-01-slr.html`, and `hw-01-slr.md`. The bottom screenshot shows the `README.md` file, which contains the assignment title 'HW 01: Simple Linear Regression' and a description of the assignment.

The middle screenshot shows a question prompt for a linear regression assignment. It asks students to analyze the simple linear regression model created in Lab 01. The question includes two parts: (1) Before fitting the regression model, students want to do two things to prepare the data: take a subset that only contains data from winter months, and create a new variable `temp_c` that is calculated as `temp * 41`. (2) Fit a regression model with `temp_c` as the predictor variable and `count` as the response and assign the results to `winter_model`. Use the code below to display the model coefficients along with the test statistics and confidence intervals for the coefficients.

The bottom screenshot shows an R Markdown template for the assignment. It includes a header section with the title 'HW 01: Simple Linear Regression' and a sub-header 'Part 1: Concepts & Computations'. The template includes three questions, each with a code block for the student to write their response. The first question is 'Question 1' and the second is 'Question 2'. The third question is 'Question 3' and it asks the student to write their response to Question 3 here. There is no code required.

Computing Assignments

The data for this part of the lab is the `Hitters` dataset in the `ISLR` package. Your goal is to fit a regression model that uses the performance statistics of baseball players to predictor their salary. There are 19 potential predictor variables, so you will use the `regsubsets` function to conduct forward selection to choose a final model.

Exercise 7. Read through the data dictionary for the `Hitters` dataset. You can access it by typing `?Hitters` in the console. What is the difference between the variables `HmRun` and `CHmRun`?

Exercise 8. Some observations have missing values for `Salary`. Filter the data, so only observations that have values for `Salary` are included. You will use this filtered data for the remainder of the lab.

Exercise 9. Fill in the code below to conduct forward selection and save the results in an object called `sel_summary` (selection summary).

```
regfit_forward <- regsubsets(_____, _____, method="forward", nvmax = 19)
sel_summary <- summary(_____)
```

The `nvmax` option indicates the maximum-sized variable subsets to consider in the model selection.

Homework Assignment

Part 2: Data Analysis

The *Data Analysis* section of homework contains open-ended data analysis questions. Your response should be neatly organized and read as a complete narrative. This means that in addition to addressing the question there should also be exploratory data analysis and an analysis of the model assumptions. In short, these questions should be treated as “mini-projects”.

Question 8. For this portion, you will use the `housing` data you started analyzing in [Lab 04](#). Use the code below to load the data and prepare the data.

```
houses <- read_csv("data/KingCountyHouses.csv")
houses <- houses %>%
  filter(bedrooms <= 5 ) %>%
  mutate(floorsCat = as.factor(floors),
         sqftCent = sqft - mean(sqft),
         bedroomsCent = bedrooms - mean(bedrooms),
         bathroomsCent = bathrooms - mean(bathrooms),
         logprice = log(price))
```

Fit a regression model with `logprice` as the response and `floorsCat`, `sqftCent`, `bedroomsCent`, `bathroomsCent`, and `waterfront` as predictor variables. In your analysis, include the following:

- Briefly explain why we should use the log-transformed version of `price` instead of the original version of the variable.
- Describe the relationship between a house’s price and square footage (holding all else constant), including the appropriate confidence intervals.
- Describe how the expected price differs based on the number of floors in the house (holding all else constant). Include discussion about whether or not the differences are statistically significant.

Introduction

Assignments

Challenges
+
Learning
Outcomes

Challenges

Different computing backgrounds

- Begin semester with short in-class exercises to expose students to RStudio and GitHub early-on
- Have students work in groups that include diverse computing experiences

Learning curve for Git/GitHub

- Use Git through Git pane in RStudio
- Only focus on basic Git functions (push, pull, commit)

Handling large classes

- RStudio Cloud (or other server) to reduce computing differences
- **ghclass** package for easy course management

Learning Outcomes

Iterative Workflow

- Students "submit" work frequently by pushing it to GitHub
- Easier for students to incorporate feedback, make changes and include new ideas along the way

Professional Computing Skills

- R and GitHub are popular tools in industry
- Students gain computing skills that are applicable for internships, jobs, and higher-level classes

Professional Collaboration

- GitHub makes it easier to truly collaborate on group assignments
- Students gain skills for collaboration and sharing work in the workplace

Thank You!



maria.tackett@duke.edu



[@MT_statistics](https://twitter.com/MT_statistics)



www.introregression.org

Maria Tackett
Duke University

References

- Çetinkaya-Rudel, M., & Rundel, C. (2017). Infrastructure and Tools for Teaching Computing Throughout the Statistical Curriculum. *The American Statistician*, 72, 58-65.
- Cloud and GitHub icons from www.flaticon.com
- Course Management with ghclass. Retrieved from <https://rundel.github.io/ghclass/>
- Curriculum Guidelines for Undergraduate Programs in Statistical Science. Retrieved from <https://www.amstat.org/asa/files/pdfs/EDU-guidelines2014-11-15.pdf>
- Data Science in a Box. Retrieved from <https://datasciencebox.org/>