ECPR Methods Summer School: Automated Collection of Web and Social Data

Pablo Barberá

School of International Relations
University of Southern California
pablobarbera.com

Networked Democracy Lab
www.netdem.org

Course website:
github.com/pablobarbera/ECPR-SC103

→

Shift in communication patterns

How can we collect web and social data to answer social science questions?

Course outline

1. Scraping data from the web
   - Key tools for webscraping
   - Tables; web data in unstructured format
2. Working with APIs
   - How to build an http request
   - Interacting with newspapers' APIs
3. Collecting social media data
   - Twitter: streaming and static data
   - Facebook: posts on public pages
4. New types of data
   - Text as data methods
   - Social network analysis
5. Advanced topics
   - Parsing data in PDF format
   - Data manipulation

Hello!

About me

- Assistant Professor in Computational Social Science at the London School of Economics as of January 2018
- Currently Assistant Prof. at Univ. of Southern California
- Director, Networked Democracy Lab, `netdem.org`
- PhD in Politics, New York University (2015)
- Data Science Fellow at NYU, 2015–2016
- My research:
  - Social media and politics, comparative electoral behavior, corruption and accountability
  - Social network analysis, Bayesian statistics, text as data methods
  - Author of R packages to analyze data from social media
- Contact:
  - `pbarbera@usc.edu`
  - `www.pablobarbera.com`

Juraj Medzihorsky

- Post-doc at the V-Dem Institute at the University of Gothenburg as of August 2017
- Currently post-doc at CEU
- PhD in political science, CEU (2015)
- Research interests:
  - Mixture models, categorical data analysis, measurement models, Bayesian statistics
  - Elections and assemblies
- Contact:
  - juraj.medzihorsky@gmail.com

Your turn!

1. Name?
2. Affiliation?
3. Research interests?
4. Previous experience with R?
5. Why are you interested in this course?

Course philosophy

How to learn the techniques in this course?

- Lecture approach: not ideal for learning how to code
- You can only learn by doing.

→ We will cover each concept three times during each session

1. Introduction to the topic (20-30 minutes)
2. Guided coding session (30-40 minutes)
3. Coding challenges (30 minutes)

- You're encouraged to continue working on the coding challenges after class. Solutions will be posted the following day.
- Additional questions? We can arrange one-on-one meetings after class

Course logistics

ECTS credits:

- Attendance: 2 credits (pass/fail grade)
- Submission of at least 3 coding challenges: +1 credit
  - Due before beginning of following class via email
  - Only applies to challenge 2 of the day
  - Graded on a 100-point scale
- Submission of class project: +1 credit
  - Due by August 20th
  - Goal: collect and analyze data from the web or social media
  - 10 pages max (including code) in Rmarkdown format
  - Graded on a 100-point scale

If you wish to obtain more than 2 credits, please indicate so in the attendance sheet

Social event

Save the date:

Wednesday Aug. 2nd, 6pm

Location TBA

Why we're using R

- Becoming *lingua franca* of statistical analysis in academia
- What employers in private sector demand
- It's free and open-source
- Flexible and extensible through *packages* (over 10,000 and counting!)
- Powerful tool to conduct automated text analysis, social network analysis, and data visualization, with packages such as quanteda, igraph or ggplot2.
- Command-line interface and scripts favors reproducibility.
- Excellent documentation and online help resources.

R is also a full programming language; once you understand how to use it, you can learn other languages too.

RStudio Server

RStudio

File Edit Code View Project Workspace Plots Tools Help

Go to file/function Project: (None)

diamondPricing.R* | formatPlot.R | diamonds

Source on Save Source

```
1  library(ggplot2)
2  source("plots/formatPlot.R")
3
4  View(diamonds)
5  summary(diamonds)
6
7  summary(diamonds$price)
8  aveSize <- round(mean(diamonds$carat), 4)
9  clarity <- levels(diamonds$clarity)
10
11 p <- qplot(carat, price,
12            data=diamonds, color=clarity,
13            xlab="Carat", ylab="Price",
14            main="Diamond Pricing")
15
```

15:1 (Top Level) R Script

Console ~/

```
       x                y                z
 Min.   : 0.000   Min.   : 0.000   Min.   : 0.000
 1st Qu.: 4.710   1st Qu.: 4.720   1st Qu.: 2.910
 Median : 5.700   Median : 5.710   Median : 3.530
 Mean   : 5.731   Mean   : 5.735   Mean   : 3.539
 3rd Qu.: 6.540   3rd Qu.: 6.540   3rd Qu.: 4.040
 Max.   :10.740   Max.   :58.900   Max.   :31.800
> summary(diamonds$price)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    326     950    2401    3933    5324   18820
> aveSize <- round(mean(diamonds$carat), 4)
> clarity <- levels(diamonds$clarity)
> p <- qplot(carat, price,
+            data=diamonds, color=clarity,
+            xlab="Carat", ylab="Price",
+            main="Diamond Pricing")
>
> format.plot(p, size=24)
>
```

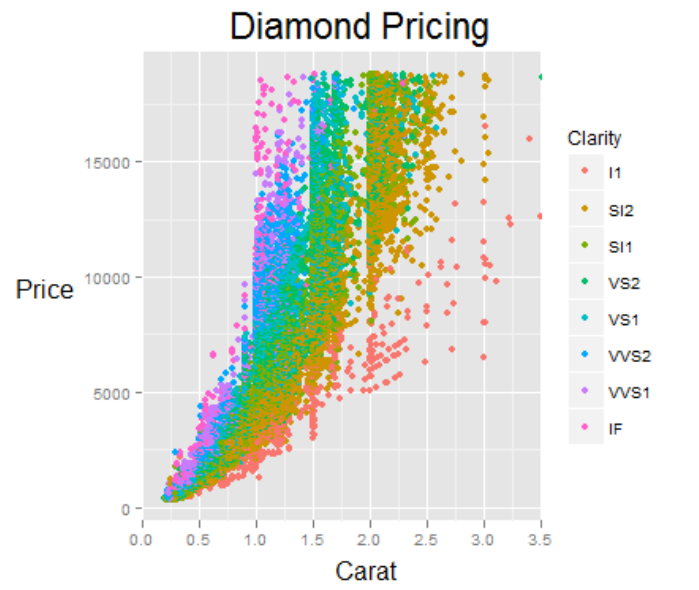
Workspace | History

Load Save Import Dataset Clear All

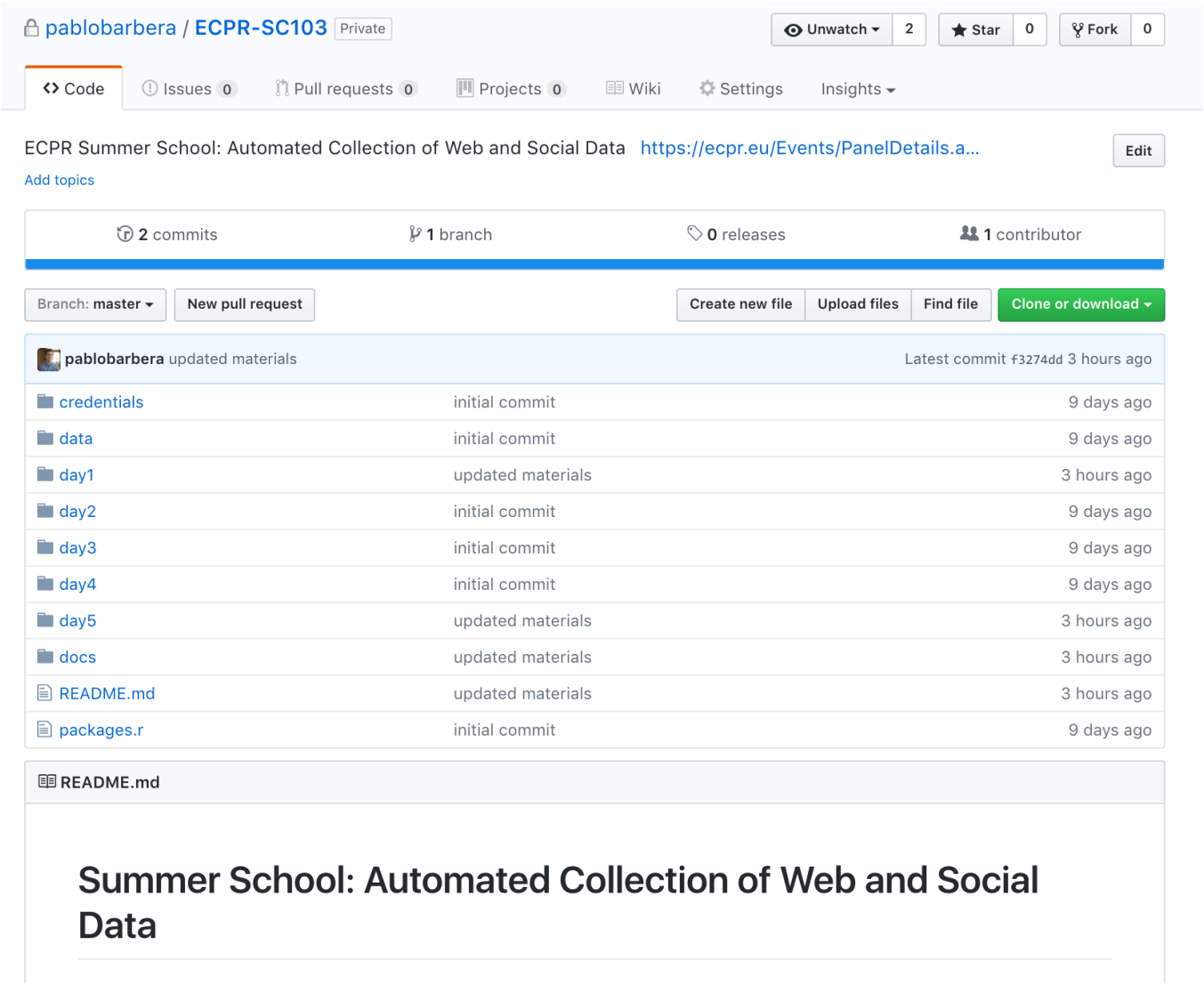
Data	
diamonds	53940 obs. of 10 variables
Values	
aveSize	0.7979
clarity	character[8]
p	ggplot[8]
Functions	
format.plot(plot, size)	

Files | Plots | Packages | Help

Zoom Export Clear All



Course website



github.com/pablobarbera/ECPR-SC103