Wrangling and Visualizing Data in R

Amit_Levinson
Amit_Levinson
amitlevinson.com

About **today**

We'll talk about

- SPSS: How *I* learned to work with data
- Some good alternatives
- R as a **recommended** alternative
  - What is R
  - Some cool plotting features
  - Why I fell in love ♡

We won't talk about

- Which alternative is better
- Practical code
- coconut, we definitely won't talk about that (I mean who likes it? 😜)

Anything said is based on my personal views and experience working with R

About me

- Graduate student for Sociology & Anthropology @ Ben-Gurion University of the Negev
- Research assistant for Dr. Jeniffer Oser researching online & offline political participation
- Political activist who likes to disseminate data as a way of advocacy
- Using R for about 7 months

About SPSS

- Learned and used it in quantitative courses in my BA & MA

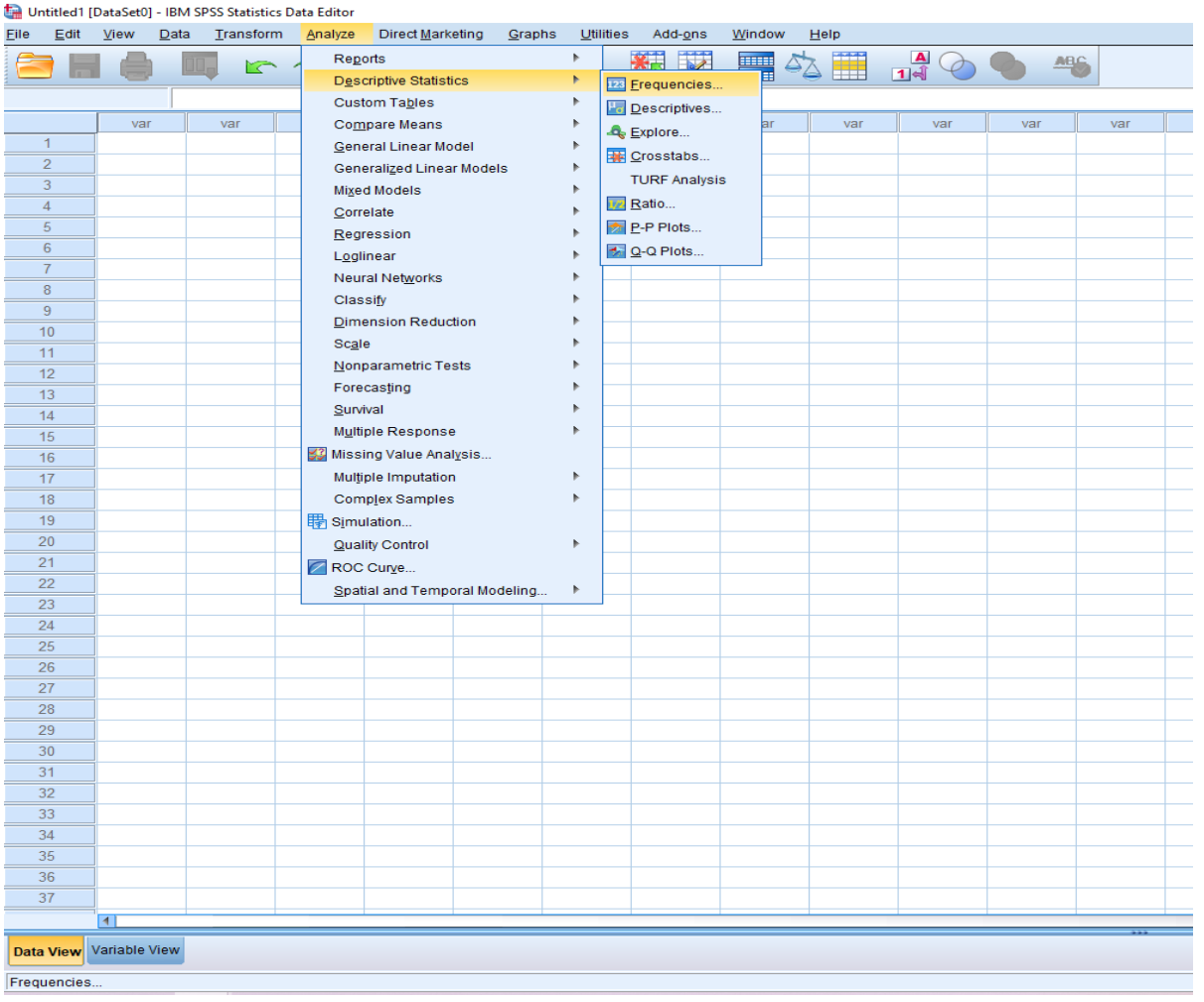
Pros

- Has a solid infrastructure (IBM)
- Many functions
- Our faculty uses it
- Knowledge of it is sometimes a demand in industry

Cons

- **Costs money**
- An **inefficient workflow**
- **Difficult to tidy data** in it
- Plots are nice (?), but you can **make nicer plots**.
- Its graphic user interface **(GUI) is overloaded**

SPSS... Remind me?




Ain't nobody got time for that!

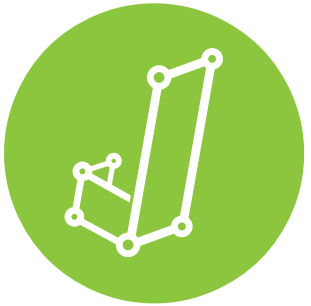
Quick Alternatives

Jamovi

Tableu

Power BI

JASP



R is...

- A "software environment for **statistical computing and graphics**" (r-project)
- **Free** to use
- Open source
- Has an **amazing community**

```
##
##  -----
## Did you mean are?
##  ------
##     \   ^__^
##      \  (oo)\ ________
##         (__)\         )\ /\
##              ||------w|
##              ||       ||
```

Some of the basics

Basics - Math operations

- You can do simple **calculations**:

```
1+3
```

```
## [1] 4
```

```
4^3
```

```
## [1] 64
```

Use objects to store vectors and operate on them:

```
x ← c(1:10)
```

```
mean(x)
```

```
## [1] 5.5
```

Basics - Analyzing text

- It's easy to manipulate and work with **text**
- We can use **reg**ular **exp**ressions (**regex**) to work out the magic
- For e.g, imagine you want to extract any word that doesn't have a vowel:

> "**Why** this is some random text with some words that don't have vowels such as **myth**, **shy**, or **gym**"

- We want to create an expression that captures everything **that isn't a vowel** and use that to filer:

```
words ← unlist(str_split("Why this is some random text with some wo
grep("^[^aeiou]+$", x= words, value = TRUE)
```

```
## [1] "Why"  "myth" "shy"  "gym"
```

Basics - Reading data

Read data from **online sources***

```
countries ← read_delim("https://perso.telecom-paristech.fr/eagan/cla
```

Let's have a look at our top 6 rows:

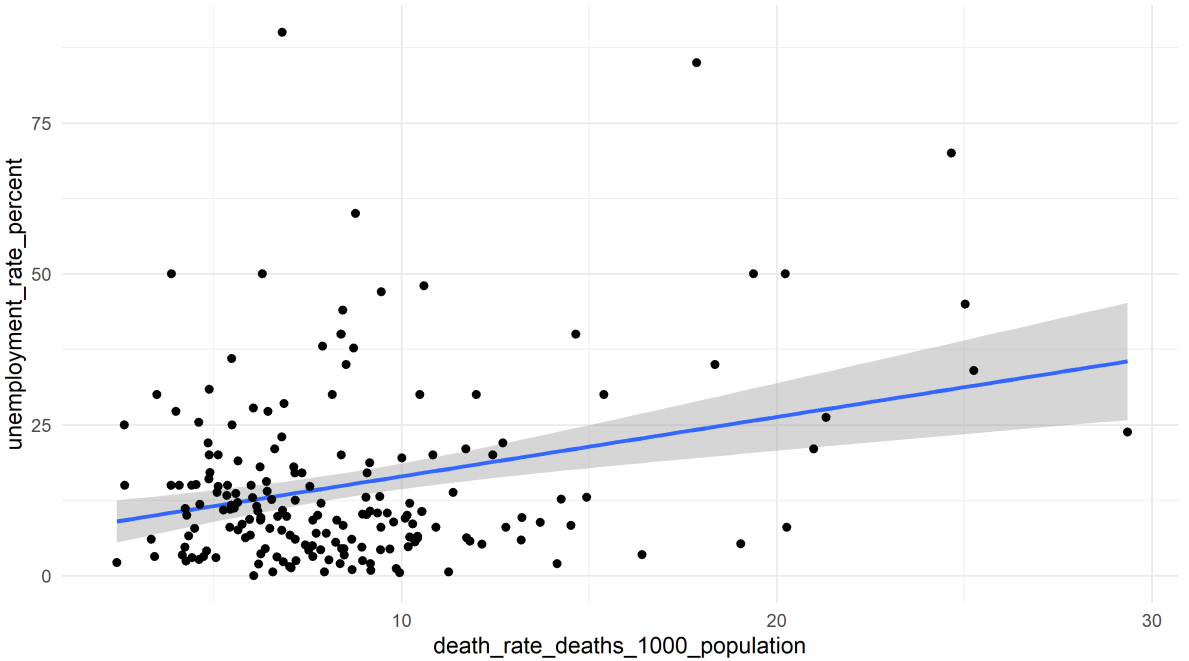
country	area_sq_km	birth_rate_births_1000_population	current_acc
Afghanistan	647500	47	
Akrotiri	123	NA	
Albania	28748	15	
Algeria	2381740	17	
American Samoa	199	23	
Andorra	468	9	

[*] Data from Project datasets

Basics - plot

R's simple plotting features:

```
ggplot(countries, aes(x = death_rate_deaths_1000_population, y= unemp
  geom_smooth(method = "lm")+
  geom_point()+
  theme_minimal()
```



Basics - reports

Rmarkdown's reproducible and **automated** work flow makes it easy to work with reports and documents:

For example this:

```
"The lowest GDP per capita is
`min(countries$gdp_per_capita)`
and the highest unemployment
rate is `max(countries$unemploym
The average birth rate for 1000
people is `mean(countries$birth_
births_1000_population, na.rm =
The correlation of unemployment
GDP per capita is cor(
countries$gdp_per_capita,
countries$unemployment_rate,
"complete.obs").
```

Will render this:

"The lowest GDP per capita is **400** and the highest unemployment rate is **90**. The average birth rate for 1000 people is **22.15.** The correlation of unemployment and GDP per capita is **-0.44**.

Let's look at some **cool stuff**

you can do with R

Rmarkdown Efficiency

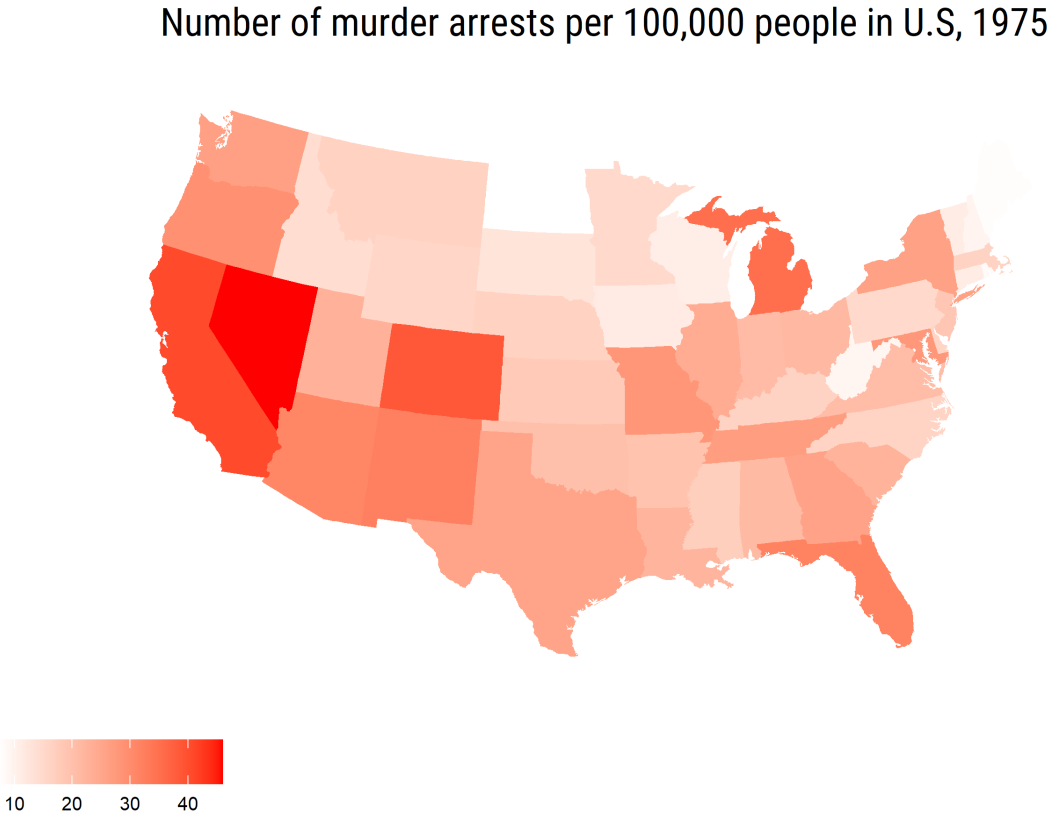
We can use code output inline with our text

No more ~~Copy+Paste~~ 😱

'Print' documents in one click

Maps

- You can make some neat and easy maps in R



Data: USArrests

Interactive maps

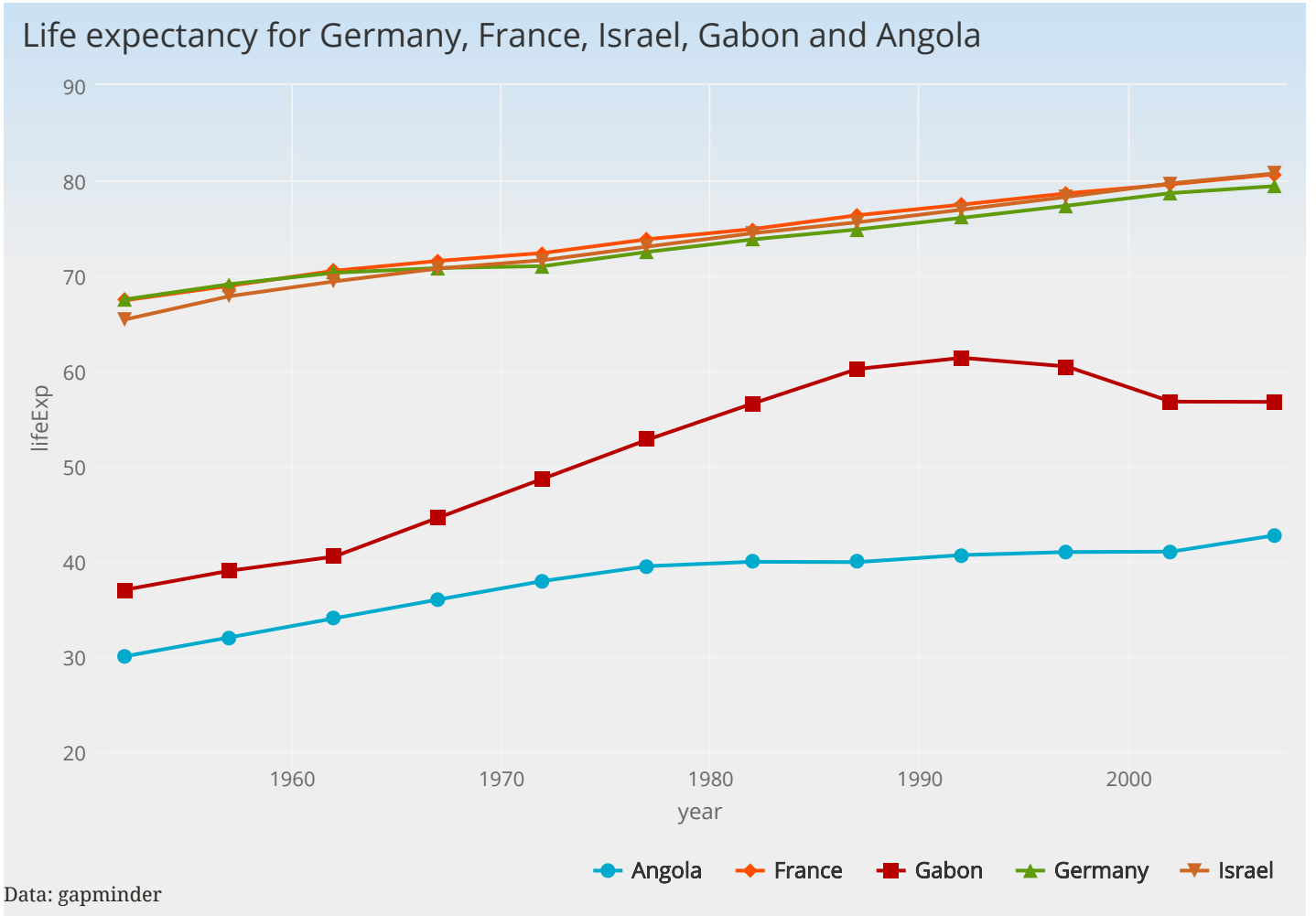
- When missiles are fired towards Israel
- And your city has open data such as bomb shelter locations

Interactive plots

Make interactive graphs with **{plotly}**

Data: gapminder

Or with **{highcharter}**



Animated plots

- Use with caution

```
library(gganimate)
```

```
chat_raw ← read_delim("chat.txt", delim = "-")
head(chat_raw)
```

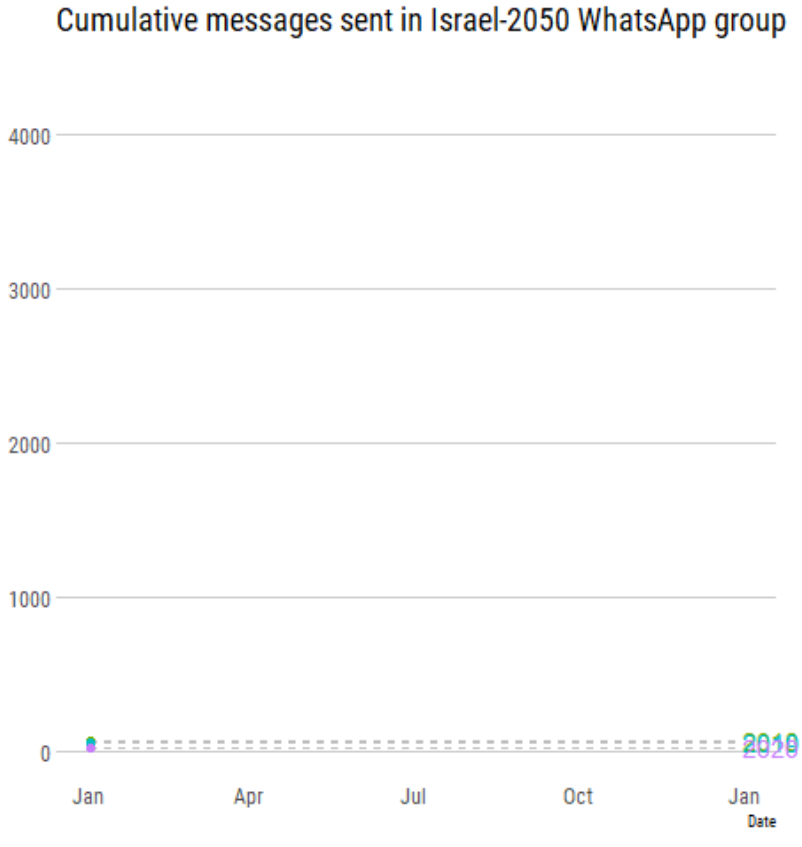
```
## # A tibble: 6 x 4
##   `04/08/2017, 12:50 `       ` Messages to this g~ to    `end encryption.
##   <chr>                      <chr>                 <chr> <chr>
## 1 "12/08/2016, 12:32 "       " +972 52"            374   "9319 created gro
## 2 "04/08/2017, 12:50 "       " +972 52"            374   "9319 added you"
## 3 "04/08/2017, 12:50 "       " +972 52"            374   "9319 added +972
## 4 "04/08/2017, 12:52 "       " +972 54"            760   "2588: ם וולקאם ו
## 5 "שמחים מאוד שהצטרפתם לכאן~  <NA>                  <NA>   <NA>
## 6 "המטרה של קבוצת הרשת היא ~  <NA>                  <NA>   <NA>
```

Data: WhatsApp group

Animated plots

- How does it work?

```
g + transition_reveal(date)
```



Let's talk some TwitteR

TwitteR

We can use the {rtweet} package:

- Search tweets containing a word (word, hashtag, etc)
- Get a user's list of friends
- Stream live tweets
- Get timelines from a user
- And more here...

TwitteR

Let's get the past tweets for some political candidates in the past elections*

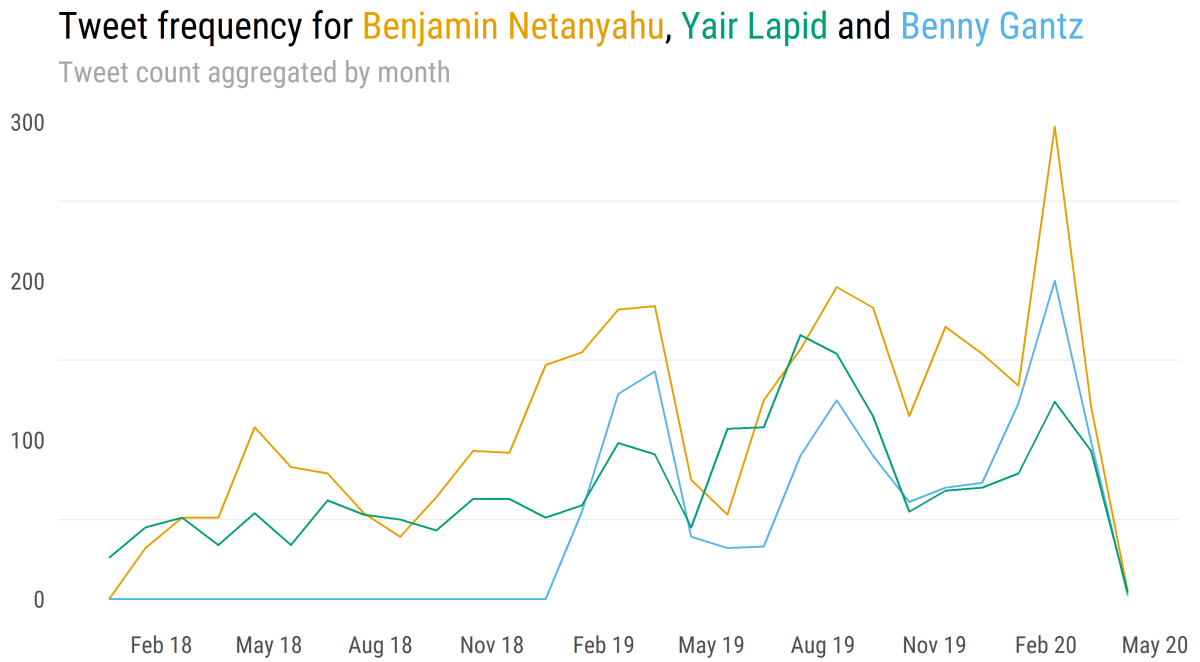
```
candidates_rtweet ← rtweet::get_timeline(c("netanyahu", "gantzbe", '
```

Which gives us a lot of information:

x
user_id
status_id
created_at
screen_name
text
source
display_text_width

[*]: Data collected on April 11, 2020.

Tweet frequency



Most favorited tweet

Benny Gantz

Most favorited tweet

~~Benny Gantz~~

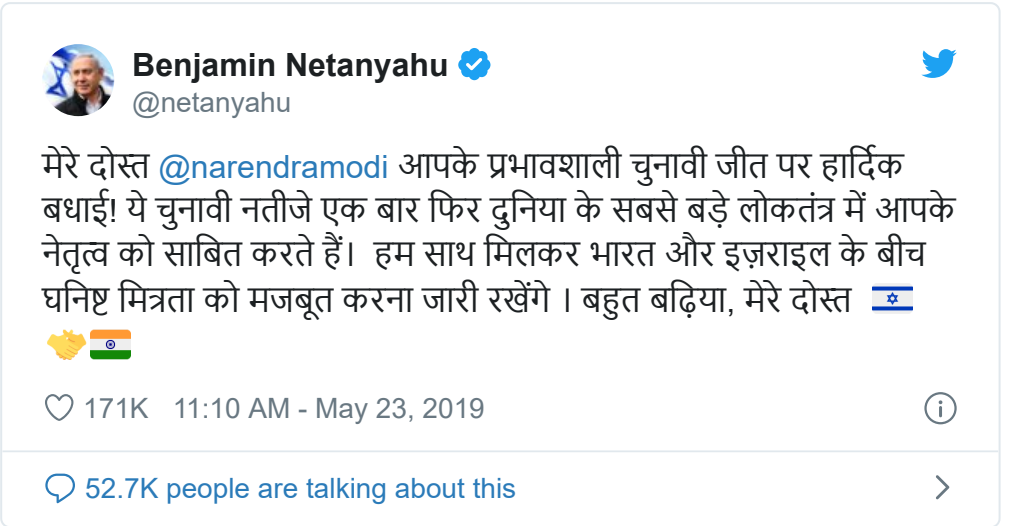
Yair Lapid

Most favorited tweet

~~Benny Gantz~~

~~Yair Lapid~~

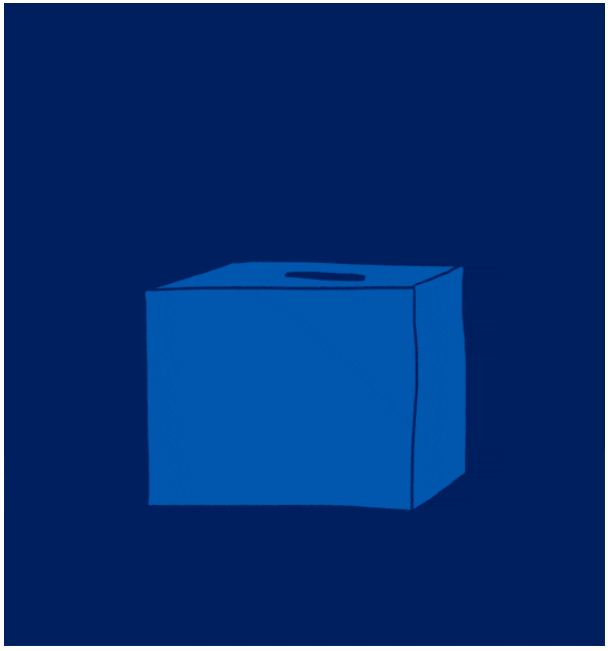
Benjamin Netanyahu



TwitteR

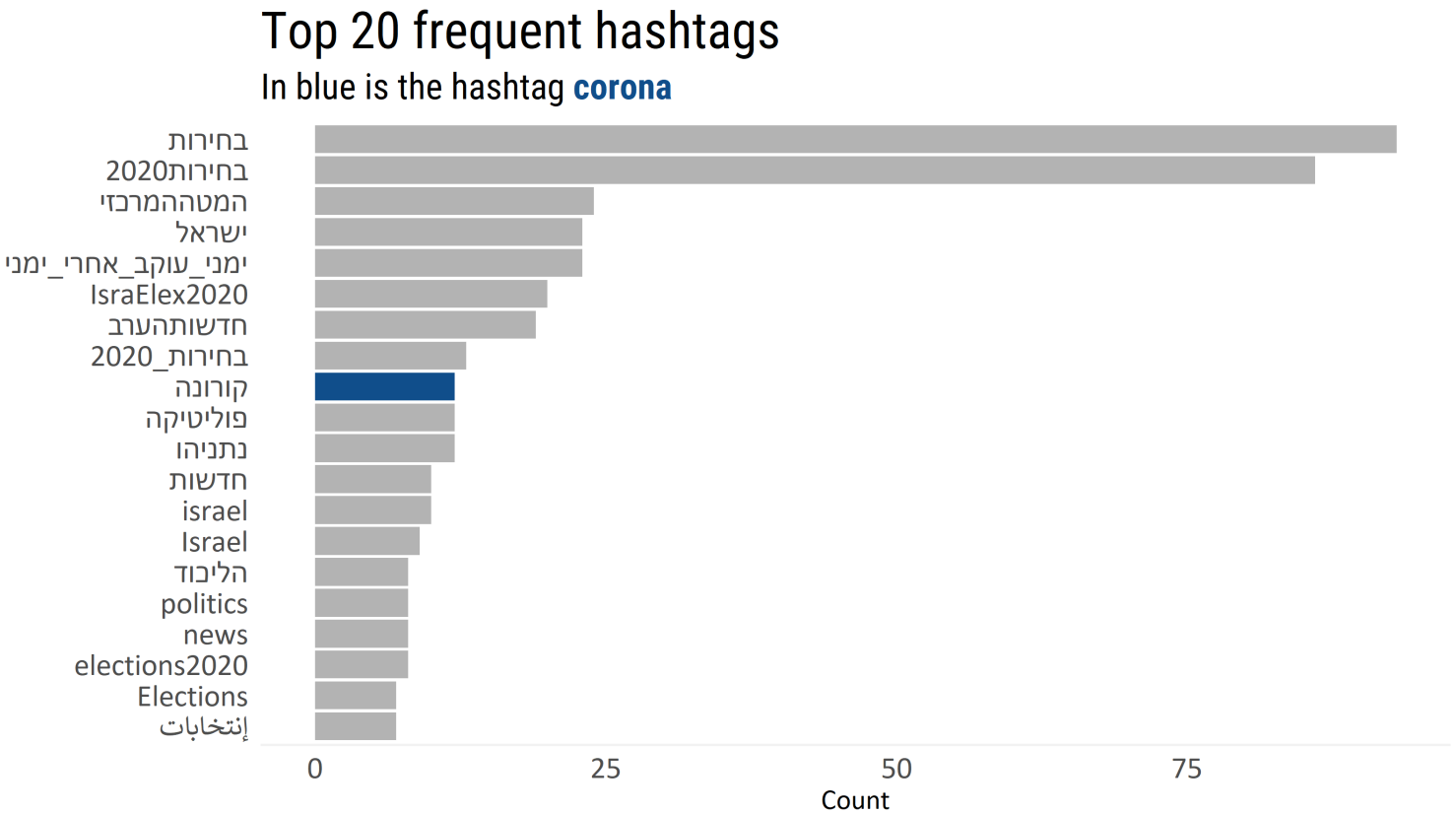
- We can also search on Twitter for a word or phrase, let's do that for 'בחירות' (elections):

```
elections ← search_tweets("בחירות", n = 25000, retryonratelimit = TR
```



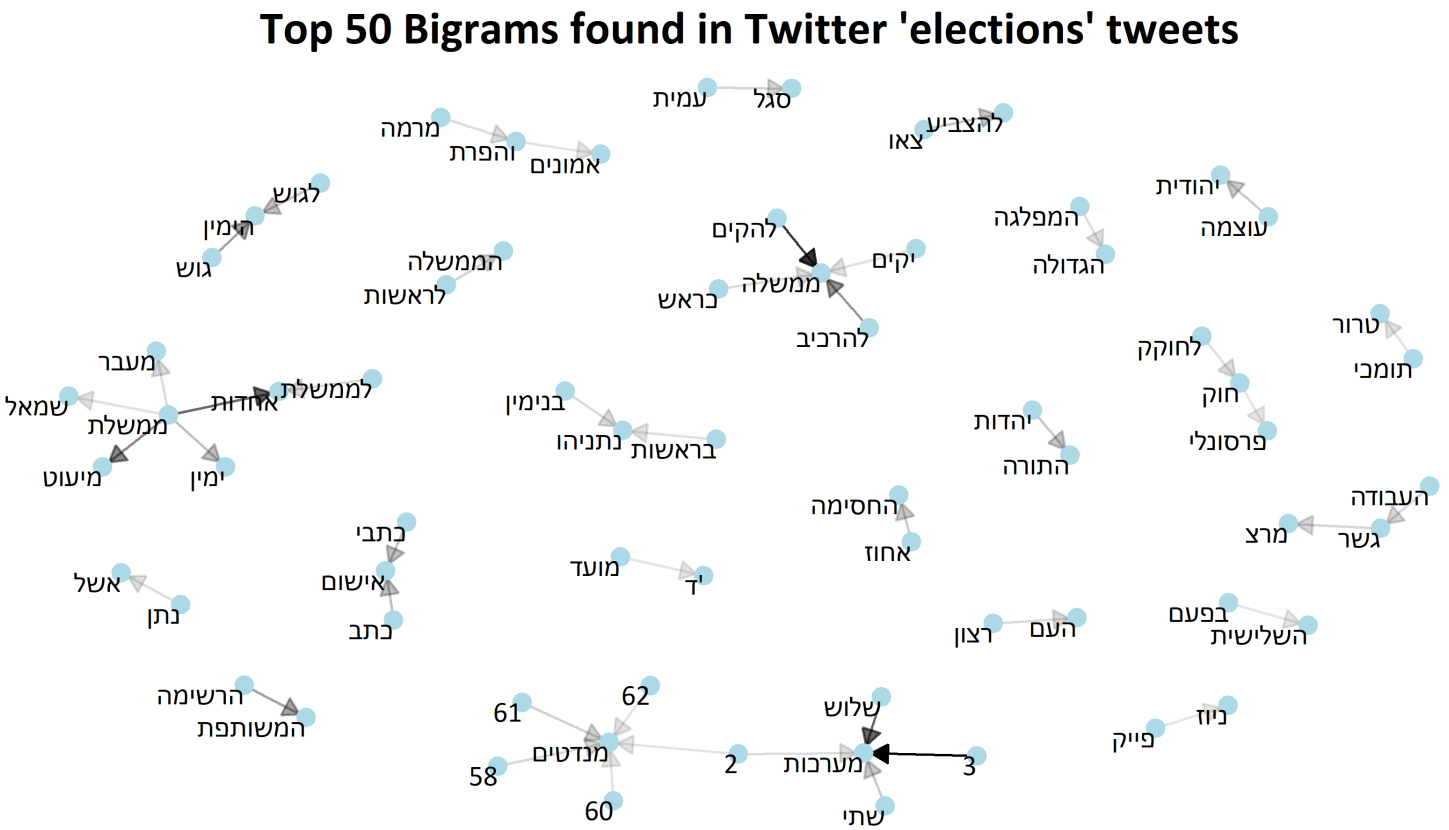
Election hashtags

- {rtweet} comes with a hashtag column containing only the hashtags 😮



Frequency of words?

- We could look at word cloud, bi-grams (2 words), trigrams, etc...



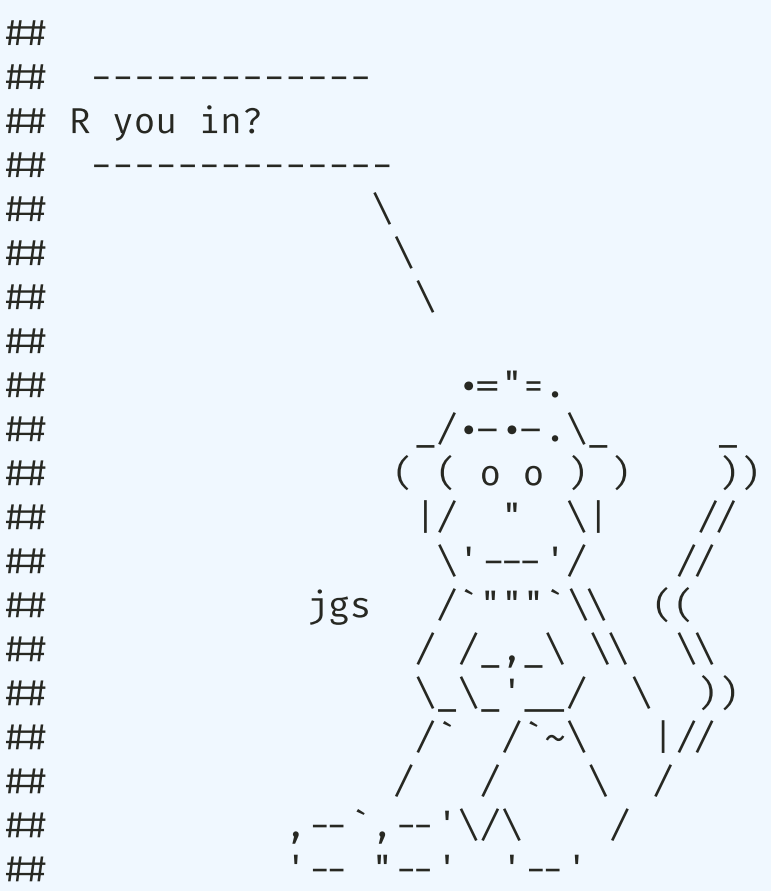
Final ~~slides~~ words

Why R?

- It's **free**
- It's **open source** where everyone and anyone can contribute
- It enabled me to **tackle quantitative questions** I was interested in
- It's an **all in one program**: Prepare data, analyze, visualize, report
- A **skill** sought after in industry
- THE COMMUNITY

The community!

- Israeli R community on **Facebook**
- R community on **Twitter**
- **Sharing** code
- **#Tidyteusday**
    - A weekly project for improving exploratory data analysis and visualizations


R you in?
jgs

There is so much more...

Packages

Websites

CV

Posters

interactive applications

Presentations (like this one)

Thank you!

And thanks to the many tutorials:

Yihui Xie

Allison Hill concise and elaborated versions

Zhi Yang

Garth Tarr

Garrick adenbuie Xaringanthemer