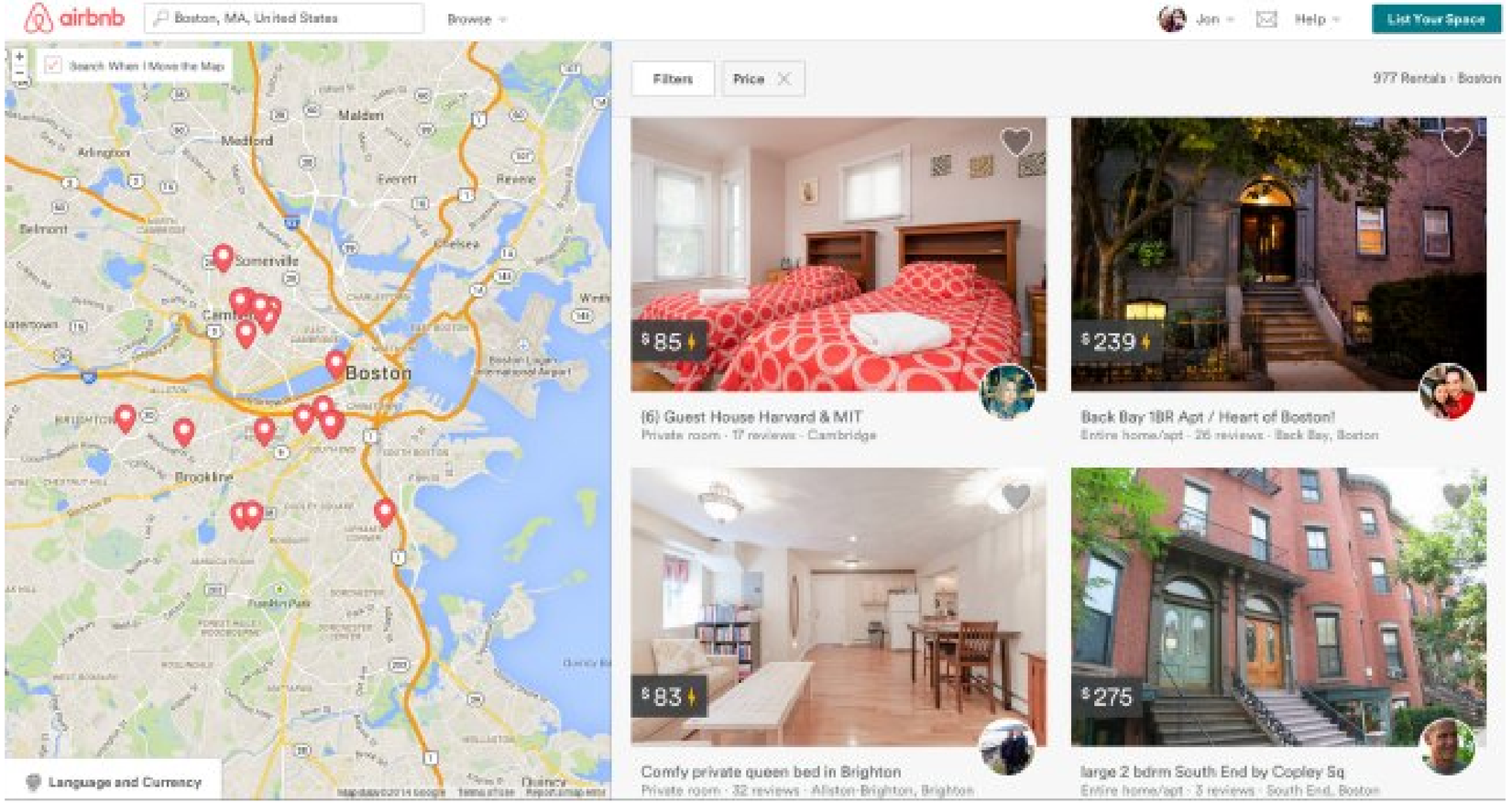
Refresher on the text mining workflow

SENTIMENT ANALYSIS IN R

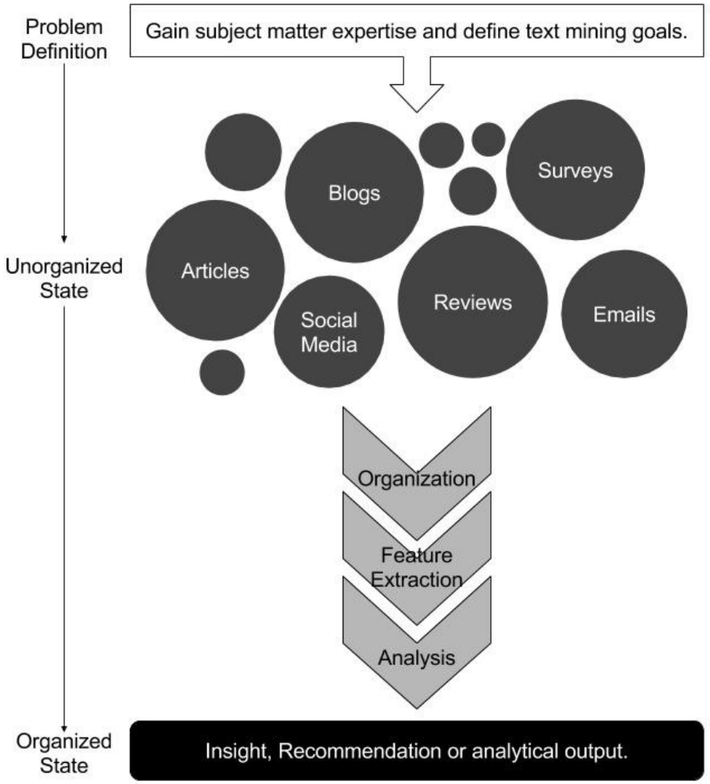
Ted Kwartler
Data Dude

So far ...

- `polarity()`
  - Valence shifters
- tidytext, dplyr, tidyr
  - bing, nrc, afinn
- Visualizations


airbnb
Boston, MA, United States
Browse
Jon
Help
List Your Space
Search When I Move the Map
Filters
Price
977 Rentals · Boston
Malden
Medford
Arlington
Everett
Revere
Belmont
Somerville
Cambridge
Boston
Brookline
Franklin Park
Quincy
Language and Currency
$85
(6) Guest House Harvard & MIT
Private room · 17 reviews · Cambridge
$239
Back Bay 1BR Apt / Heart of Boston!
Entire home/apt · 26 reviews · Back Bay, Boston
$83
Comfy private queen bed in Brighton
Private room · 32 reviews · Allston-Brighton, Brighton
$275
large 2 bdrm South End by Copley Sq
Entire home/apt · 3 reviews · South End, Boston

The text mining workflow



6 defined steps

1. Define the problem & specific goals
2. Identify the text
3. Organize the text
4. Extract features
5. Analyze
6. Draw a conclusion/reach an insight

Step 1: Define your problem

Tips:

- Be precise
- Avoid a "scope creep"
- Iterate and try new methods and/or subjectivity lexicons to ensure some consistency

Step 2: ID your text

Tips:

- Find appropriate sources (e.g. searching Wikipedia for stock prices may make less sense than examining a stock forum)
- Follow the terms of service for a site, be mindful of web scraping
- Text sources affect the language used...become familiar with the source's tone and nuances

Let's practice!

SENTIMENT ANALYSIS IN R

Step 3: Organize (& clean) the text

SENTIMENT ANALYSIS IN R

Ted Kwartler
Data Dude

Get to it!

Initial goal: Use the `polarity()` function to define subsections of the text for examination.

```
pos_comments <- subset(bos_reviews$comments,
                       bos_reviews$polarity > 0)
neg_comments <- subset(bos_reviews$comments,
                       bos_reviews$polarity < 0)

pos_terms <- paste(pos_comments, collapse = " ")
neg_terms <- paste(neg_comments, collapse = " ")
```

More organization

Goal: Use the tidy rental reviews to create the tidy formatted polarity scoring.

```
library(tidytext)
library(dplyr)

tidy_reviews <- bos_reviews %>%
    unnest_tokens(word, comments)

tidy_reviews <- tidy_reviews %>%
    group_by(id) %>%
    mutate(original_word_order = seq_along(word))
```

Tidy text polarity scoring

Recall the "bing" lexicon in `sentiments` has words categorized either as positive or negative.

```
library(tidytext)
library(tidyr)
library(dplyr)

bing <- sentiments %>%
    filter(lexicon == "bing")

pos_neg <- tidy_reviews %>%
    inner_join(bing) %>%
    count(sentiment) %>%
    pivot_wider(names_from = sentiment, values_from = n, values_fill = 0) %>%
    mutate(polarity = positive - negative)
```

Let's practice!

SENTIMENT ANALYSIS IN R

Revising the comparison cloud

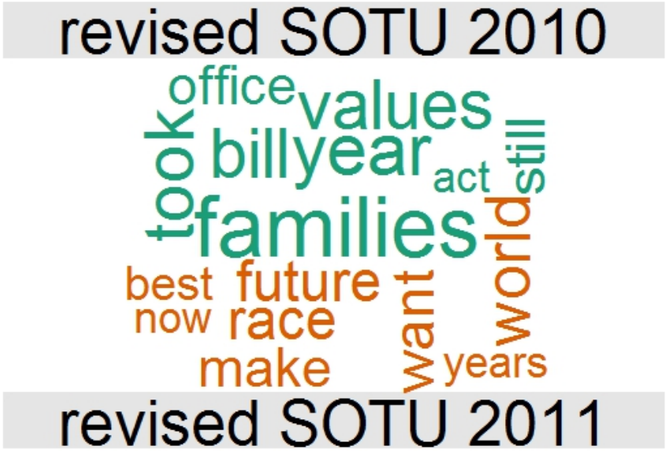
SENTIMENT ANALYSIS IN R

Ted Kwartler
Data Dude

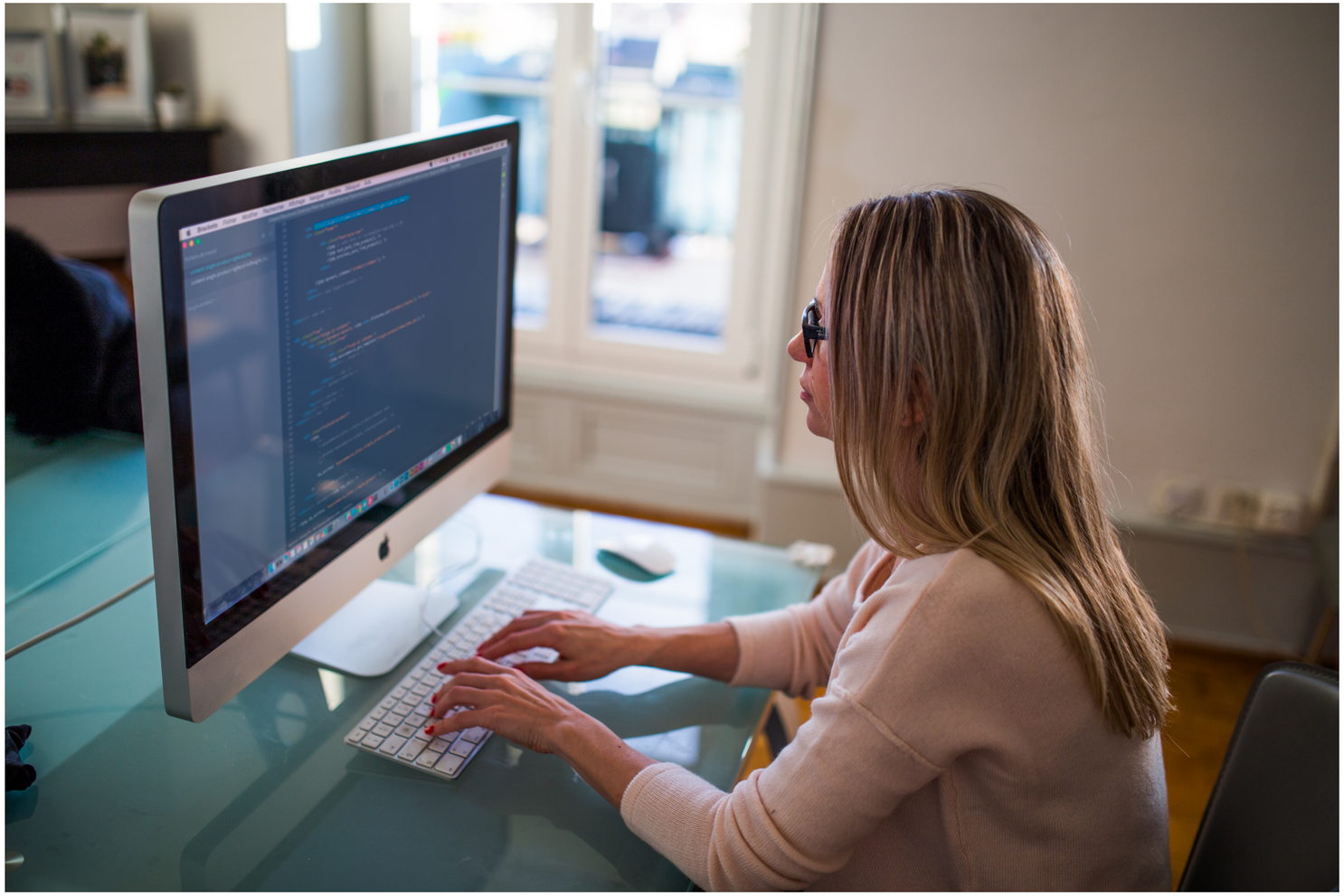
Author effort

Comparisons

Revising the comparison cloud



Always more analysis can be done!



Let's practice!

SENTIMENT ANALYSIS IN R

Step 6: Reach a conclusion

SENTIMENT ANALYSIS IN R

Ted Kwartler
Data Dude

Find the light bulb moments!

Let's practice!

SENTIMENT ANALYSIS IN R

Your turn!

SENTIMENT ANALYSIS IN R

Ted Kwartler
Data Dude

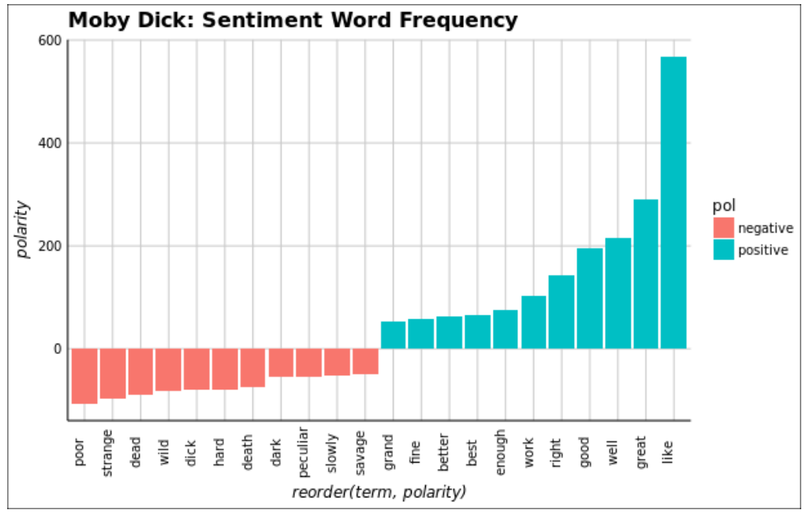
datacamp

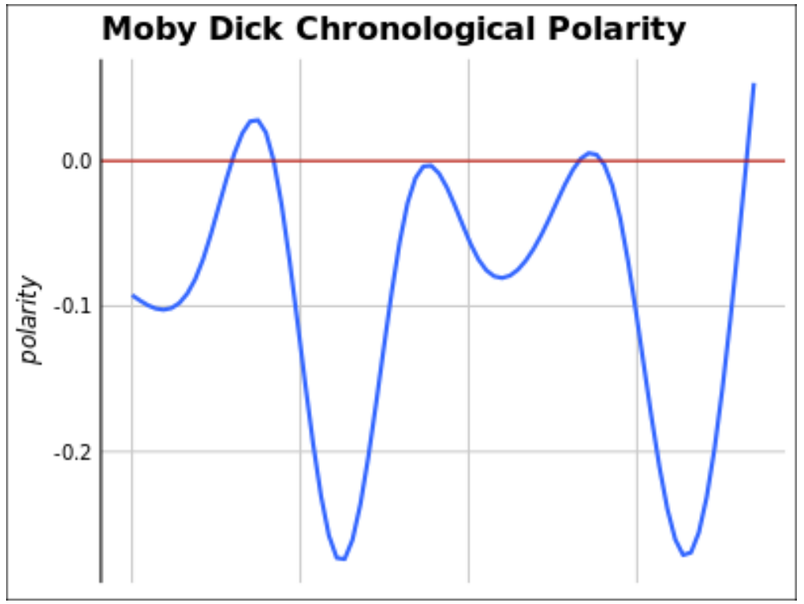
Congratulations!!

In this course you learned:

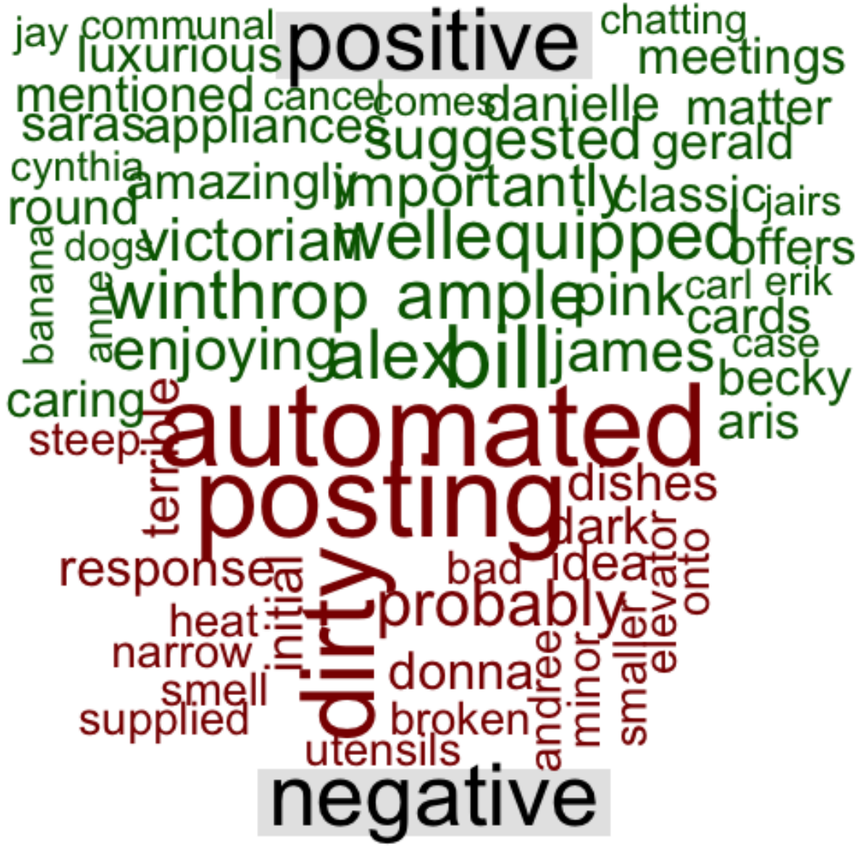
- `qdap`'s `polarity()` function
- `tidytext` data formats and `tidy` data functions
- `inner_join` with subjectivity lexicons

Congratulations!!





Congratulations!!



Good luck!

SENTIMENT ANALYSIS IN R