Reproducibility and collaboration in business analytics

Rich Louden - @Richyyl

2019-09-26

Talk overview

- Introduction
- What is reproducibility and why does it matter?
- Leaving a trail to follow
- Being on the same page

Disclaimers

- This talk does not relate to ML or anything particularly flashy, but instead focusses on real problems seen when doing analytics

My background

- Started off in science (repro issues)
- Moved into business via consulting
- Now work in analytics

What is reproducibility?

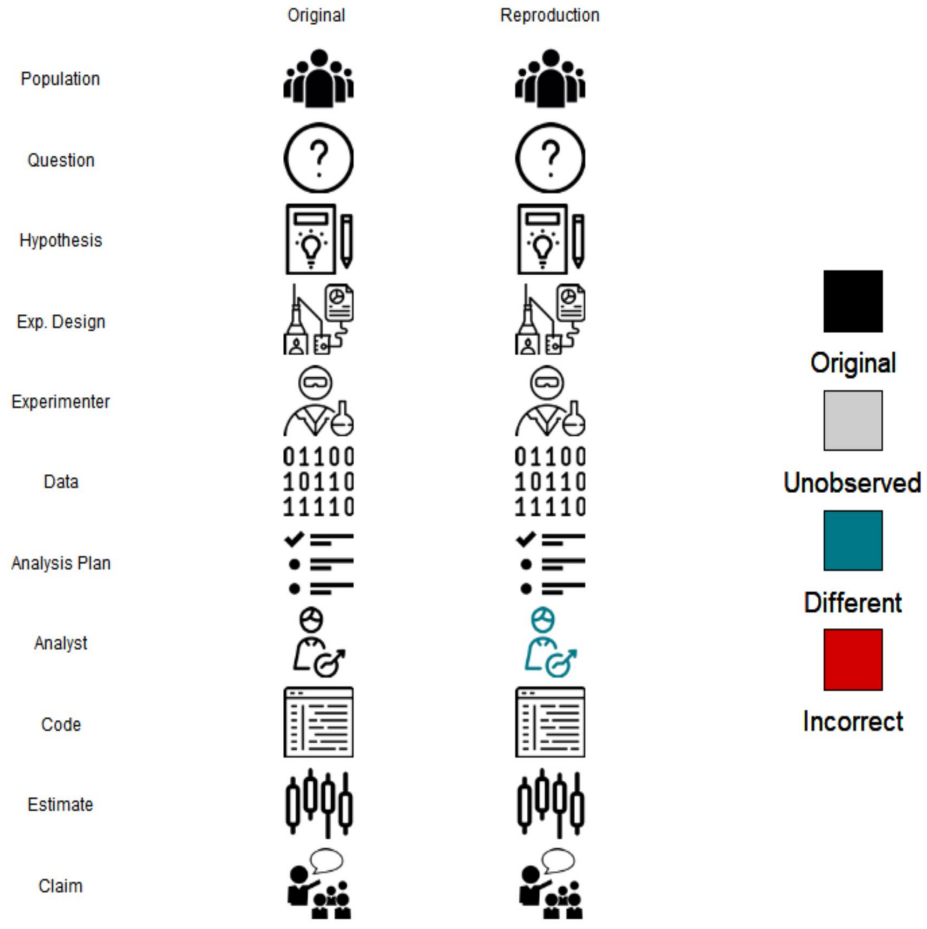
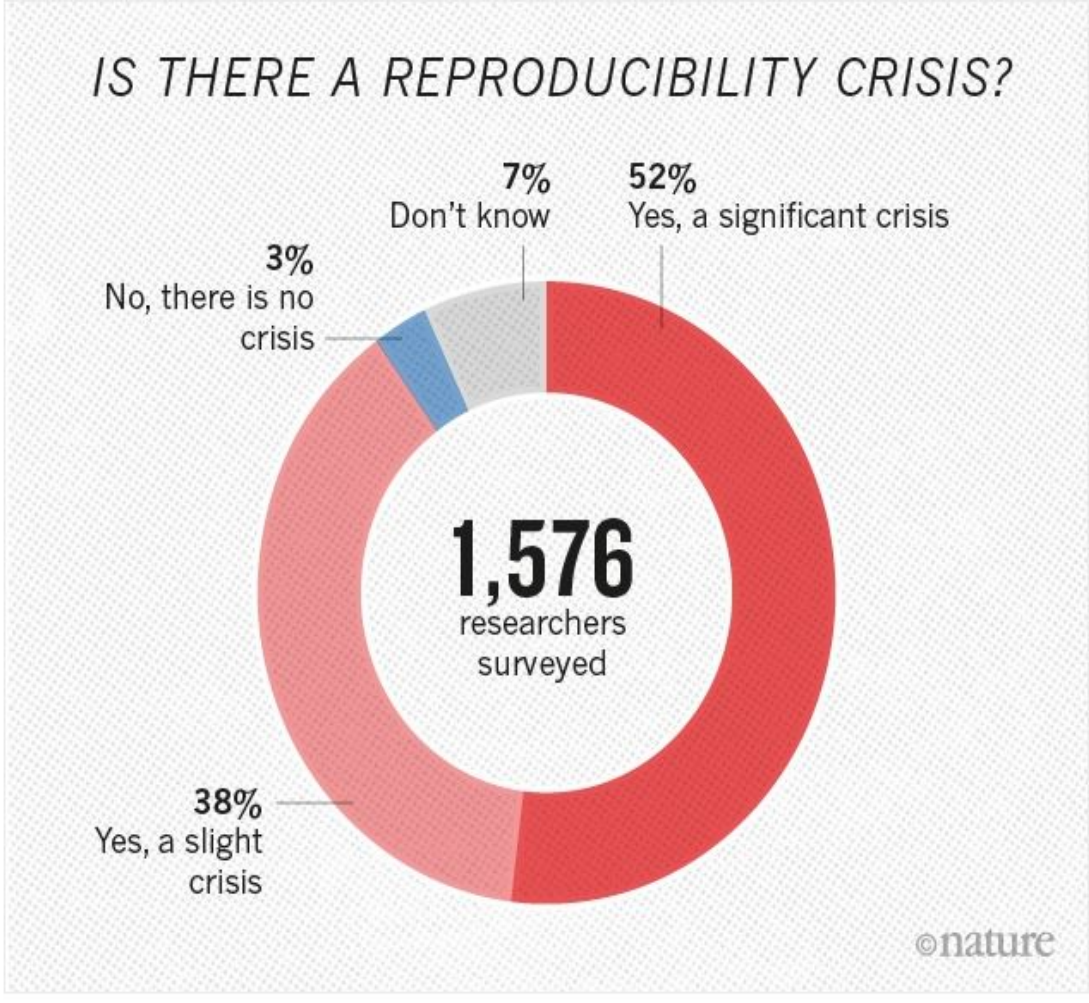


Image credit: Jeff Leek

How do issues arise?

- People leave the business
- People move internally
- People mess with the data

Why does it matter?



Why does it matter?

- Getting the same answer to the same question
- Inspires confidence in the result
- Improves efficiency and reduces effort

Leaving a trail

How can we improve reproducibility?

Number of ways including:

- Properly structuring projects
- Making things relative
- Building understandable code
- Packaging up the final products

Project structures

- Set up your analysis as a project, to help establish paths
- Build folders around this central file to:
  - Improve navigation
  - Make it modular
  - Allow it to be packaged

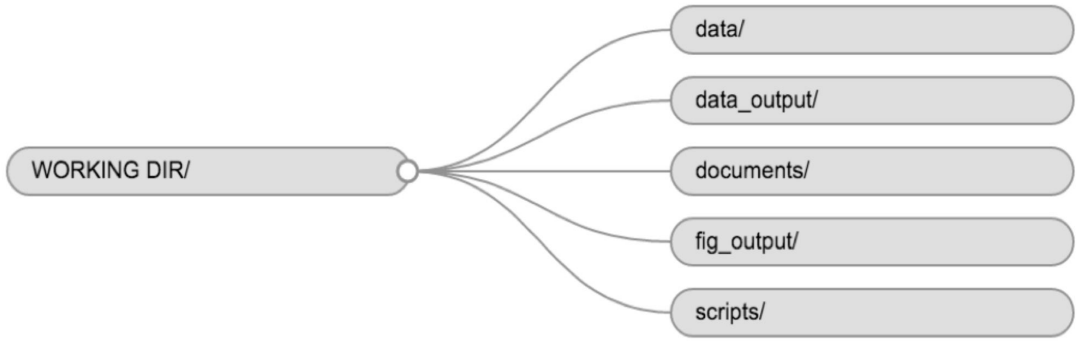


Image credit: Data Carpentries

Relative paths

```r
library(here)

setwd("A/Path/No/One/Else/Can/Use/Muahahahahaha")

a = read_csv("file_x.csv")

b = read_csv(here("data", "file_x.csv"))
```

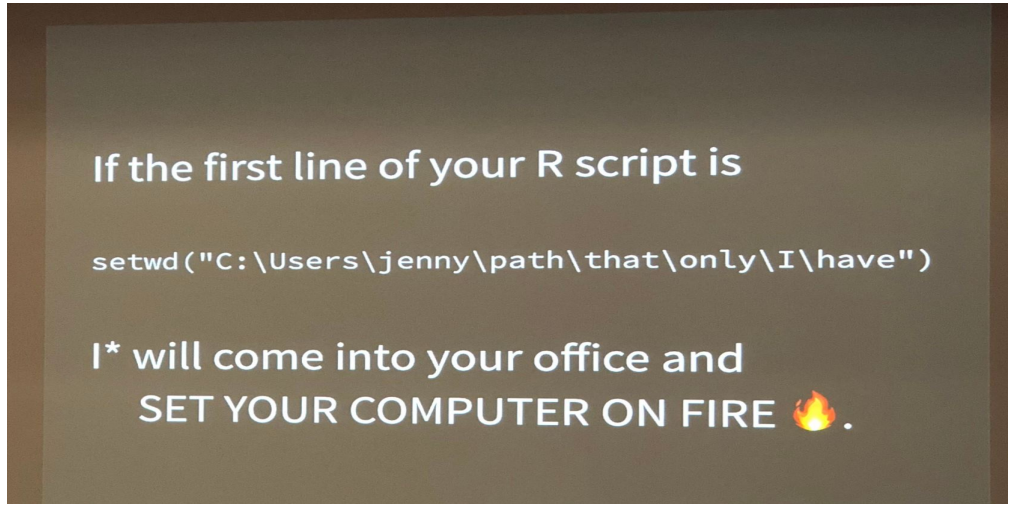
```python
import pandas as pd
from pathlib import Path

x = pd.read_csv("A/Path/For/My/Machine/Only")

data_folder = Path("data")

file_to_open = data_folder / "raw_data.txt"

y = pd.read_csv(file_to_open)
```



Building understandable code

Building understandable code

- Peoples taste develops faster than their ability - refactor!
- Use comments for business logic
- Use functions and make them modular

Gives you more maintainable, readable code that will persist and can be reused

Image credit: Jenny Bryan

Packaging up the products - Docker and Binder

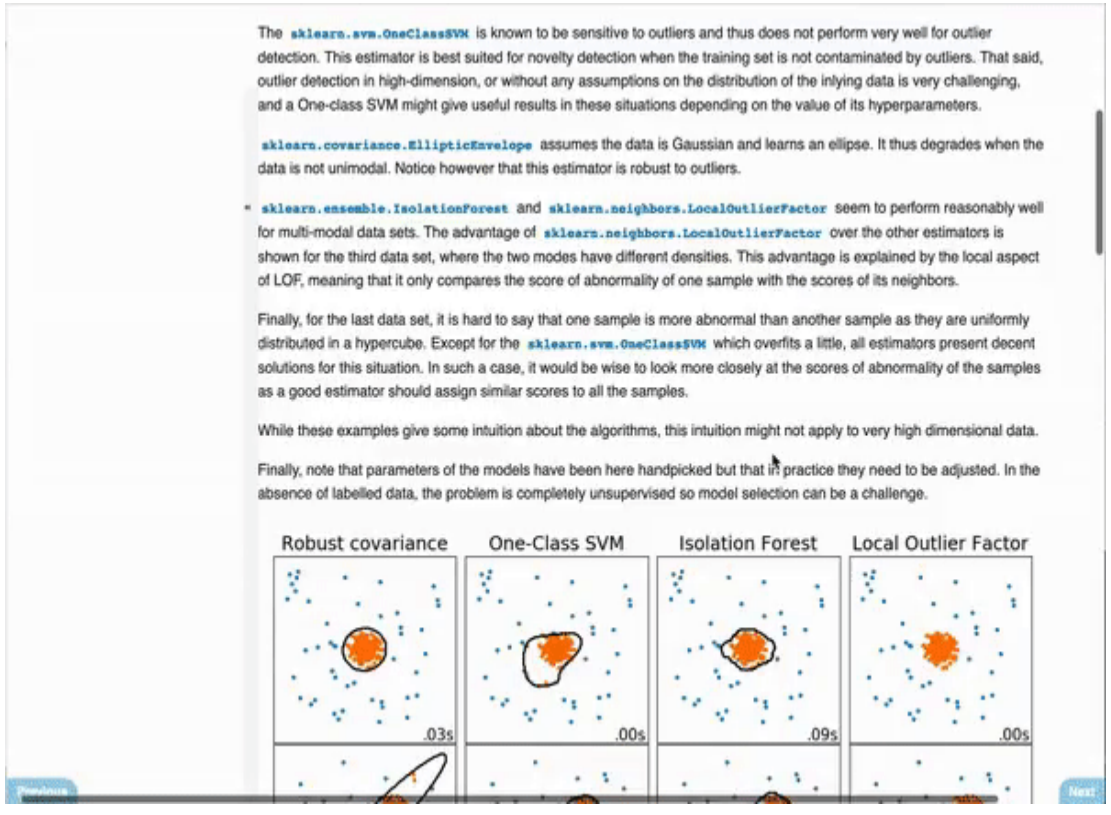


Image credit: Tim Head

Being on the same page

- Good analytics combines technical expertise and business knowledge
- Unlikely that one person will have both for every analysis
- As such, being able to utilise the same toolset is important

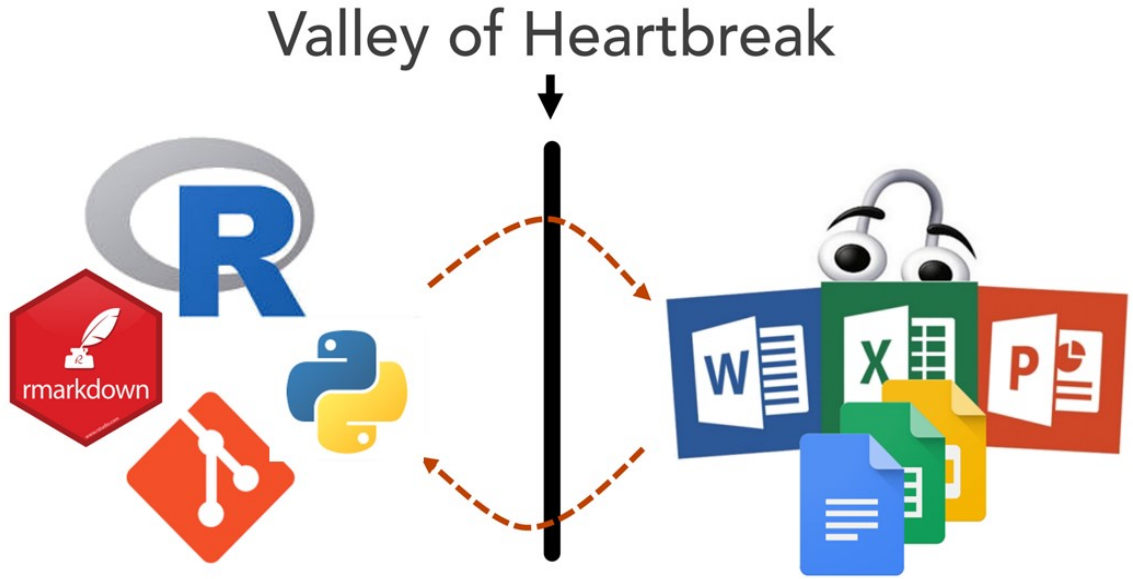


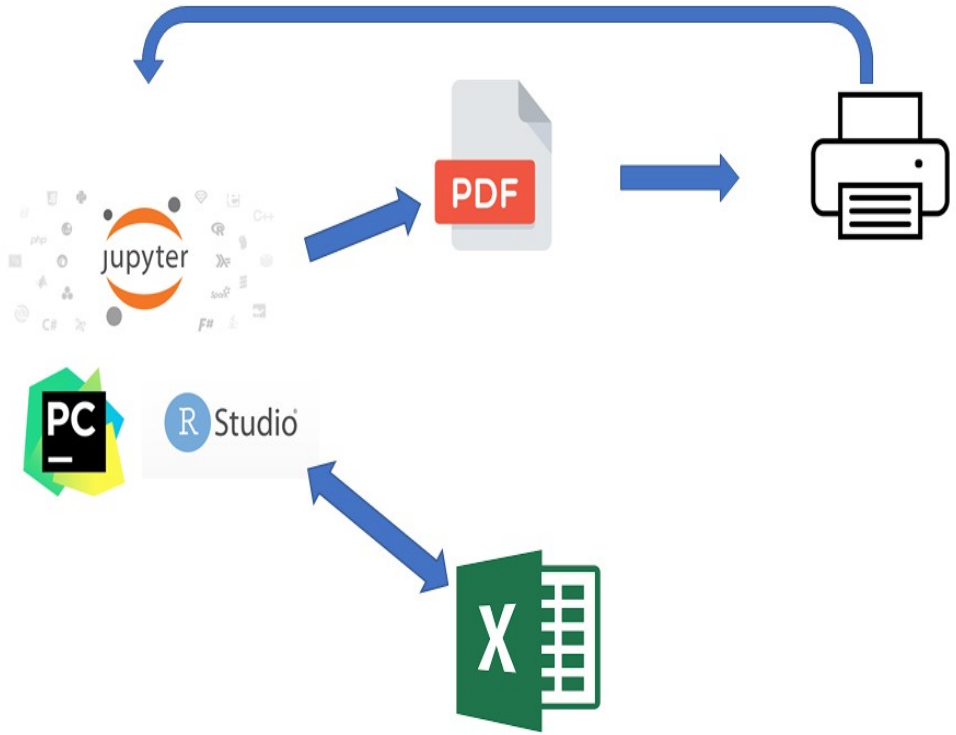
Image credit: Noam Ross

Why not just a hammer?

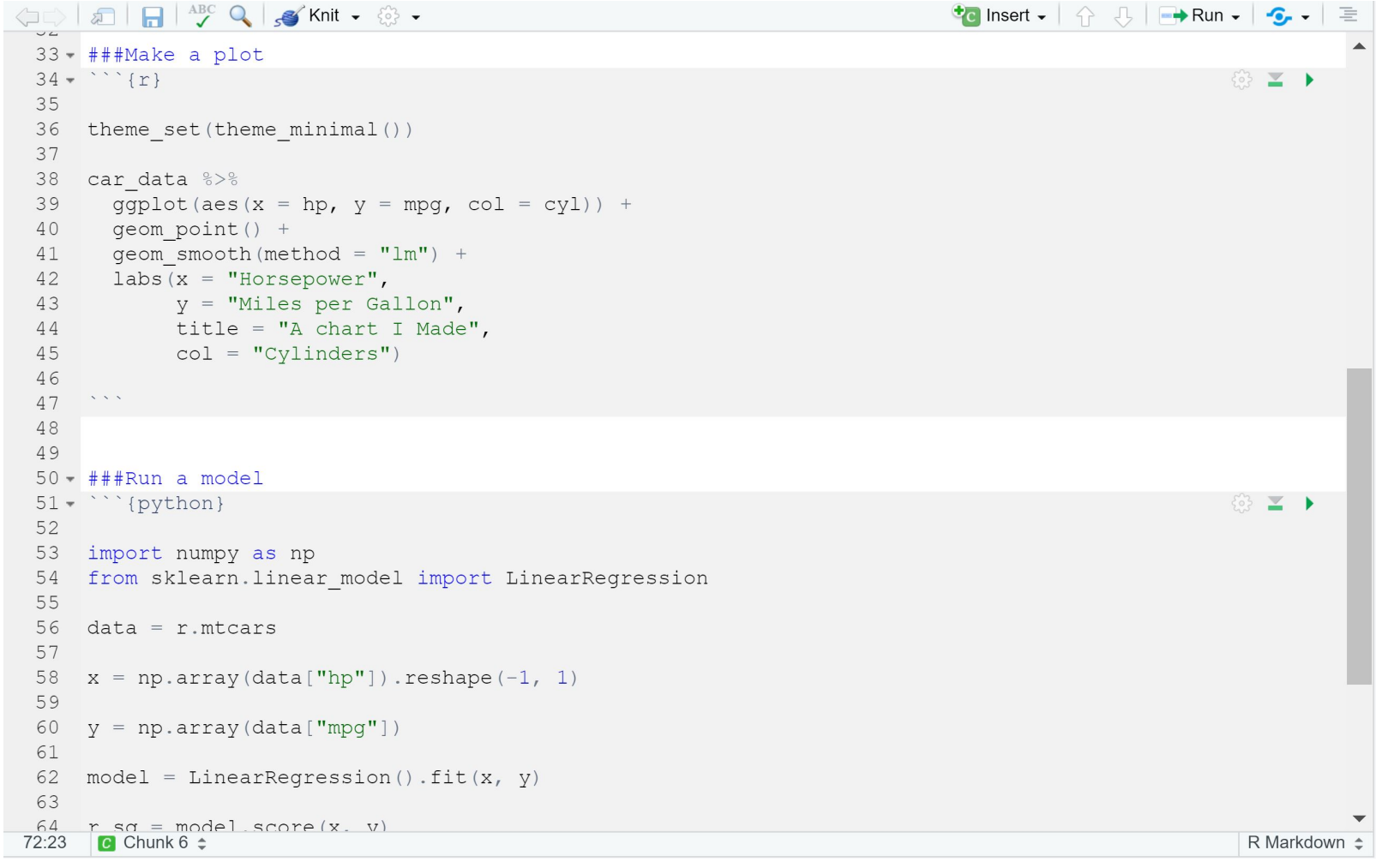
Because people like their tools

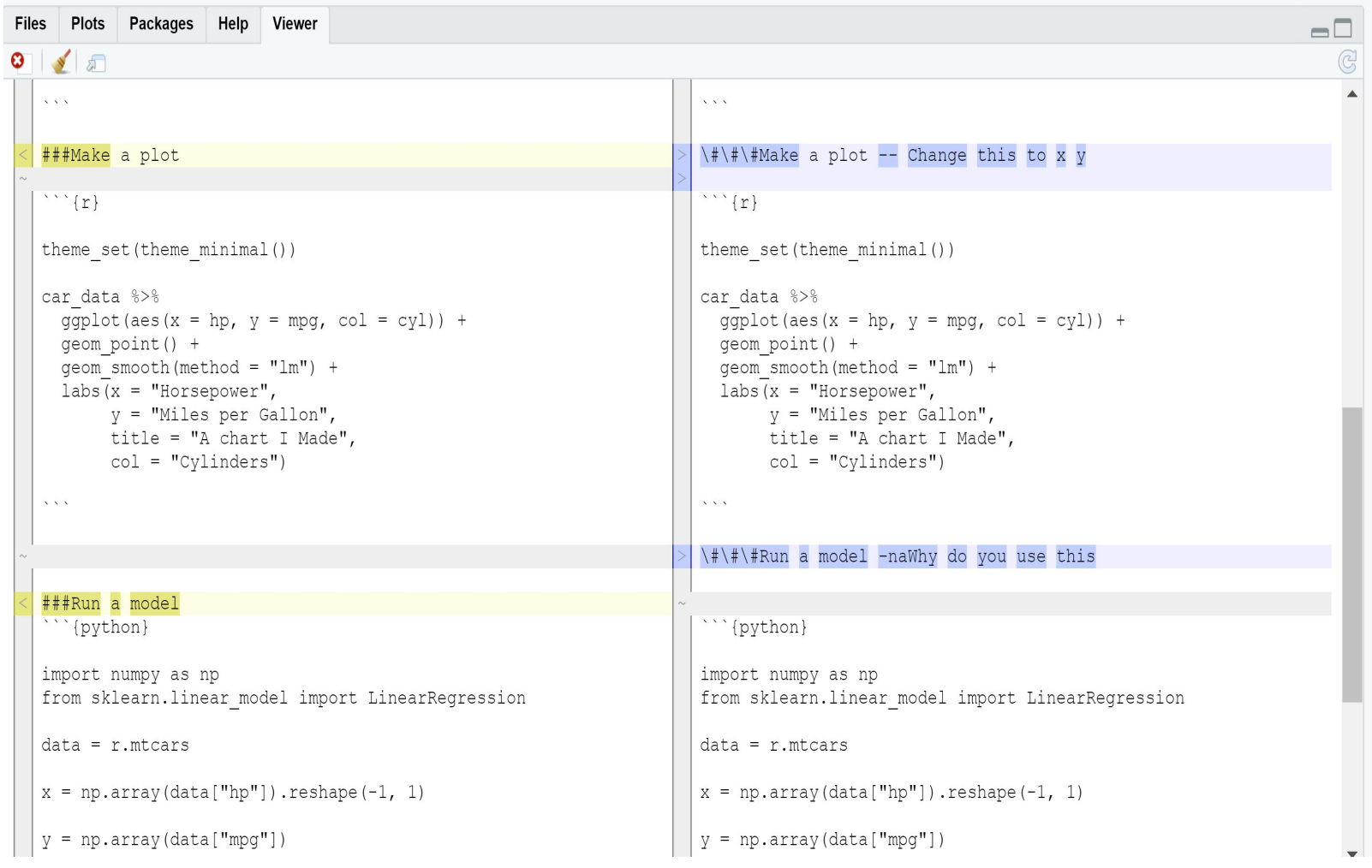
Image credit: Chris Albon

Current ways of working



A potential solution?




Files
Plots
Packages
Help
Viewer
```
###Make a plot
```{r}
theme_set(theme_minimal())
car_data %>%
ggplot(aes(x = hp, y = mpg, col = cyl)) +
geom_point() +
geom_smooth(method = "lm") +
labs(x = "Horsepower",
y = "Miles per Gallon",
title = "A chart I Made",
col = "Cylinders")
```
###Run a model
```{python}
import numpy as np
from sklearn.linear_model import LinearRegression
data = r.mtcars
x = np.array(data["hp"]).reshape(-1, 1)
y = np.array(data["mpg"])
```
\#\#\#Make a plot -- Change this to x y
\#\#\#Run a model -naWhy do you use this
```