Stat405

Project 2

Hadley Wickham

1. Regular expressions
2. More about stringr
3. Project 2 overview
4. Dates & times
5. Brainstorming

Recap

Your turn

Recall what each of the following regular expressions match:

"ba(na){2,}s"

"[a-z]+@[a-z]+\\.com"

".\\..\\."

```
"ba(na){2,}s"
```
ba, followed by na repeated 2 or more times, followed by s

```
"[a-z]+@[a-z]+\\.com"
```
one or more letters, followed by @, followed by one or more letters, followed by .com

```
".\\..\\."
```
any character, followed by ., followed by any character

String	Regexp	Matches
"[abc]"	[abc]	a, b, or c
"[a-c]"	[a-c]	a, b, or c
"[ac-"]	[ac-]	a, c, or -
"[ae-g.]	[ae-g.]	a, e, f, g, or .
"[^abc]"	[^abc]	Not a, b, or c
"[^a-c]"	[^a-c]	Not a, b, or c
"[ac^]"	[ac^]	a, c, or ^

String	Regexp	Matches
"^a"	^a	a at start of string
"a$"	a$	a at end of string
"^a$"	^a$	complete string = a
"\\$a"	\$a	$a

stringr

Function	Parameters	Result
`str_detect`	string, pattern	logical vector
`str_locate`	string, pattern	numeric matrix
`str_extract`	string, pattern	character vector
`str_replace`	string, pattern, **replacement**	character vector
`str_split_fixed`	string, pattern	character matrix

Single (output usually vector or matrix)	Multiple (output usually a list)
`str_detect`	
`str_locate`	`str_locate_all`
`str_extract`	`str_extract_all`
`str_replace`	`str_replace_all`
`str_split_fixed`	`str_split`

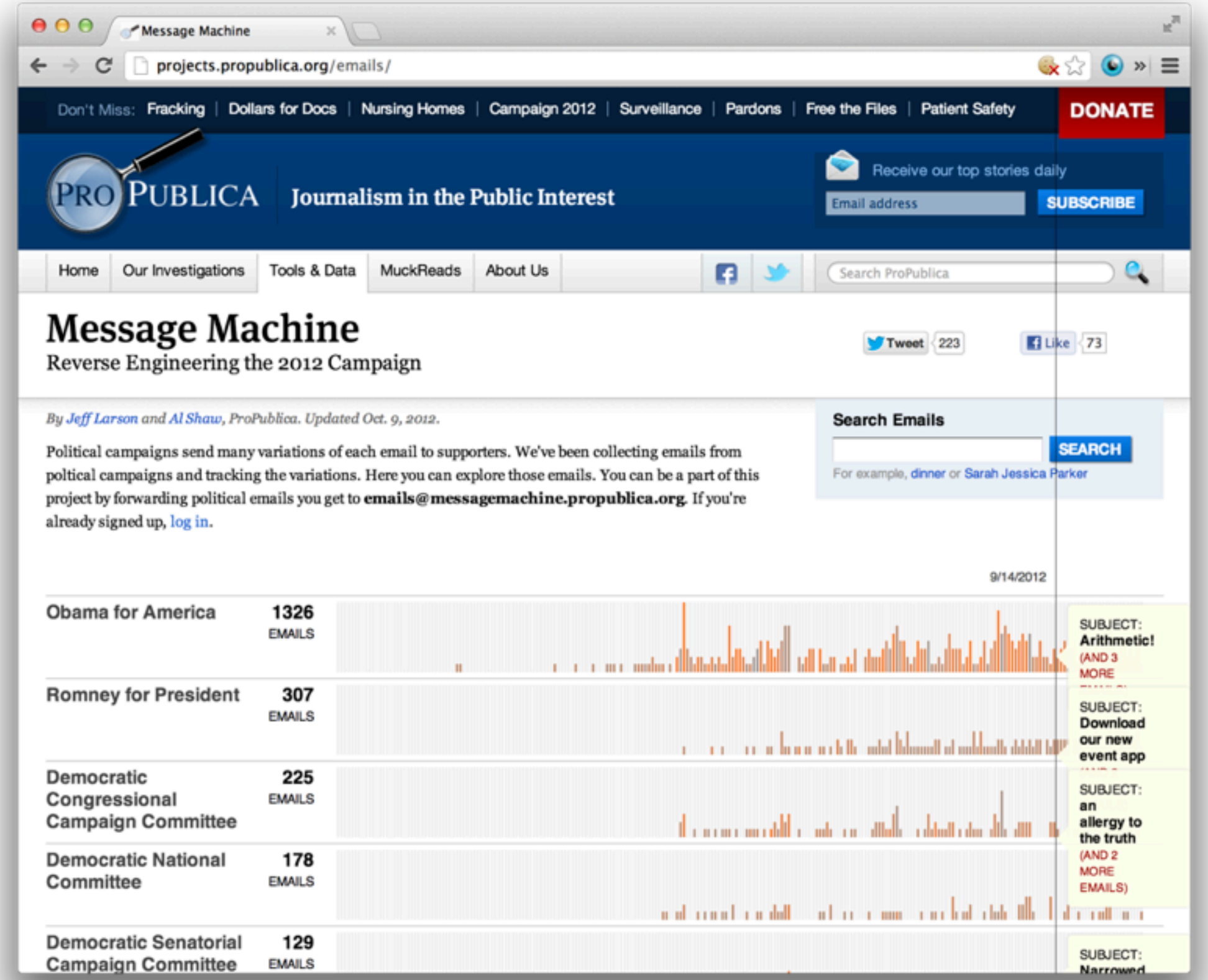
More info at:

http://vita.had.co.nz/papers/stringr.html

Other useful links

- http://en.wikibooks.org/wiki/R_Programming/Text_Processing

Project 2


Message Machine
projects.propublica.org/emails/
Don't Miss: Fracking | Dollars for Docs | Nursing Homes | Campaign 2012 | Surveillance | Pardons | Free the Files | Patient Safety
DONATE
PRO PUBLICA
Journalism in the Public Interest
Receive our top stories daily
Email address
SUBSCRIBE
Home
Our Investigations
Tools & Data
MuckReads
About Us
Search ProPublica
Message Machine
Reverse Engineering the 2012 Campaign
Tweet 223
Like 73
By Jeff Larson and Al Shaw, ProPublica. Updated Oct. 9, 2012.
Political campaigns send many variations of each email to supporters. We've been collecting emails from poltical campaigns and tracking the variations. Here you can explore those emails. You can be a part of this project by forwarding political emails you get to emails@messagemachine.propublica.org. If you're already signed up, log in.
Search Emails
SEARCH
For example, dinner or Sarah Jessica Parker
9/14/2012
Obama for America 1326 EMAILS
SUBJECT: Arithmetic! (AND 3 MORE
Romney for President 307 EMAILS
SUBJECT: Download our new event app
Democratic Congressional Campaign Committee 225 EMAILS
SUBJECT: an allergy to the truth (AND 2 MORE EMAILS)
Democratic National Committee 178 EMAILS
Democratic Senatorial Campaign Committee 129 EMAILS
SUBJECT: Narrowed

Data

- `user.csv`: information about each person
- `email.csv`: the contents of each email variant
- `email-user.csv`: which people recieved which emails
- `explore.r`: some code to help you get started

Your turn

Run the code in explore.r. What do the numbers on the y-axes mean?

THIS IS IMPORTANT.

Dates & times

```
# These are the absolute essentials.
# We'll talk more about dates next week

install.package("lubridate")
library(lubridate)
```

```
# Strings -> dates
email$first_seen <- ymd_hms(email$first_seen)
ymd("2010-01-01")
dmy("01/01/2010")
mdy("10 10 2010")
```

```
# Extracting date components

wday(email$first_seen)
mday(email$first_seen)
yday(email$first_seen)
month(email$first_seen)
year(email$first_seen)
```

```
# Rounding

round_date(email$first_seen, "day")
round_date(email$first_seen, "month")
round_date(email$first_seen, "year")
```

Your turn

Create a plot that shows the number of variants sent each week day by hour of the day.

```
emails$wday <- wday(email$first_seen)
emails$hour <- hour(email$first_seen)
wh <- count(emails, c("wday", "hour")

qplot(wday, hour, data = wh, size = freq)
```

String practice

Your turn

Write a regular expression to match dollar amounts.

```
dollars <- str_extract_all(email$clean_text,
  "\\$[0-9,]+")

one <- dollars[[3585]]
```

Your turn

Given a character vector that contains all of the amounts in an email, how could you find the maximum amount?

(Hint: what do you need to remove so that `as.numeric` works)

```
str_replace_all(one, "[$,]", "")
as.numeric(str_replace_all(one, "[$,]", ""))
max(as.numeric(str_replace_all(one, "[$,]", "")))

max_money <- function(x) {
  max(as.numeric(str_replace_all(x, "[$,]", "")))
}
max_money(character())
max_money <- function(x) {
  if (length(x) == 0) return(NA)
  max(as.numeric(str_replace_all(x, "[$,]", "")))
}
max_money(character())
```

```
email$max_dollars <- NA
for(i in seq_along(dollars)) {
  email$max_dollars[i] <- max_money(dollars[[i]])
}
```

Brainstorming

Your turn

Break into your project teams and start brainstorming some potential ideas.

I’ll share some of my ideas at the end of class.

Thursday

No class!

Homework for next week will be to prepare a 2-3 page project plan.