

R PROGRAMMING

**for environmental health
research**

Brooke Anderson
Colorado State University

April 8, 2019

Today's plan

ORGANIZE

TRACK

PACKAGE

COLLECT

PROCESS

Homework?!

<https://bit.ly/2WQV6XT>

PREREQUISITES

Setting up

■ Install RStudio Desktop

<https://www.rstudio.com/>

Install git

<https://git-scm.com/downloads>

Create GitHub account

<https://github.com/>

■ Download example project

[https://github.com/geanders/
columbia_env_health_examples](https://github.com/geanders/columbia_env_health_examples)

ORGANIZE

Setting up

One project : One directory

Rule #1 of research project file organization

Use consistent names

Rule #2 of research project file organization

Use relative filenames

Rule #3 of research project file organization

Common project subdirectories

data-raw Raw data and R scripts to clean the raw data.

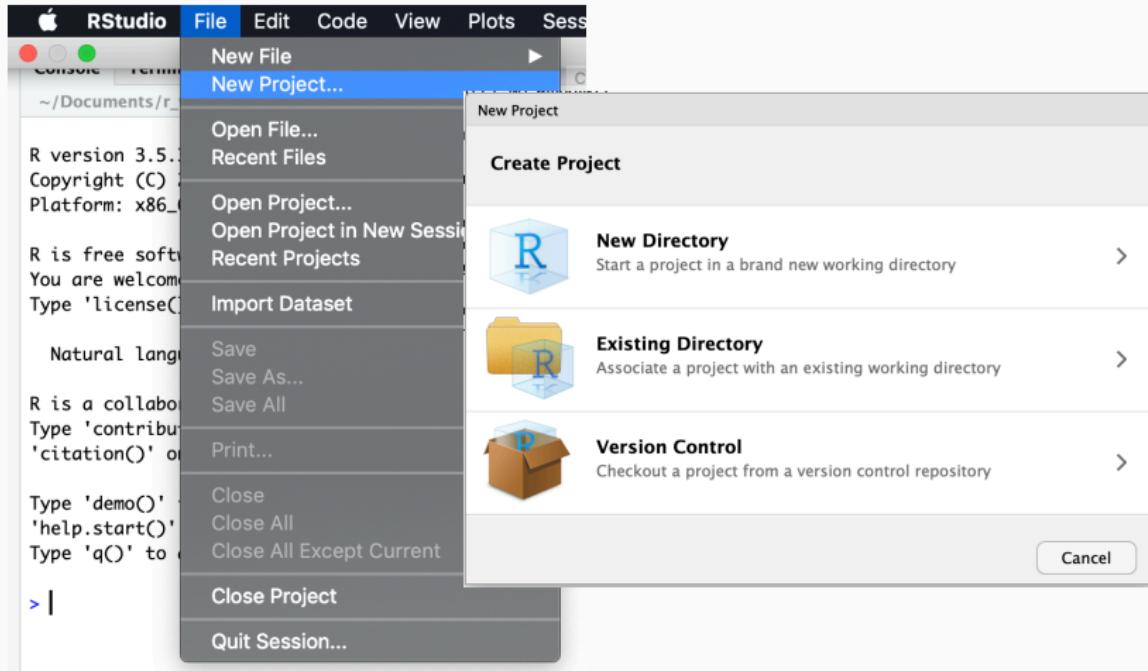
data Cleaned data, often saved as .RData after being generated by a script in data-raw.

R Code for any functions used in analysis.

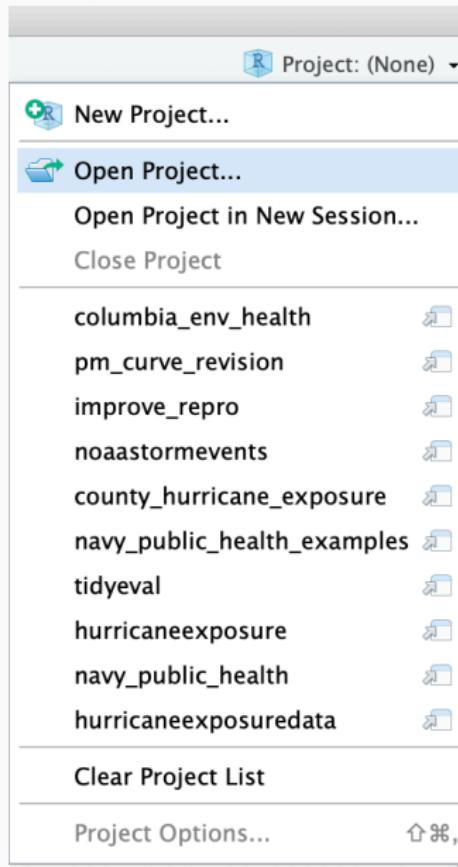
figures Figures created from R code.

reports R Markdown files and products rendered from those files (e.g., paper drafts, presentations).

Create R project



Navigating R Projects



Navigating R Projects

The screenshot shows the RStudio interface with a project titled "columbia_env_health_examples".

Left Panel (Code Editor): Displays three R script files: "collect.R", "package.R", and "process.R". The code in "collect.R" is as follows:

```
1 # Load general packages
2
3 library(dplyr)
4
5 # Load example data
6
7 ## The package `dlm` includes an example dataset with weather and
8 ## mortality data from Chicago, IL, for 1987--2000 (originally from
9 ## the NMMAPS dataset). To load this data, run:
10
11 library(dlm)
12 data(chicagoNMMAPS)
13
14 # Example: identify and plot heat wave days
15
```

Right Panel (Project View): Shows the project structure with the following files and folders:

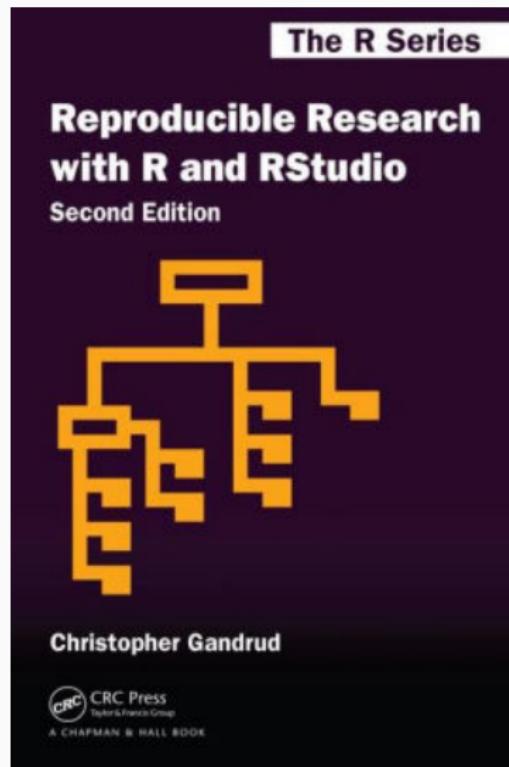
- New Folder
- Delete
- Rename
- More
- workshops > columbia_env_health_examples
 - Name
 - .gitignore (49 B)
 - columbia_env_health_examples... (205 B)
 - R

```
[Georgianas-MacBook-Pro:columbia_env_health georgianaanderson$ ls -a
.
..
.DS_Store
.Rproj.user
.git
.gitignore
.nojekyll
01-organize.Rmd
02-track.Rmd
03-package.Rmd
04-collect.Rmd
05-process.Rmd
06-summary.Rmd
DESCRIPTION
LICENSE
R
README.md
_bookdown.yml
_bookdown_files
```

.Rproj/

```
_build.sh
_output.yml
_workshop_slides
book.bib
columbia_env_health.Rproj
columbia_env_health.log
data
flexdashboard
images
index.Rmd
irma_fatalities.pdf
now.json
old_data
packages.bib
preamble.tex
skeleton.bib
style.css
toc.css
```

Resources



```
irma_week_accs <- fl_accidents %>%  
  group_by(fips) %>%  
  summarize(fatals = sum(fatals))
```

[Live coding example]

```
irma_accs <- fl_counties %>%  
  full_join(irma_week_accs, by = c("GEOID" ~ "fips")) %>%  
  mutate(fatals = ifelse(is.na(fatals), 0, fatals))  
  
fl_accidents <- fl_accidents %>%  
  st_as_sf(coords = c("longitud", "latitude")) %>%  
  st_set_crs(st_crs(st_read(dsn, layer, ...)))
```



```
irma_track <- st_read("data/al112017_best_track",  
                      layer = "al112017_lin") %>%  
  st_transform(crs = st_crs(irma_accs))
```

TRACK

git and GitHub for version control

```
[Georgianas-MacBook-Pro:columbia_env_health georgianaanderson$ ls -a
.
..
.DS_Store
.Rproj.user
.git
.gitignore
.nojekyll
01-organize.Rmd
02-track.Rmd
03-package.Rmd
04-collect.Rmd
05-process.Rmd
06-summary.Rmd
DESCRIPTION
LICENSE
R
README.md
_bookdown.yml
_bookdown_files
```



Using GitHub to collaborate



<https://github.com/ropenscilabs/miner>

Hosting content with GitHub Pages



R for Environmental Health Research

Workshop for Climate and Health students at Columbia Mailman School of Public Health

Brooke Anderson

April 9, 2019

Chapter 1 Prerequisites

1.0.1 Overview

BASED ON REQUESTS FROM some of the students for this workshop, I've focused here on a few topics relevant to environmental health research: organizing projects and tracking them with version control, creating your own packages, and collecting and processing large datasets relevant to environmental health research. You can download the slides from the workshop by [clicking here](#).

```
irma_week_accs <- fl_accidents %>%  
  group_by(fips) %>%  
  summarize(fatals = sum(fatals))
```

[Live coding example]

```
irma_accs <- fl_counties %>%  
  full_join(irma_week_accs, by = c("GEOID" ~ "fips")) %>%  
  mutate(fatals = ifelse(is.na(fatals), 0, fatals))  
  
fl_accidents <- fl_accidents %>%  
  st_as_sf(coords = c("longitud", "latitude")) %>%  
  st_set_crs(st_crs(st_read(dsn, layer, ...)))  
  
irma_track <- st_read("data/al112017_best_track",  
  layer = "al112017_lin") %>%  
  st_transform(crs = st_crs(irma_accs))
```

PACKAGE

Collect R functions in **packages**

Why write R packages

Software development in biostatistics

So I have a new policy when evaluating CV's of candidates for jobs, or when I'm reading a paper as a referee. If the paper is about a new statistical method or machine learning algorithm and there is no software available for that method - I simply mentally cross it off the CV. If I'm reading a data analysis and there isn't code that reproduces their analysis - I mentally cross it off. In my mind, new methods/analyses without software are just vapor ware. Now, you'd definitely have to cross a few papers off my CV, based on this principle. I do that. But I'm trying really hard going forward to make sure nothing gets crossed off.

Source: Jeff Leek, Simply Statistics

Why write R packages

Research impacts of NMMAPS package (*Source: Barnett, Huang, and Turner, "Benefits of Publicly Available Data", Epidemiology 2012*):

As of November 2011, 67 publications had been published using this data, with 1,781 citations to these papers

Research using NMMAPS has been used by the US EPA in creating regulatory impact statements for air pollution (particulates and ozone)

"Thanks to NMMAPS, there is probably no other country in the world with a greater understanding of the health effects of air pollution and heat waves in its population."

What an R package looks like

Folders	Documents	Developer
weathermetrics	cran-comments.md NEWS.md README.md	data.R heat_index.R moisture_conversions.R rainmeasure_conversion.R temperature_conversions.R weathermetrics.R wind_conversions.R
PDF Documents		
weathermetrics.pdf		
Other		
weathermetrics_1.2.0.tar.gz weathermetrics_1.2.2.tar.gz		
	Folders data inst man R vignettes	
	Other DESCRIPTION NAMESPACE README.Rmd weathermetrics.Rproj	

R package template

The screenshot shows the RStudio interface with a project titled "convertr" open. The left pane displays the "hello.R" script, which contains a simple function definition:5 # You can learn more about package authoring with RStudio at:
6 #
7 # <http://r-pkgs.had.co.nz/>
8 #
9 #
10 # Some useful keyboard shortcuts for package authoring:
11 #
12 # Build and Reload Package: 'Cmd + Shift + B'
13 # Check Package: 'Cmd + Shift + E'
14 # Test Package: 'Cmd + Shift + T'
15
16 hello <- function() {
17 print("Hello, world!")
18 }
19

The right pane shows the "Files" view with the following contents:

Name	Size
.Rbuildignore	28 B
convertr.Rproj	356 B
DESCRIPTION	369 B
man	
NAMESPACE	31 B
R	

The "Console" tab shows the R startup message and basic help information.

Required files

R/ or data/ If you don't have one of these, your package won't do anything

DESCRIPTION Needed, but you can't keep the template version as-is

NAMESPACE Needed, but you can't keep the template version as-is

```
irma_week_accs <- fl_accidents %>%  
  group_by(fips) %>%  
  summarize(fatals = sum(fatals))
```

[Live coding example]

```
irma_accs <- fl_counties %>%  
  full_join(irma_week_accs, by = c("GEOID" ~ "fips")) %>%  
  mutate(fatals = ifelse(is.na(fatals), 0, fatals))  
  
fl_accidents <- fl_accidents %>%  
  st_as_sf(coords = c("longitud", "latitude")) %>%  
  st_set_crs(st_crs(st_read(dsn, layer, ...)))
```



```
irma_track <- st_read("data/al112017_best_track",  
                      layer = "al112017_lin") %>%  
  st_transform(crs = st_crs(irma_accs))
```

Resources



<http://r-pkgs.had.co.nz/>

Resources

Package Development: : CHEAT SHEET

Package Structure

A package is a convention for organizing files into directories.

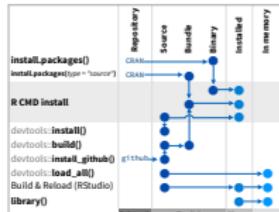
This sheet shows how to work with the 7 most common parts of an R package:



The contents of a package can be stored on disk as a:

- **source** - a directory with sub-directories (as above)
- **bundle** - a single compressed file (.tar.gz)
- **binary** - a single compressed file optimized for a specific OS

Or installed into an R library (loaded into memory during an R session) or archived online in a repository. Use the functions below to move between these states.



Adds file to .Rbuildignore, a list of files that will not be included when package is built.



Setup (DESCRIPTION)

The `DESCRIPTION` file describes your work, sets up how your package will work with other packages, and applies a copyright.

- You must have a `DESCRIPTION` file
- Add the packages that yours relies on with `devtools::use_package()`
- Add a package to the Imports or Suggests field

CC0	MIT	GPL-2
No strings attached.	MIT license applies to your code if MIT-licensed.	GPL-2 license applies to your code, and all code anyone bundles with it, is free-shared.

Write Code (tests/)

All of the R code in your package goes in `tests/`. A package with just an `R` directory is still a very useful package.

- Create a new package project with `devtools::create("path/to/name")`
- Create a template to develop into a package.
- Save your code in `tests/R` as scripts (extension.R)

WORKFLOW

1. Edit your code.
2. Load your code with one of `devtools::load_all()`
Re-load all saved files in `tests/R` into memory.
`Ctrl/Cmd + Shift + L` (keyboard shortcut)
Saves all open files then calls `load_all()`.

3. Experiment in the console.
4. Repeat.

- Use consistent style with r-pkg.had.co.nz/r.html/style
- Click on a function and press F2 to open its definition
- Search for a function with Ctrl + F



Visit r-pkg.had.co.nz to learn much more about writing and publishing packages for R



Package: mypackage
Title: Title of Package
Version: 0.1.0
Author/R: person("Hadley", "Wickham", email = "hadley.wickham@gmail.com")
Description: What this package does (one paragraph)
Depends: R (>= 3.1.0)
License: MIT
LazyData: true
Imports:
 ggplot2 (>= 0.4.0),
 ggviz (>= 0.2)
Suggests:
 knitr (>= 0.1.0)

import packages that your package must have to work. It will install them when it installs your package.
Suggest packages that are not very essential but you may want to install them manually, or not, as they like.

Test (tests/)

Use `tests/testthat` to store tests that will alert you if your code breaks.

- Add a `tests/` directory
- Import `testthat` with `devtools::use_testthat()`, which sets up package to use automated tests with `testthat`
- Write tests with `context()`, `test()`, and `expect` statements
- Save your tests as .R files in `tests/testthat/`

WORKFLOW

1. Modify your code or tests.
2. Test your code with one of `devtools::test()`
Runs all tests in `tests/`
`Ctrl/Cmd + Shift + T` (keyboard shortcut)
3. Repeat until all tests pass

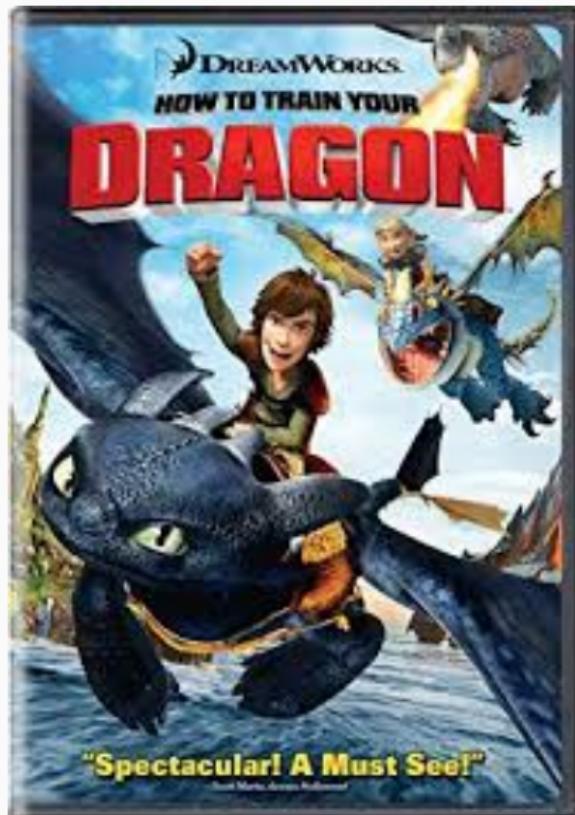
Example Test

```
context("Arithmetic")
test_that("math works", {
  expect_equal(1 + 1, 2)
  expect_equal(1 + 2, 3)
  expect_equal(1 + 3, 4)
})
```

Expected statement	Tests
<code>expect_equal()</code>	is equal within small numerical tolerance?
<code>expect_identical()</code>	is exactly equal?
<code>expect_match()</code>	matches specified string or regular expression?
<code>expect_subset()</code>	prints specified output?
<code>expect_message()</code>	displays specified message?
<code>expect_warning()</code>	displays specified warning?
<code>expect_error()</code>	throws specified error?
<code>expect_no_error()</code>	outputs inherits from certain class?
<code>expect_file()</code>	returns FALSE?
<code>expect_true()</code>	returns TRUE?

<https://www.rstudio.com/resources/cheatsheets/>

Resources



COLLECT

Leverage **open data** tools for collecting data

Data packages

```
library(hurricaneexposure)
county_wind(counties = "36061",
            start_year = 1988, end_year = 2015,
            wind_limit = 17.5) %>%
  select(storm_id, vmax_sust, storm_dist, closest_date)

##      storm_id vmax_sust storm_dist closest_date
## 1 Bob-1991    18.19559  161.571830  1991-08-19
## 2 Bertha-1996   28.95496   16.966013  1996-07-13
## 3 Floyd-1999    20.50178   45.408483  1999-09-16
## 4 Hanna-2008    19.25390   29.916672  2008-09-06
## 5 Irene-2011    25.68553    5.796733  2011-08-28
## 6 Sandy-2012    21.99213  158.040788  2012-10-29
```

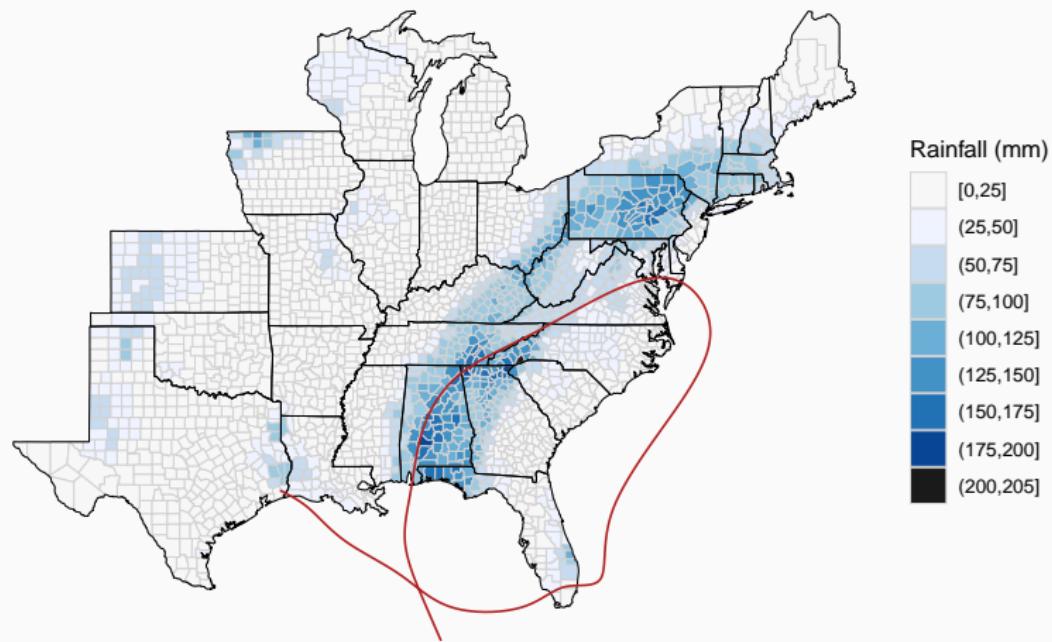
Data packages

```
county_events(counties = "36061",
              start_year = 1988, end_year = 2015,
              event_type = "flood") %>%
  select(storm_id, storm_dist, closest_date)

##      storm_id storm_dist closest_date
## 1    Floyd-1999   45.408483  1999-09-16
## 2 Allison-2001  158.909890  2001-06-17
## 3 Frances-2004  379.343696  2004-09-09
## 4     Ivan-2004  311.346881  2004-09-18
## 5 Jeanne-2004  222.900157  2004-09-29
## 6    Beryl-2006  207.358443  2006-07-20
## 7    Barry-2007  148.251718  2007-06-04
## 8    Irene-2011   5.796733  2011-08-28
## 9 Andrea-2013  92.381282  2013-06-08
```

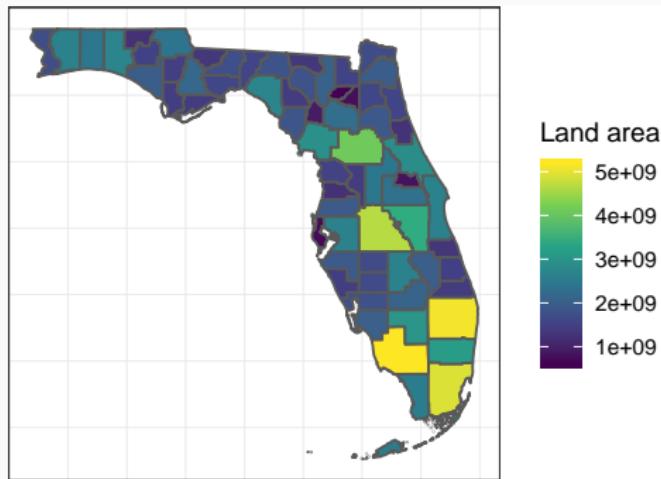
Data packages

```
map_counties(storm = "Ivan-2004", metric = "rainfall")
```



Open Data APIs

```
library(tigris)  
fl_counties <- counties(state = "FL",  
                         class = "sf")
```



```
irma_week_accs <- fl_accidents %>%  
  group_by(fips) %>%  
  summarize(fatals = sum(fatals))
```

[Live coding example]

```
irma_accs <- fl_counties %>%  
  full_join(irma_week_accs, by = c("GEOID" ~ "fips")) %>%  
  mutate(fatals = ifelse(is.na(fatals), 0, fatals))  
  
fl_accidents <- fl_accidents %>%  
  st_as_sf(coords = c("longitud", "latitude")) %>%  
  st_set_crs(st_crs(st_read(dsn, layer, ...)))
```



```
irma_track <- st_read("data/al112017_best_track",  
  layer = "al112017_lin") %>%  
  st_transform(crs = st_crs(irma_accs))
```

Resources

#rstats



Dirk Eddelbuettel @eddelbuettel · 27 Jan 2017

Big congratulations to @gbwanderson whose new package 'hurricaneexposure' just became package 10,000 on CRAN !!

CRAN Package Updates @CRANberriesFeed

9999 packages on CRAN right now, so imagine dozens of R nerds hanging in suspense waiting for the package to make it 10k ...

2

35

93



PROCESS

Find and make **R packages** for processing data

```
irma_week_accs <- fl_accidents %>%  
  group_by(fips) %>%  
  summarize(fatals = sum(fatals))
```

[Live coding example]

```
irma_accs <- fl_counties %>%  
  full_join(irma_week_accs, by = c("GEOID" ~ "fips")) %>%  
  mutate(fatals = ifelse(is.na(fatals), 0, fatals))  
  
fl_accidents <- fl_accidents %>%  
  st_as_sf(coords = c("longitud", "latitude")) %>%  
  st_set_crs(st_crs(st_read(dsn, layer, ...)))
```



```
irma_track <- st_read("data/al112017_best_track",  
                      layer = "al112017_lin") %>%  
  st_transform(crs = st_crs(irma_accs))
```

Resources

ROpenSci

Homework!!

<https://bit.ly/2WQV6XT>