



LECTURE 02:

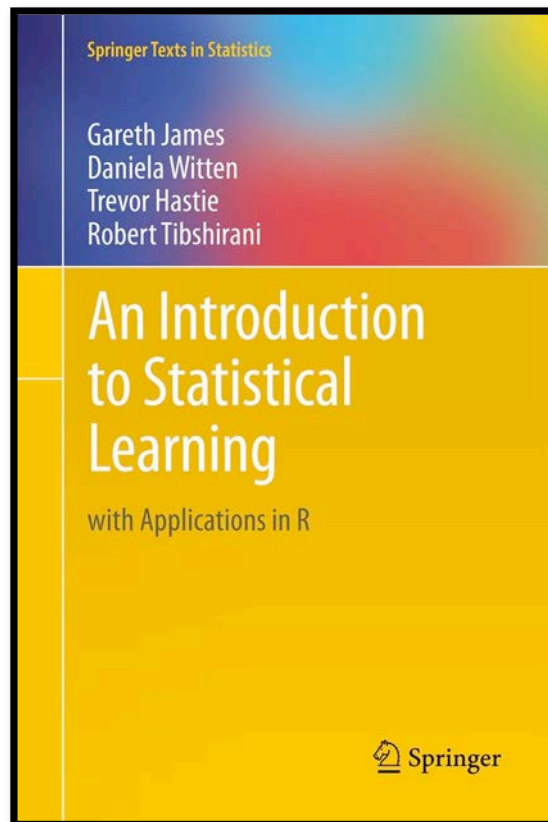
EVALUATING MODELS

September 13, 2017

SDS 293: Machine Learning

Announcements / Questions

- Jordan's office hours: Monday 10:30am – noon
 - does anyone have a permanent conflict?
- Textbook



Course Description

The field of statistical learning encompasses a variety of computational tools, many of the most popular of these tools, such as sparse regression, and the mathematics underlying the computational methods, students will learn to apply these tools to real-world data.

Prerequisite: MTH 220 (or an equivalent intro. statistics course)

Schedule

Date	Topic
09-11	Introduction to Machine Learning (p.1-28)
09-13	Evaluating Models (p.29-51)
09-18	Simple and Multiple Linear Regression (p.59-82)
09-20	Assumptions and Other Potential Problems (p.82-119)

If you like to read ahead, pages are posted on the course website for each lecture

Outline

- Finish course overview
 - General info
 - Topics
 - Textbook
 - Grading
 - Expectations
- Evaluating models
 - Regression
 - Classification
 - Bias-variance trade off
- Quick R demo (time permitting)

General information

- Course website:

cs.smith.edu/~jcrouser/SDS293

- Slack Channel is live:

sds293.slack.com

- Syllabus (with slides before each lecture)
- Textbook download
- Assignments
- Grading
- Accommodations

What we'll cover in this class

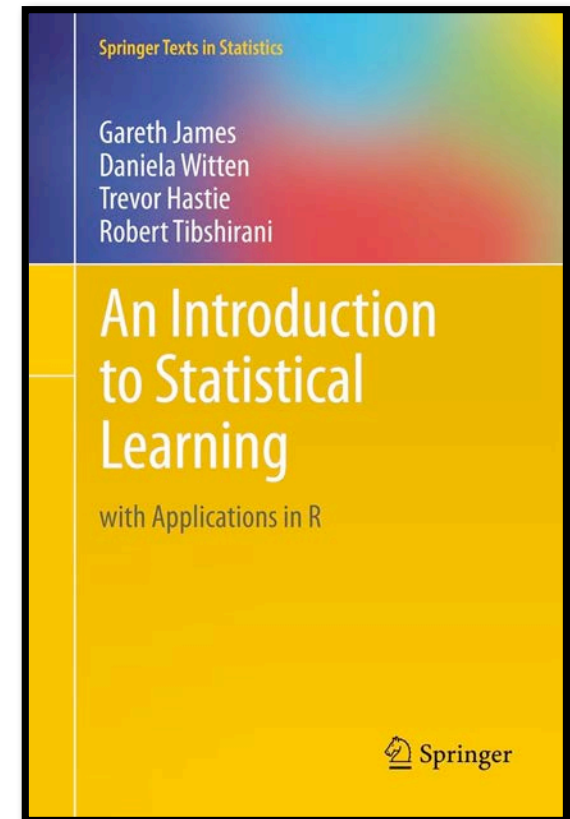
- Ch. 2: Statistical Learning Overview (today)
- Ch. 3: Linear Regression
- Ch. 4: Classification
- Ch. 5: Resampling Methods
- Ch. 6: Linear Model Selection
- Ch. 7: Beyond Linearity
- Ch. 8: Tree-Based Methods
- Ch. 9: Support Vector Machines
- Ch. 10: Unsupervised Learning

About the textbook

- Digital edition available for free at:
www.statlearning.com
- Lots of useful R source code (including labs)
- The `ISLR` package includes all the datasets referenced in the book:

```
> install.packages('ISLR')
```
- Many excellent GitHub repositories of solution sets available

...wait, what?



Disclaimer

this class is an experiment in
constructionism

(the idea that people learn most effectively
when they're building personally-meaningful things)

- My job as the instructor:



Assignments and grading

- **Labs (20%)**: run during regular class time, help you get a hands-on look at how various ML techniques work
- **8 (short) Assignments (40%)**: built to help you become comfortable with applying the techniques
- **Engagement (20%)**:
 - Show up, ask questions, engage on Slack
 - Take DataCamp courses
 - Go to bonus lectures
 - etc.
- **Course project (20%)**

Preparing for labs in R



Two options available for using R:

1. You can install R Studio on your own machine: rstudio.com
2. You can use Smith's RStudio Server: rstudio.smith.edu:8787

If you're unfamiliar with R, you might want to take a look at Smith's "Getting Started with R" tutorial:

www.math.smith.edu/tutorial/r.html

Preparing for labs in python



ANACONDA®

- I like the Anaconda distribution from continuum.io, but you're welcome to use whatever you like
- You'll need to know how to **install packages**
- Either 2.7 or 3.6 is fine – we'll run into bugs either way 😊

Course project (20%)

- Topic: ANYTHING YOU WANT
- Goals:
 - Learn how to break big, unwieldy questions down into clear, manageable problems
 - Figure out if/how the techniques we cover in class apply to your specific problems
 - **Use ML to address them**
- Several (graded) milestones along the way
- Demos and discussion on the final day of class
- More on this later...

What I expect from you

- You like difficult problems and you're excited about **“figuring stuff out”**
- You have a solid foundation in **introductory statistics** (or are ready to work to get there)
- You are proficient in **coding and debugging** (or are ready to work to get there)
- You're willing to ask **questions**

What you can expect from me

- I value your learning **experience and process**
- I'm **flexible** w.r.t. the topics we cover
- I'm happy to share my **professional connections**
- Somewhat **limited in-person access**, but I respond quickly on Slack

Course learning objectives



1. Understand what ML is (and isn't)



2. Learn some foundational methods / tools



3. Be able to choose methods that make sense

One model to rule them all...?

Question: why not just teach you the **best** method first?



Answer: there isn't one

- No single method dominates
- One method may prove useful in answering some questions on a given data set
- On a related (not identical) dataset or question, another might prevail



Measuring “quality of fit”

- *Question we often ask:* how **good** is my model?
- *What we usually mean:* how well do my model’s predictions **actually match** the observations?

How do we choose the **right approach**?



Mean squared error

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

True response
for the i^{th} observation

We take the average
over all observations

Prediction our model gives
for the i^{th} observation

“Training” MSE

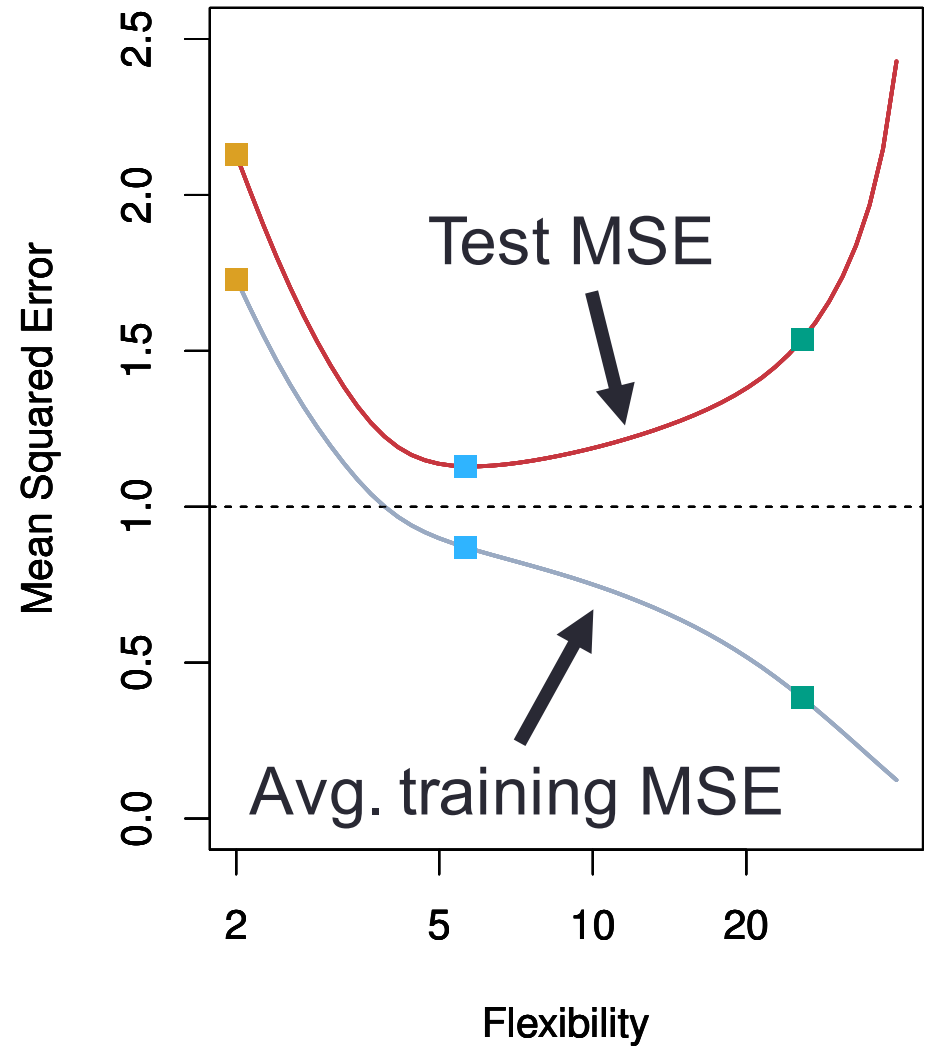
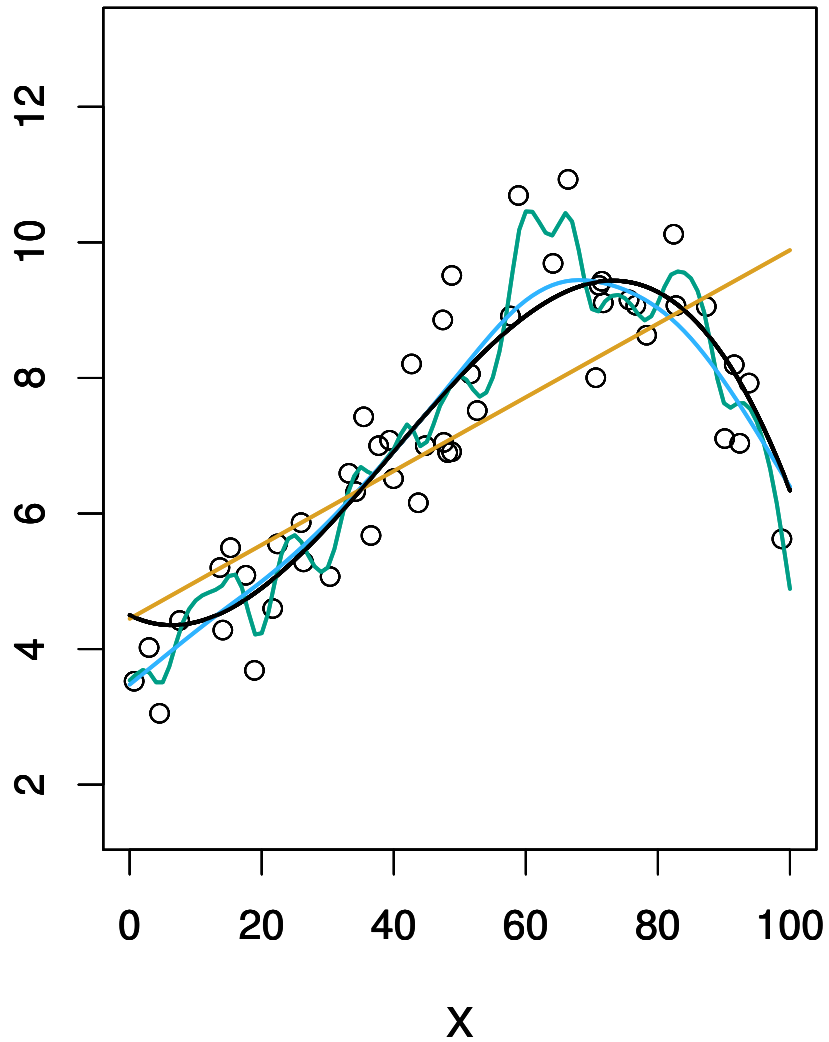
- This version of MSE is computed using the **training data** that was used to fit the model
- **Reality check:** is this what we care about?



Test MSE

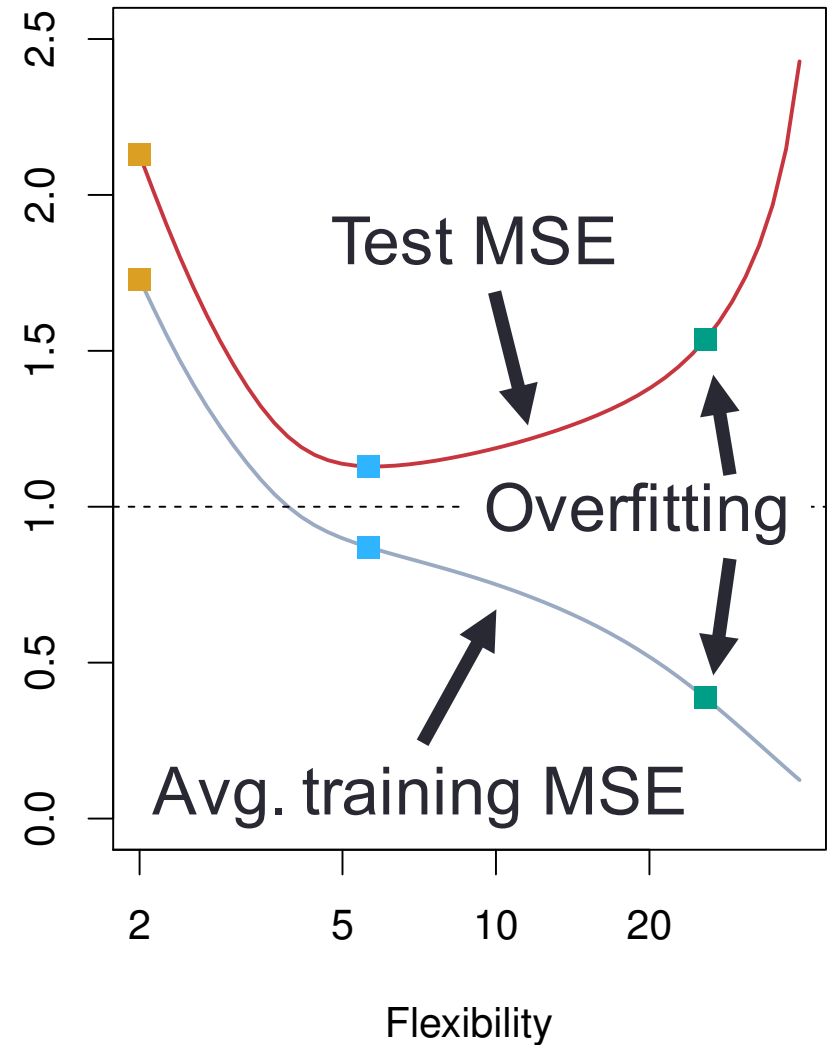
- **Better plan:** see how well the model does on observations we *didn't* train on
- Given some never-before-seen examples, we can just calculate the MSE on those using the same method
- What if we don't have any new observations to test?
 - Can we just use the training MSE?
 - Why or why not?

Example



Training vs. test MSE

- As flexibility \uparrow :
 - monotone \downarrow in training MSE
 - U-shape in the test MSE
- **Fun fact:** occurs regardless of data or statistical method
- This is called **overfitting**



Training vs. test MSE

Question: why does this happen?

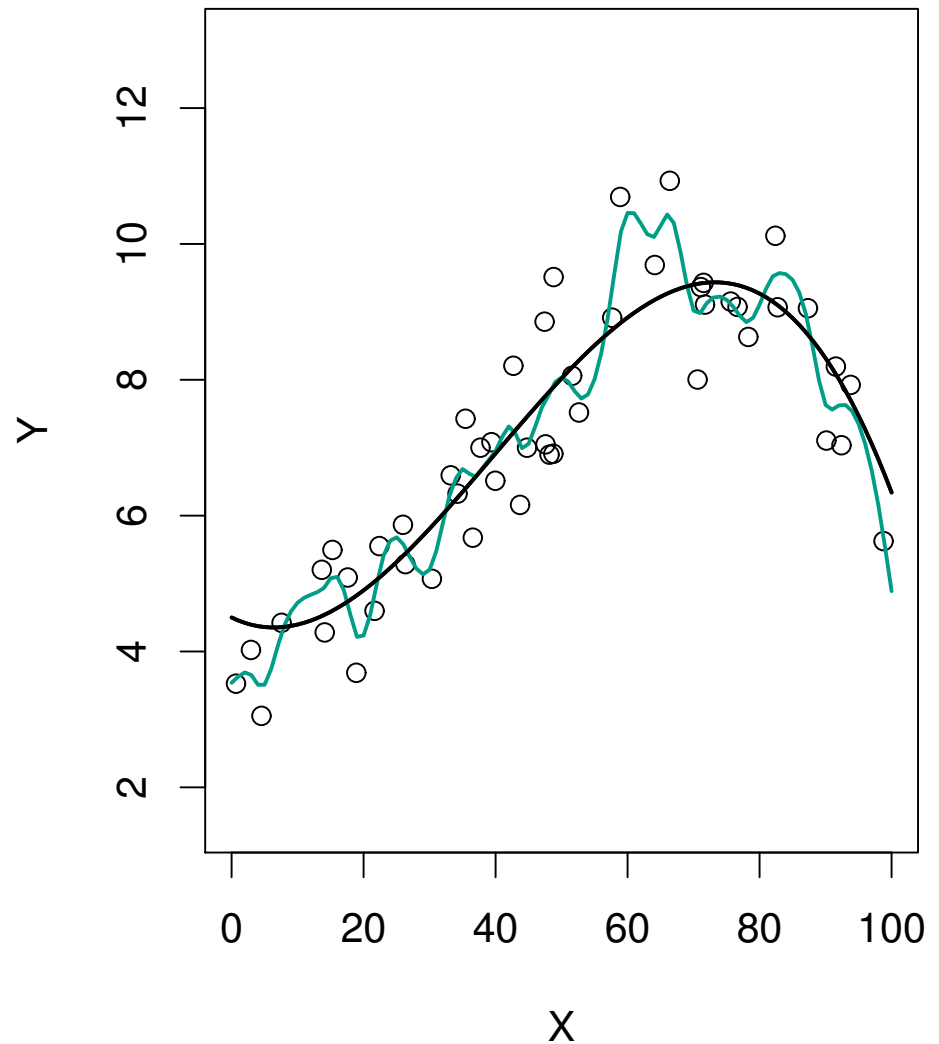
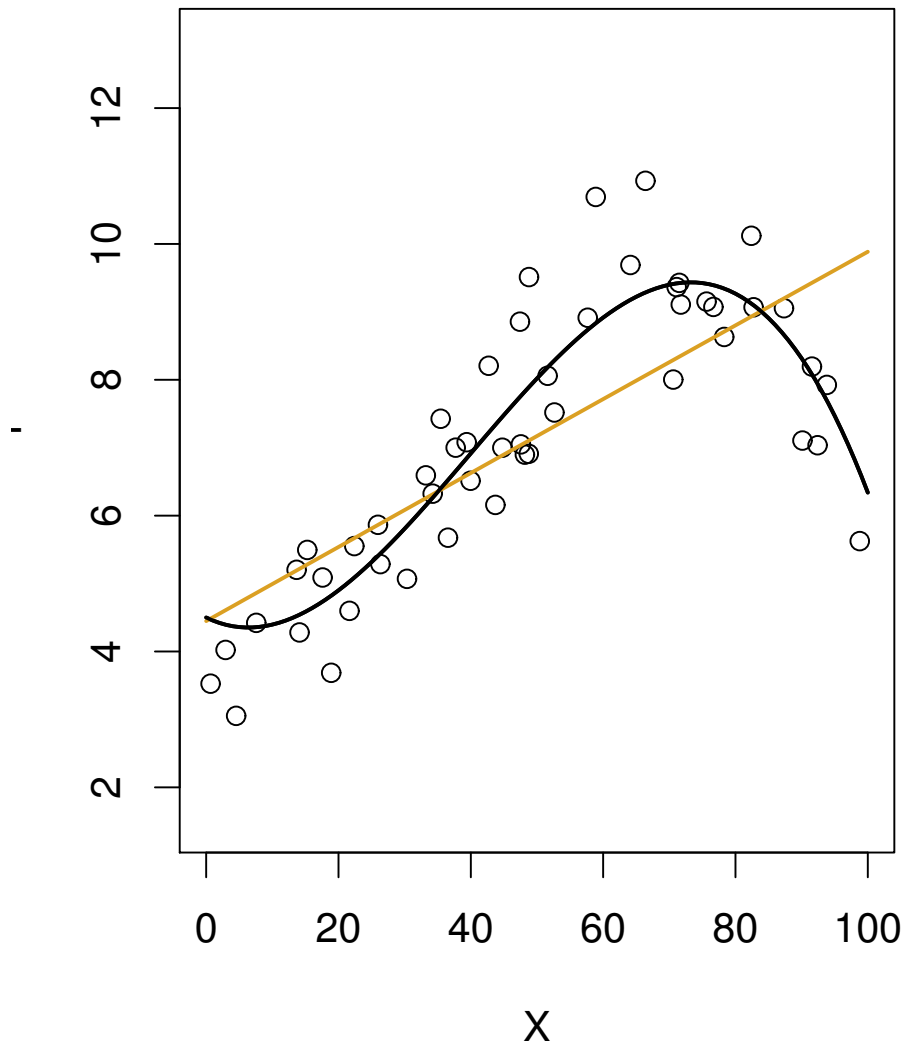


Trade-off between bias and variance

- The U-shaped curve in the Test MSE is the result of two competing properties: *bias* and *variance*
- **Variance**: the amount the model would change if we had different training data
- **Bias**: the error introduced by approximating a complex phenomenon using a simple model

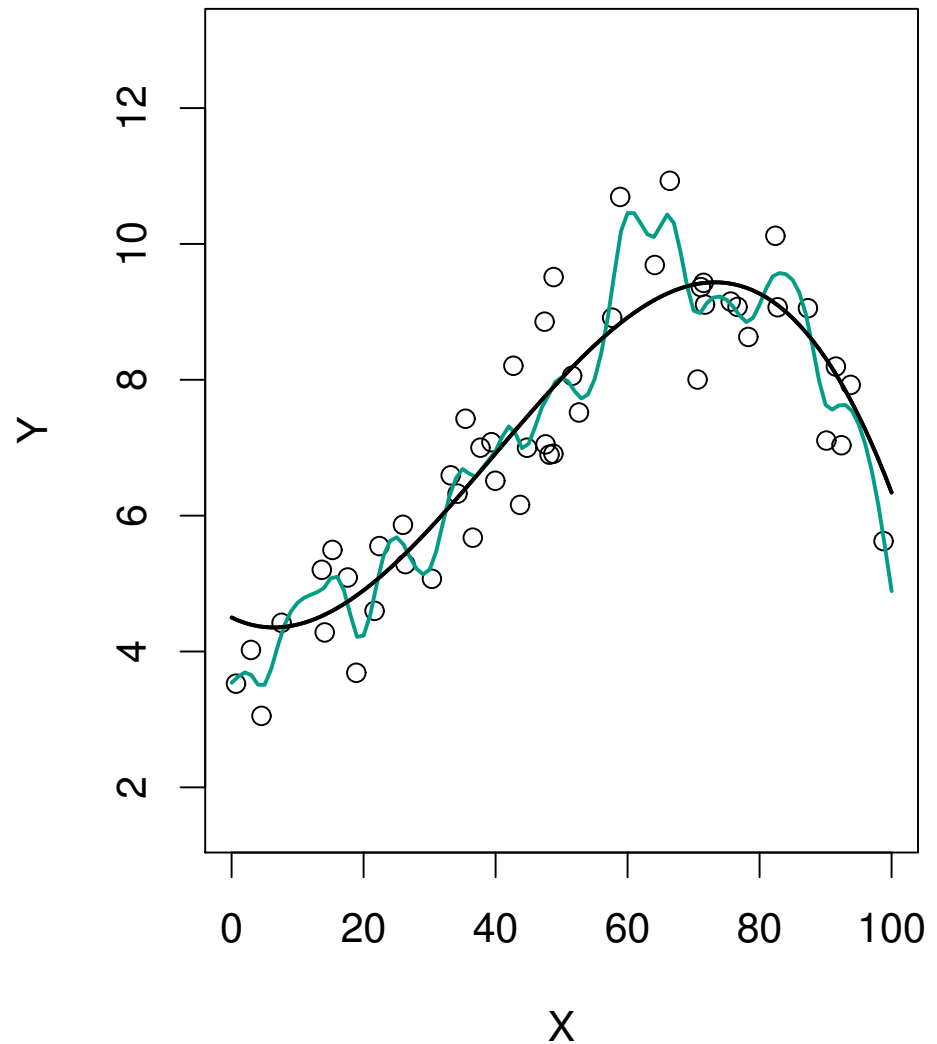
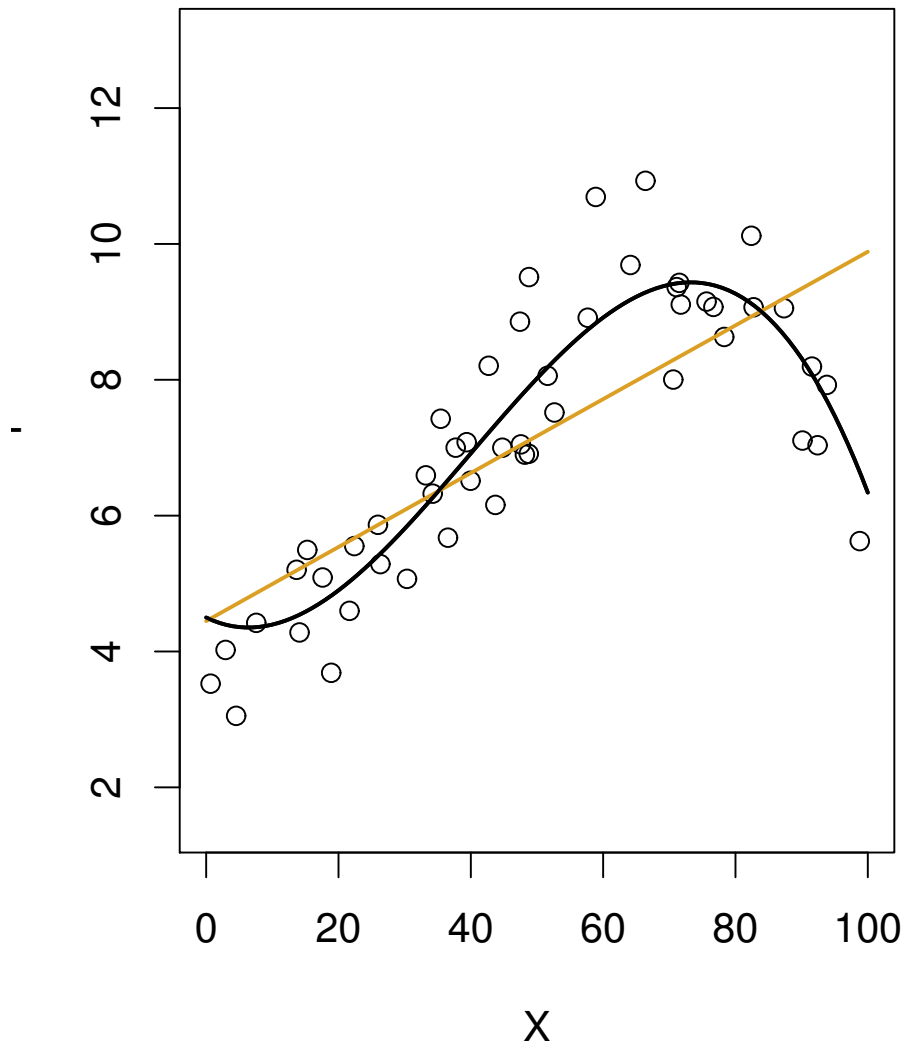
Relationship between bias and variance

- In general, more flexible methods have **higher variance**



Relationship between bias and variance

- In general, more flexible methods have **lower bias**



Trade-off between bias and variance

- Expected test MSE can be decomposed into three terms:

$$E\left(y_0 - \hat{f}(x_0)\right)^2 = \text{Var}\left(\hat{f}(x_0)\right) + \left[\text{Bias}\left(\hat{f}(x_0)\right)\right]^2 + \text{Var}(\varepsilon)$$

The variance of our model
on the test value



The bias of our model
on the test value



The variance
of the error terms



Balancing bias and variance

- It's easy to build a model with
low variance but **high bias** (how?)
- Just as easy to build one with
low bias but **high variance** (how?)
- The challenge: finding a method for which both the variance and the squared bias are low
- This trade-off is one of the most important recurring themes in this course

What about classification?

- So far: how to evaluate a **regression** model
- Bias-variance trade-off also present in **classification**
- Need a way to deal with **qualitative responses**

What are some options?



Training error rate

- **Common approach:** measure the proportion of the times our model incorrectly classifies a training data point

and take the average → $\frac{1}{n} \sum_{i=1}^n I(y_i \neq \hat{y}_i)$

↑
tally up all the times

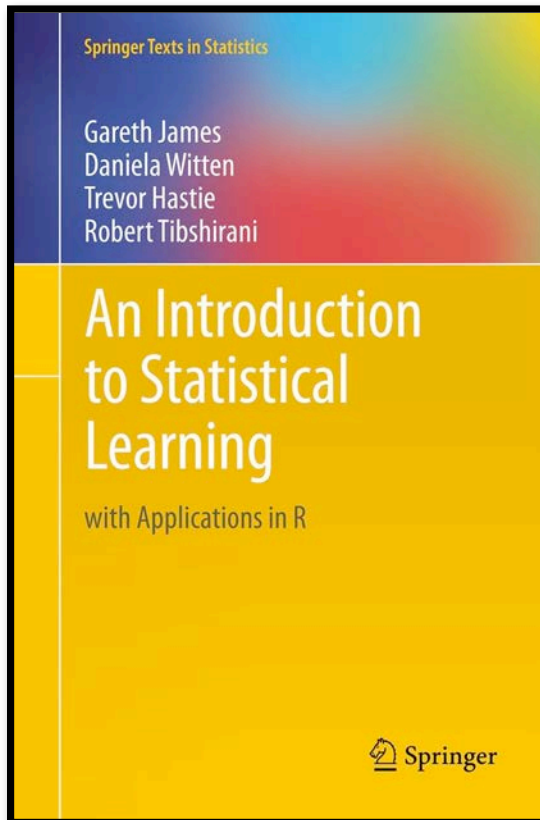
↑
where the model's classification was **different** from the true class

Takeaways

- Choosing the “right” level of flexibility is **critical** (in both regression and classification)
- Bias-variance trade off makes this challenging
- Coming up in Ch. 5:
 - Various methods for **estimating** test error rates
 - How to use these estimates to find the **optimal level** of flexibility

Reading

- In today's class, we covered ISLR: p. 29-37
- Next class, we'll have a crash course in linear regression (ISLR: p. p.59-82)

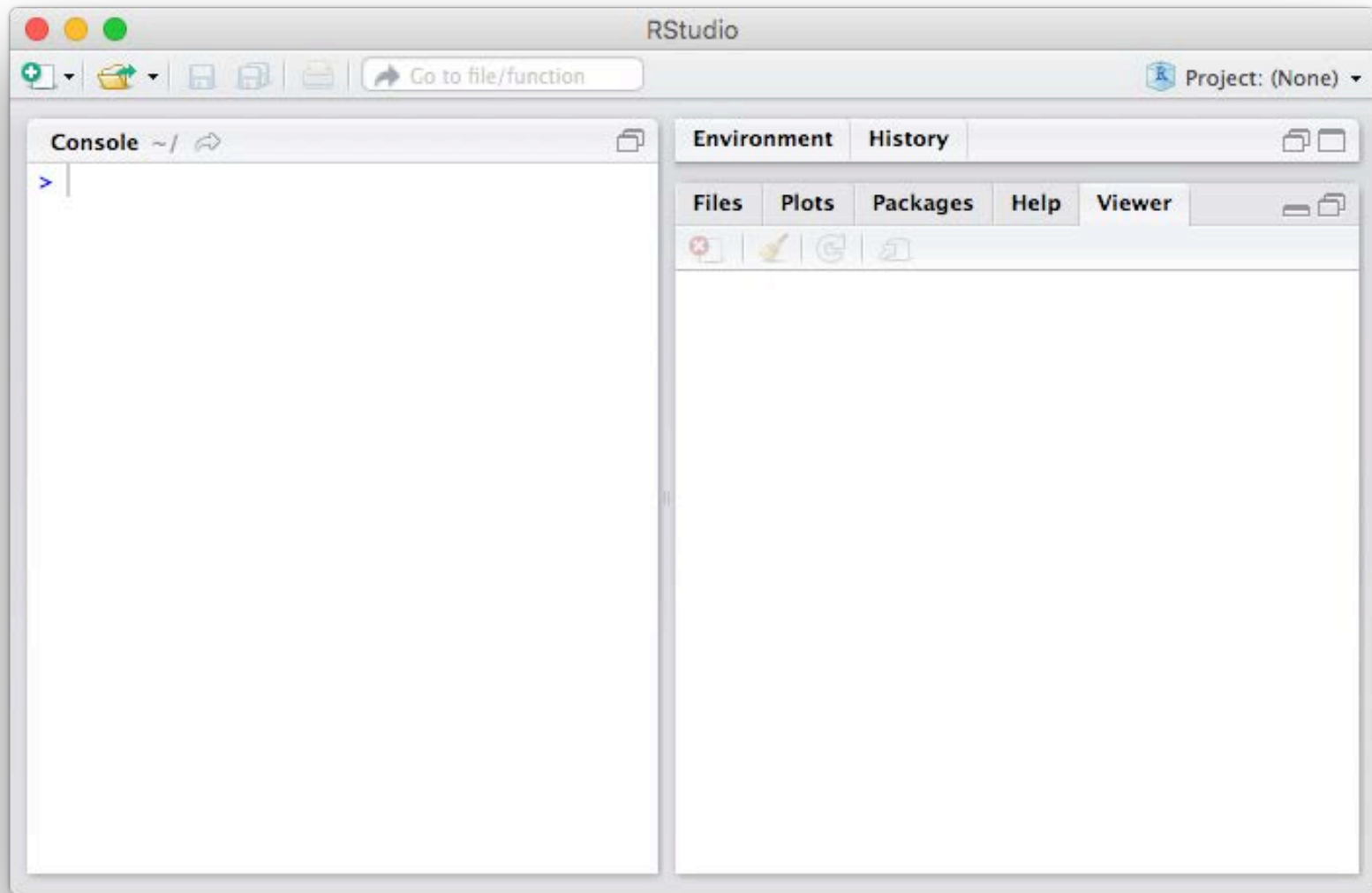


Introduction to R

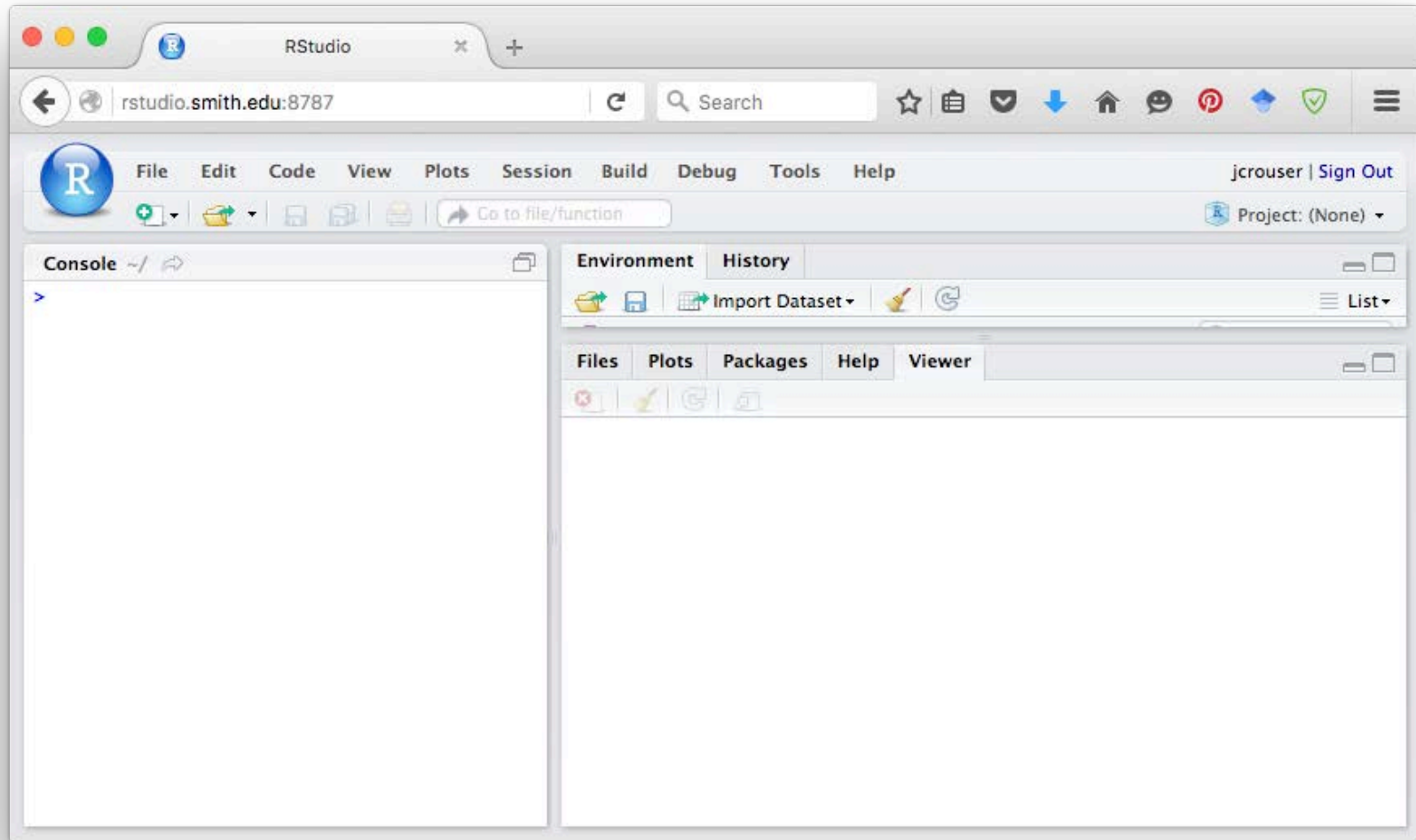


- Basic commands
- Loading external data
- Data wrangling 101
- Graphics
- Generating summaries

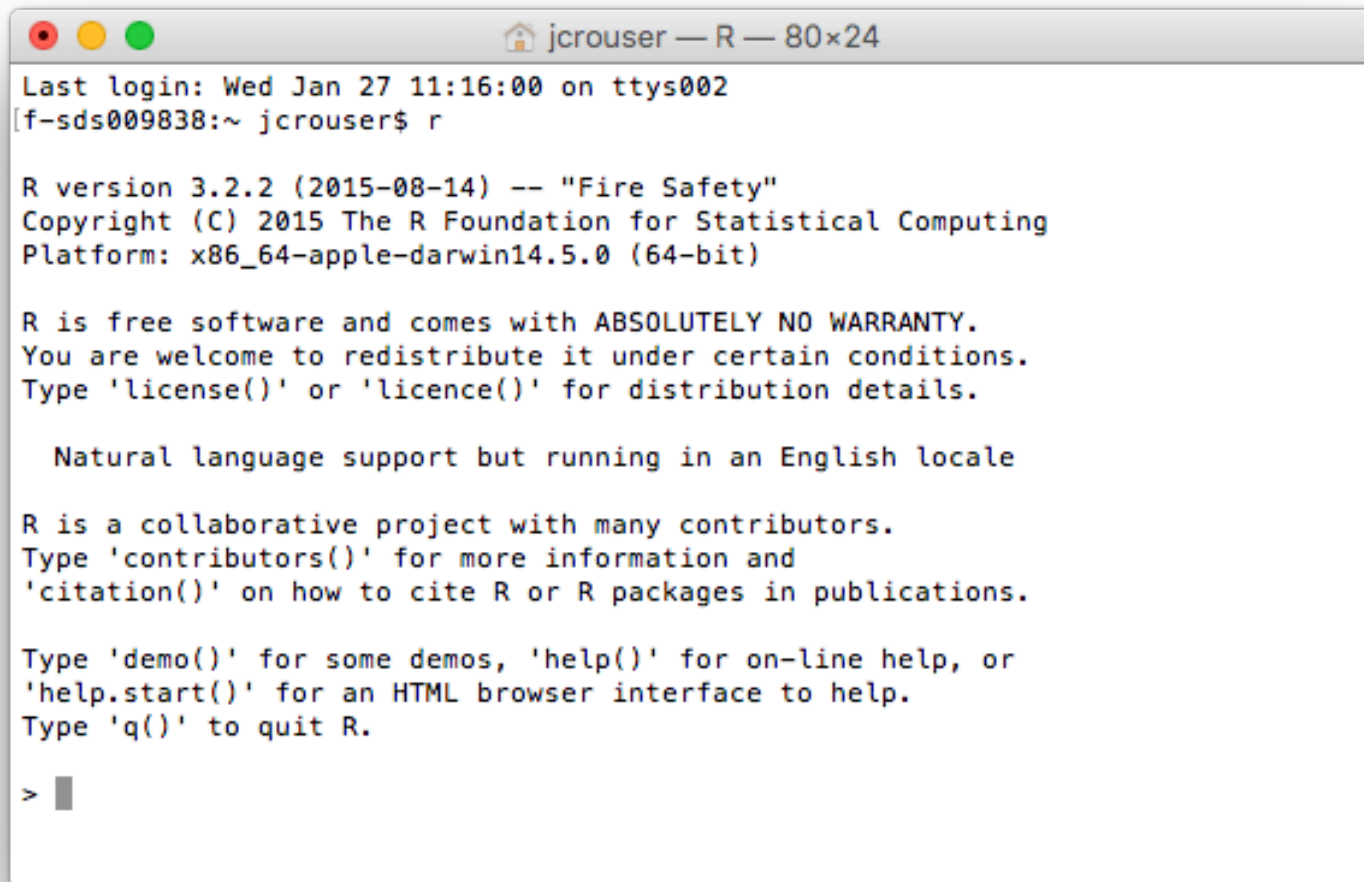
Introduction to R



Introduction to R



Introduction to R

A terminal window titled "jcrouser — R — 80x24" showing the output of running the R command. The window has a standard macOS-style title bar with red, yellow, and green window control buttons. The text inside the terminal is as follows:

```
Last login: Wed Jan 27 11:16:00 on ttys002
[f-sds009838:~ jcrouser$ r

R version 3.2.2 (2015-08-14) -- "Fire Safety"
Copyright (C) 2015 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin14.5.0 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

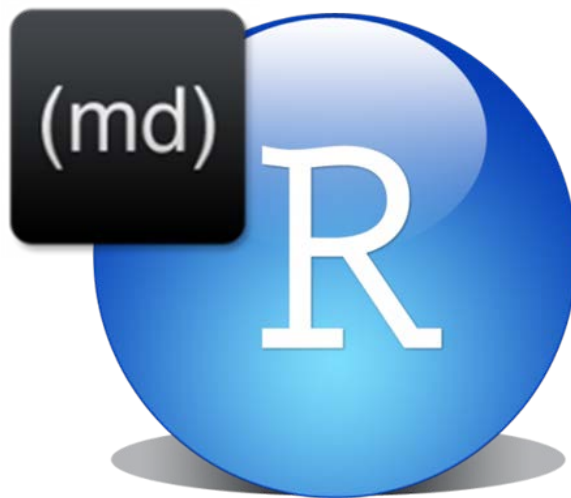
R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> █
```

Introduction to R

- Today's walkthrough was run using R Markdown:



- This allows me to build “notebooks” to combine step-by-step code and instructions/descriptions
- Want to learn more? Check out the “**Reporting with R Markdown**” course on DataCamp!

For Monday



Make sure you
can access the
slack channel



DataCamp

Need a refresher
on something?
Just ask!



Install the tool(s)
you're planning
to use for lab

#questions?

