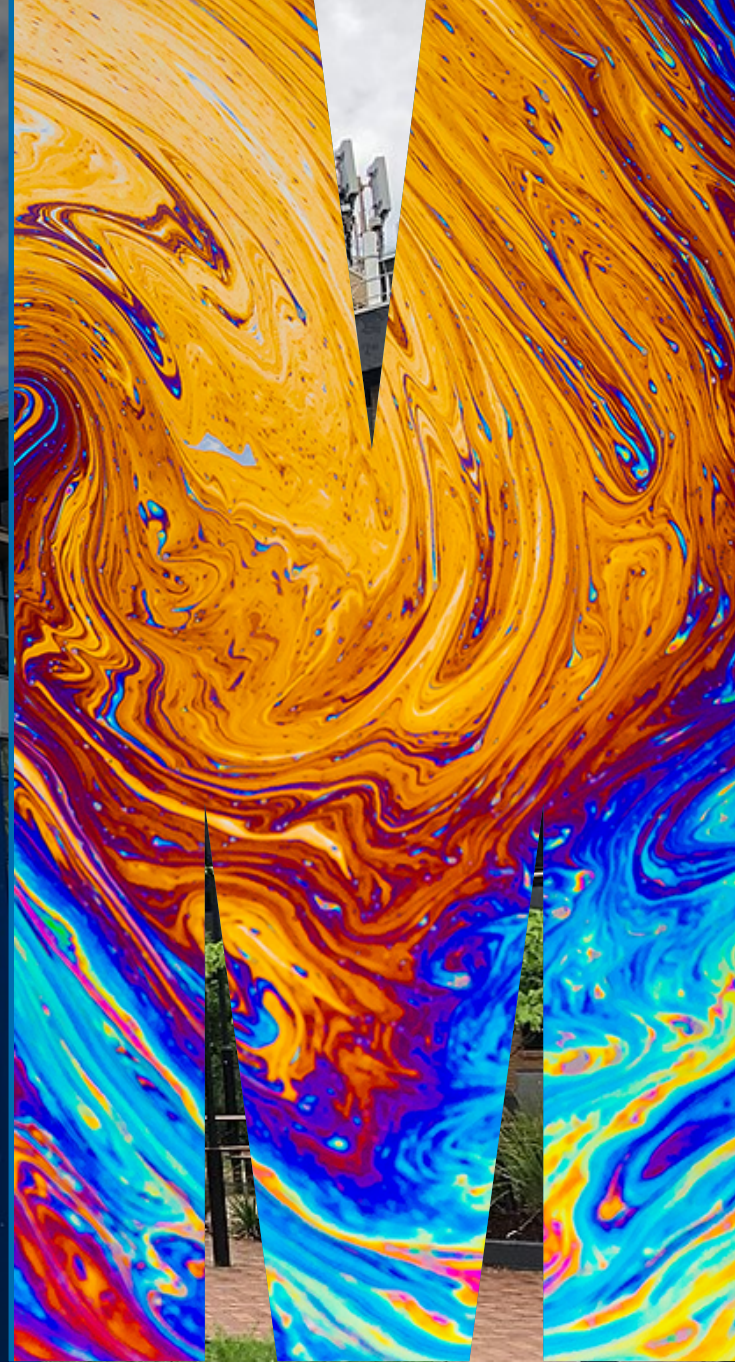
ETC5510: Introduction to Data Analysis

Week 8, part B

Text analysis Part 2

Lecturer: *Nicholas Tierney & Stuart Lee*
Department of Econometrics and Business Statistics
ETC5510.Clayton-x@monash.edu
May 2020



Recap

- tidying up text
- stop_words - (I, am, be, the, this, what, we, myself)

Overview

- tidy text continued

Sentiment analysis

Sentiment analysis tags words or phrases with an emotion, and summarises these, often as the positive or negative state, over a body of text.

Sentiment analysis: examples

- Examining effect of emotional state in twitter posts
- Determining public reactions to government policy, or new product releases
- Trying to make money in the stock market by modeling social media posts on listed companies
- Evaluating product reviews on Amazon, restaurants on zomato, or travel options on TripAdvisor

Lexicons

The `tidytext` package has a lexicon of sentiments, based on four major sources: AFINN, bing, Loughran, nrc

emotion

What emotion do these words elicit in you?

- summer
- hot chips
- hug
- lose
- stolen
- smile

Different sources of sentiment

- The `nrc` lexicon categorizes words in a binary fashion ("yes"/"no") into categories of positive, negative, anger, anticipation, disgust, fear, joy, sadness, surprise, and trust.
- The `bing` lexicon categorizes words in a binary fashion into positive and negative categories.
- The `AFINN` lexicon assigns words with a score that runs between -5 and 5, with negative scores indicating negative sentiment and positive scores indicating positive sentiment.

Different sources of sentiment

```
get_sentiments("afinn")
## # A tibble: 2,477 x 2
##    word       value
##    <chr>      <dbl>
##  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 2,467 more rows
```

Sentiment analysis

- Once you have a bag of words, you need to join the sentiments dictionary to the words data.
- Particularly the lexicon `nrc` has multiple tags per word, so you may need to use an "inner_join".
- `inner_join()` returns all rows from x where there are matching values in y, and all columns from x and y.
- If there are multiple matches between x and y, all combination of the matches are returned.

Exploring sentiment in Jane Austen

`janeaustenr` package contains the full texts, ready for analysis for for Jane Austen's 6 completed novels:

1. "Sense and Sensibility"
2. "Pride and Prejudice"
3. "Mansfield Park"
4. "Emma"
5. "Northanger Abbey"
6. "Persuasion"

Exploring sentiment in Jane Austen

```
library(janeaustenr)
library(stringr)

tidy_books <- austen_books() %>%
  group_by(book) %>%
  mutate(linenumber = row_number(),
         chapter = cumsum(str_detect(text,
                                     regex("^chapter [\\divxlc]",
                                           ignore_case = TRUE)))) %>%
  ungroup() %>%
  unnest_tokens(word, text)
```

Exploring sentiment in Jane Austen

```
tidy_books
## # A tibble: 725,055 x 4
##    book                linenumber chapter word
##    <fct>                    <int>   <int> <chr>
##  1 Sense & Sensibility          1       0 sense
##  2 Sense & Sensibility          1       0 and
##  3 Sense & Sensibility          1       0 sensibility
##  4 Sense & Sensibility          3       0 by
##  5 Sense & Sensibility          3       0 jane
##  6 Sense & Sensibility          3       0 austen
##  7 Sense & Sensibility          5       0 1811
##  8 Sense & Sensibility         10       1 chapter
##  9 Sense & Sensibility         10       1 1
## 10 Sense & Sensibility         13       1 the
## # … with 725,045 more rows
```

Count joyful words in "Emma"

```
nrc_joy <- get_sentiments("nrc") %>%
  filter(sentiment == "joy")

tidy_books %>%
  filter(book == "Emma") %>%
  inner_join(nrc_joy) %>%
  count(word, sort = TRUE)
## # A tibble: 303 x 2
##    word        n
##    <chr>   <int>
##  1 good      359
##  2 young     192
##  3 friend    166
##  4 hope      143
##  5 happy     125
##  6 love      117
##  7 deal       92
##  8 found      92
##  9 present    89
## 10 kind       82
```

Count joyful words in "Emma"

"Good" is the most common joyful word, followed by "young", "friend", "hope".

All make sense until you see "found".

Is "found" a joyful word?

Comparing lexicons

- All of the lexicons have a measure of positive or negative.
- We can tag the words in Emma by each lexicon, and see if they agree.

```
nrc_pn <- get_sentiments("nrc") %>%
  filter(sentiment %in% c("positive",
                          "negative"))

emma_nrc <- tidy_books %>%
  filter(book == "Emma") %>%
  inner_join(nrc_pn)

emma_bing <- tidy_books %>%
  filter(book == "Emma") %>%
  inner_join(get_sentiments("bing"))

emma_afinn <- tidy_books %>%
  filter(book == "Emma") %>%
  inner_join(get_sentiments("afinn"))
```

Comparing lexicons

```
emma_nrc
```

```
## # A tibble: 13,944 x 5
##    book  linenumber chapter word       sentiment
##    <fct>      <int>   <int> <chr>      <chr>
##  1 Emma          15       1 clever     positive
##  2 Emma          16       1 happy      positive
##  3 Emma          16       1 blessings  positive
##  4 Emma          17       1 existence  positive
##  5 Emma          18       1 distress   negative
##  6 Emma          21       1 marriage   positive
##  7 Emma          22       1 mistress   negative
##  8 Emma          22       1 mother     negative
##  9 Emma          22       1 mother     positive
## 10 Emma          23       1 indistinct negative
## # … with 13,934 more rows
```

Comparing lexicons

```
emma_afinn
```

```
## # A tibble: 10,901 x 5
##    book  linenumber chapter word         value
##    <fct>      <int>   <int> <chr>        <dbl>
##  1 Emma          15       1 clever           2
##  2 Emma          15       1 rich             2
##  3 Emma          15       1 comfortable      2
##  4 Emma          16       1 happy            3
##  5 Emma          16       1 best             3
##  6 Emma          18       1 distress        -2
##  7 Emma          20       1 affectionate     3
##  8 Emma          22       1 died            -3
##  9 Emma          24       1 excellent        3
## 10 Emma          25       1 fallen          -2
## # … with 10,891 more rows
```

Comparing lexicons

```
emma_nrc %>% count(sentiment) %>% mutate(n / sum(n))
## # A tibble: 2 x 3
##   sentiment     n `n/sum(n)`
##   <chr>     <int>      <dbl>
## 1 negative   4473      0.321
## 2 positive   9471      0.679

emma_bing %>% count(sentiment) %>% mutate(n / sum(n))
## # A tibble: 2 x 3
##   sentiment     n `n/sum(n)`
##   <chr>     <int>      <dbl>
## 1 negative   4809      0.402
## 2 positive   7157      0.598
```

Comparing lexicons

```
emma_afinn %>%
  mutate(sentiment = ifelse(value > 0,
                            "positive",
                            "negative")) %>%
  count(sentiment) %>%
  mutate(n / sum(n))
## # A tibble: 2 x 3
##   sentiment     n `n/sum(n)`
##   <chr>     <int>      <dbl>
## 1 negative   4429      0.406
## 2 positive   6472      0.594
```

Your Turn: Sentiment of Austen

- What are the most common "anger" words used in Emma?
- What are the most common "surprise" words used in Emma?
- Which book is the most positive? negative?
- Using your choice of lexicon (nrc, bing, or afinn) compute the proportion of positive words in each of Austen's books.

Lab exercise: The Simpsons

Data from the popular animated TV series, The Simpsons, has been made available on kaggle.

- `simpsons_script_lines.csv`: Contains the text spoken during each episode (including details about which character said it and where)
- `simpsons_characters.csv`: Contains character names and a character id

Lab exercise (bonus) Origin of Species

- Downloading books from gutenberg
- Using tf-idf to look at how editions of the Darwin's book have changed

background-image: url(images/bg1.jpg) background-size: cover
class: hide-slide-number split-70 count: false

That's it!

Lecturer: Nicholas Tierney & Stuart Lee

Department of Ec metrics and Business Stati cs

ETC5510.Clayt monash.edu