SENTIMENT ANALYSIS IN R: THE TIDY WAY

Welcome!

Julia Silge
Data Scientist at Stack Overflow

In this course, you will...

- learn how to implement sentiment analysis using tidy data principles
- explore sentiment lexicons
- apply these skills to real-world case studies

Case studies

- Geocoded Twitter data
- six of Shakespeare's plays
- text spoken on TV news programs
- lyrics from pop songs over the last 50 years

Sentiment Lexicons

```
> library(tidytext)

> get_sentiments("bing")

# A tibble: 6,788 x 2
         word sentiment
        <chr>     <chr>
 1     2-faced  negative
 2     2-faces  negative
 3         a+  positive
 4    abnormal  negative
 5     abolish  negative
 6  abominable  negative
 7  abominably  negative
 8   abominate  negative
 9 abomination  negative
10       abort  negative
# ... with 6,778 more rows
```

Sentiment Lexicons

```
> get_sentiments("afinn")

# A tibble: 2,476 x 2
         word score
        <chr> <int>
 1    abandon    -2
 2  abandoned    -2
 3   abandons    -2
 4   abducted    -2
 5  abduction    -2
 6 abductions    -2
 7      abhor    -3
 8   abhorred    -3
 9  abhorrent    -3
10     abhors    -3
# ... with 2,466 more rows
```

Sentiment Lexicons

```
> get_sentiments("nrc")

# A tibble: 13,901 x 2
         word sentiment
        <chr>     <chr>
 1      abacus     trust
 2     abandon      fear
 3     abandon  negative
 4     abandon   sadness
 5   abandoned     anger
 6   abandoned      fear
 7   abandoned  negative
 8   abandoned   sadness
 9 abandonment     anger
10 abandonment      fear
# ... with 13,891 more rows
```

SENTIMENT ANALYSIS IN R: THE TIDY WAY

Let's get started!

SENTIMENT ANALYSIS IN R: THE TIDY WAY

Sentiment analysis using an inner join

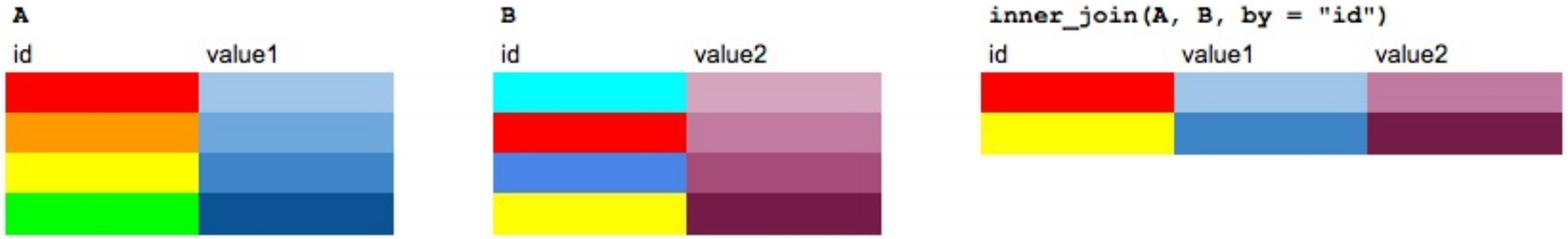
Julia Silge
Data Scientist at Stack Overflow

Geocoded Tweets

The geocoded_tweets dataset contains three columns:

- state, a state in the United States
- word, a word used in tweets posted on Twitter
- freq, the average frequency of that word in that state (per billion words)

Inner Join



Inner Join

```
> text

# A tibble: 7 x 1
      word
     <chr>
1      wow
2     what
3       an
4  amazing
5 beautiful
6 wonderful
7      day
```

```
> lexicon

# A tibble: 4 x 1
      word
     <chr>
1  amazing
2 wonderful
3      sad
4  terrible
```

Inner Join

```
> library(dplyr)

> text %>%
    inner_join(lexicon)
Joining, by = "word"

# A tibble: 2 x 1
      word
     <chr>
1   amazing
2 wonderful
```

SENTIMENT ANALYSIS IN R: THE TIDY WAY

Let's practice!

SENTIMENT ANALYSIS IN R: THE TIDY WAY

Analyzing sentiment analysis results

Julia Silge
Data Scientist at Stack Overflow

Getting to know dplyr verbs

Want to find only certain kinds of results? Use filter!

```
tweets_nrc %>%
  filter(sentiment == "positive")
```

Getting to know dplyr verbs

Want to find only certain kinds of results? Use filter!

```
tweets_nrc %>%
  filter(sentiment == "positive")
```

Need to do something for groups defined by your variables? Use group_by!

```
tweets_nrc %>%
  filter(sentiment == "positive") %>%
  group_by(word)
```

Getting to know dplyr verbs

Need to calculate something for defined groups? Use summarize!

```
tweets_nrc %>%
  filter(sentiment == "sadness") %>%
  group_by(word) %>%
  summarize(freq = mean(freq))
```

Getting to know dplyr verbs

Need to calculate something for defined groups? Use summarize!

```
tweets_nrc %>%
  filter(sentiment == "sadness") %>%
  group_by(word) %>%
  summarize(freq = mean(freq))
```

Want to arrange your results in some order? Use arrange!

```
tweets_nrc %>%
  filter(sentiment == "sadness") %>%
  group_by(word) %>%
  summarize(freq = mean(freq)) %>%
  arrange(desc(freq))
```

Common patterns

```
your_df %>%
    group_by(your_variable) %>%
    {DO_SOMETHING_HERE} %>%
    ungroup
```

SENTIMENT ANALYSIS IN R: THE TIDY WAY

Let's practice!

SENTIMENT ANALYSIS IN R: THE TIDY WAY

Differences by state

Julia Silge
Data Scientist at Stack Overflow

Exploring states

Examing one state

```
tweets_nrc %>%
  filter(state == "texas",
         sentiment == "positive")
```

Exploring states

Examing one state

```
tweets_nrc %>%
  filter(state == "texas",
         sentiment == "positive")
```

Calculating a quantity for all states

```
tweets_nrc %>%
  group_by(state)
```

spread() converts long data

id	group	value
1	A	5.5
2	A	2.2
3	A	9.9
1	B	6.6
2	B	7.7
3	B	1.1
1	C	8.8
2	C	3.3
3	C	4.4

spread() converts long data to wide data

id	group A value	group B value	group C value
1	5.5	6.6	8.8
2	2.2	7.7	3.3
3	9.9	1.1	4.4

Using spread()

```
tweets_bing %>%
  group_by(state, sentiment) %>%
  summarize(freq = mean(freq)) %>%
  spread(sentiment, freq) %>%
  ungroup()
```

SENTIMENT ANALYSIS IN R: THE TIDY WAY

Let's go!