

Data Analytics

CS301

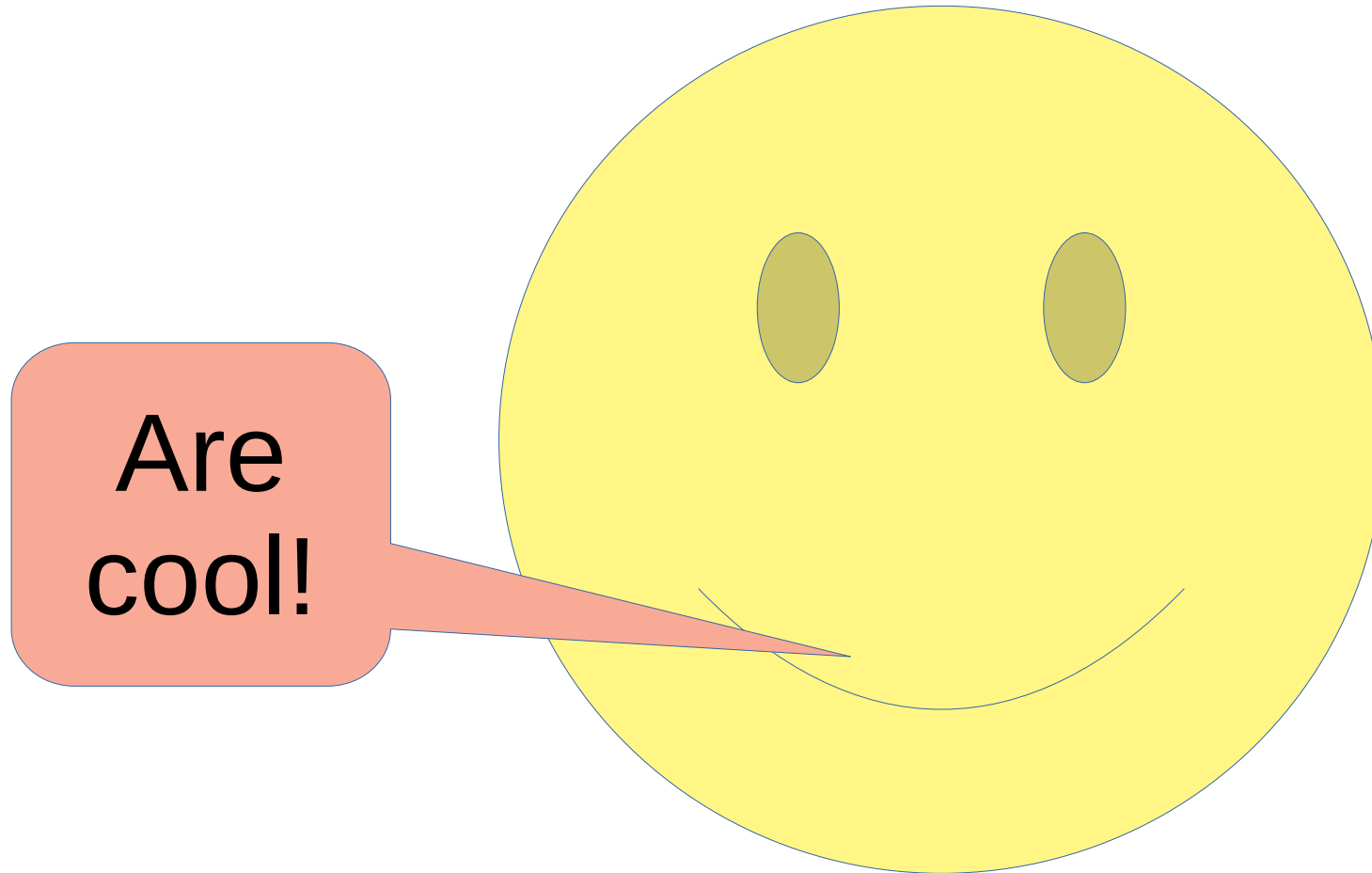
Text Analysis:

Sentiment Determination

Week 12: 11 November 2021
Oliver BONHAM-CARTER



The following tools ...





Vaccine Hesitancy by County



Vaccine hesitancy by county

Sep 17, 2021 - Sep 23, 2021

This map highlights areas of the US that would benefit most from increased vaccination acceptance. This view shows, by county, the % of survey respondents who answered "Yes, probably" or "No, probably not" when asked "If a vaccine to prevent COVID-19 were offered to you today, would you choose to get vaccinated?"

Data source: The Delphi Group at Carnegie Mellon University U.S. COVID-19 Trends and Impact Survey, in partnership with Facebook.

Detail

County

Zip code

Hesitancy ⓘ

Somewhat

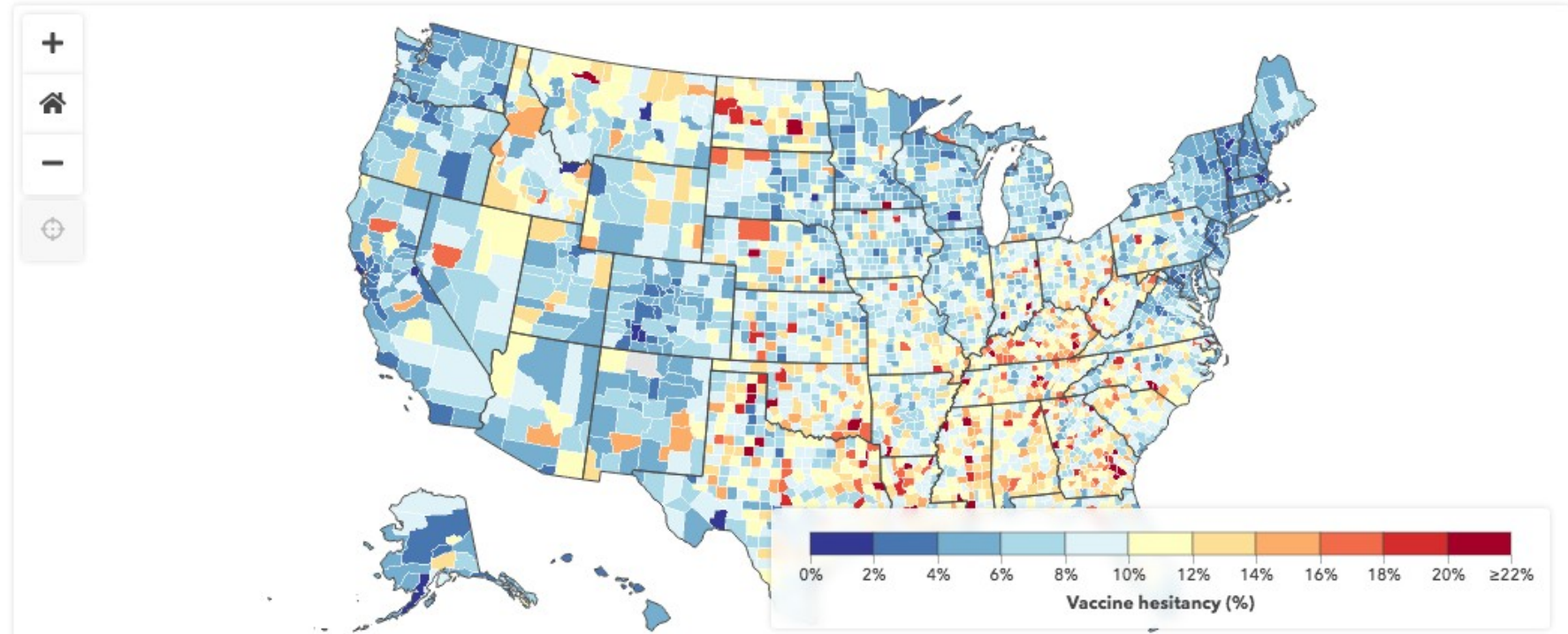
All

Show change ⓘ



Highlight

Select county(ies) or click on map ▼



<https://vaccine-hesitancy.healthdata.org/>

Online Tool: The Opportunity Atlas



The Opportunity Atlas

Which neighborhoods in America offer children the best chance to rise out of poverty?

The Opportunity Atlas answers this question using anonymous data following 20 million Americans from childhood to their mid-30s.

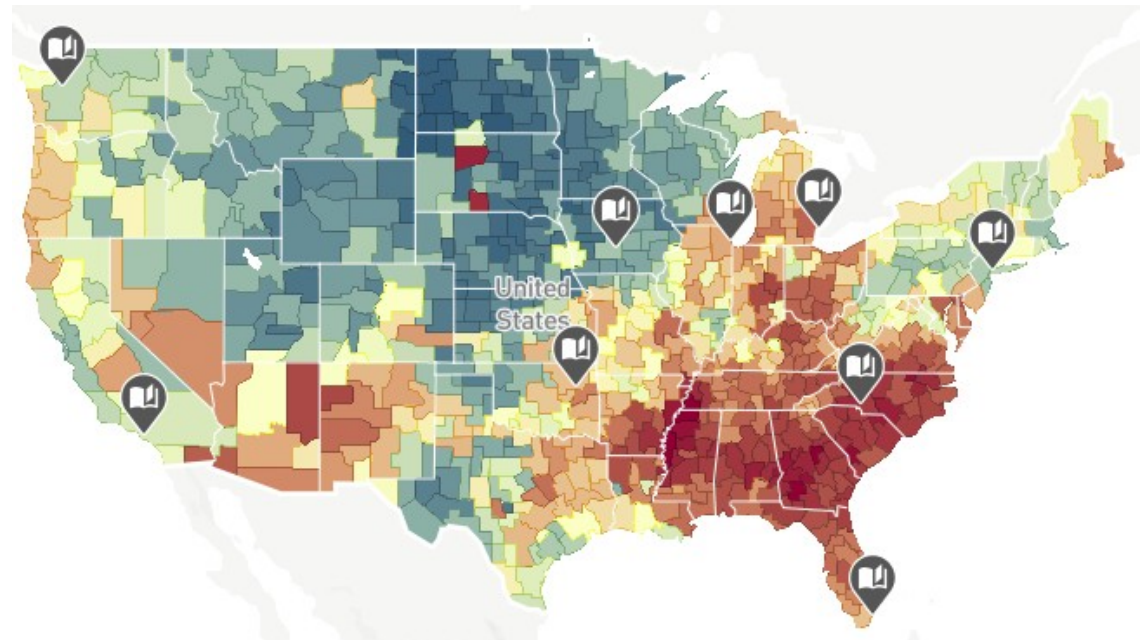
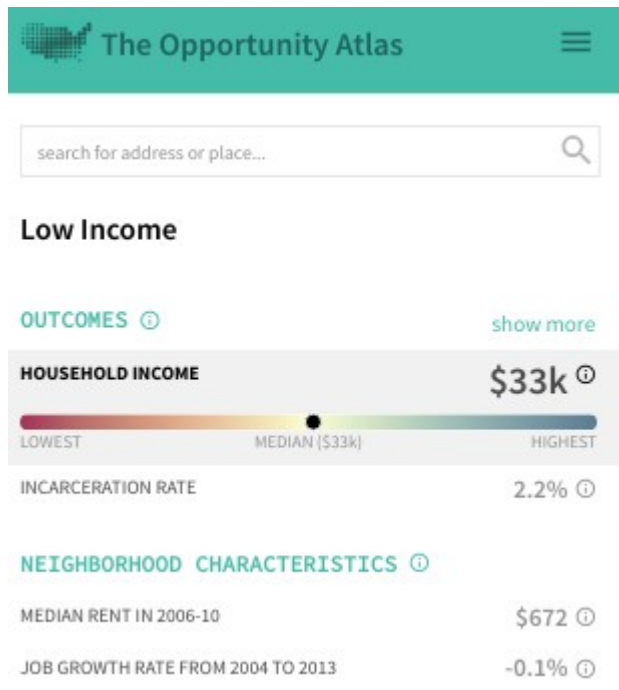
Now you can trace the roots of today's affluence and poverty back to the neighborhoods where people **grew up**.

See where and for whom opportunity has been missing, and develop local solutions to help more children rise out of poverty.

BEGIN EXPLORING

<https://www.opportunityatlas.org/>

Online Tool: The Opportunity Atlas



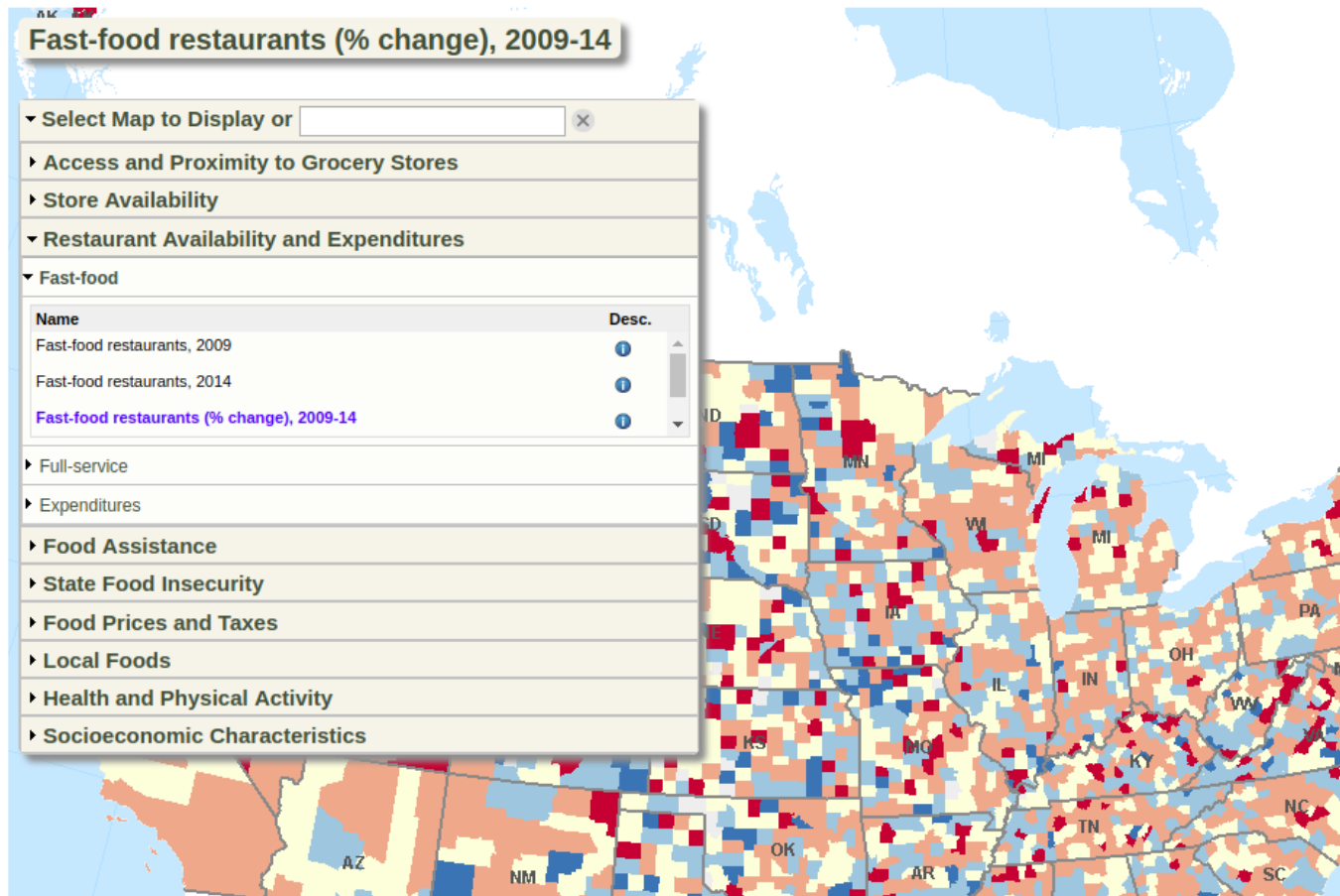
Determine the statistical amount of opportunity for careers, educational development and similar by a map.

<https://www.opportunityatlas.org/>



Online Tool: The US Dept of Agriculture

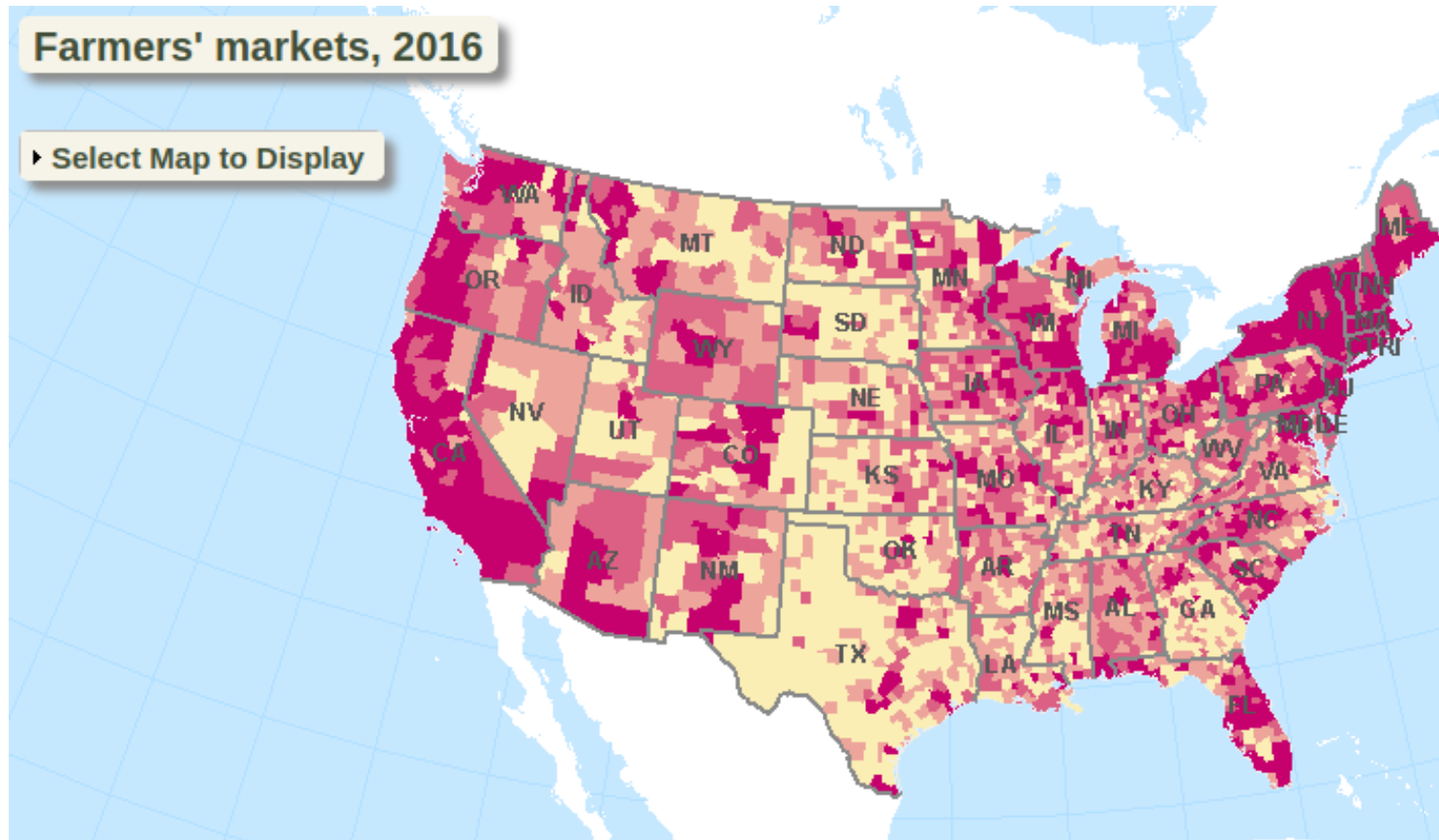
Go to the Atlas



<https://www.ers.usda.gov/data-products/food-environment-atlas/go-to-the-atlas/>



Online Tool: The US Dept of Agriculture



Mapping the number of Farmer's Markets available in 2016

<https://www.ers.usda.gov/data-products/food-environment-atlas/go-to-the-atlas/>



ALLEGHENY
COLLEGE

Online Tool: The Institute for Health Metrics and Evaluation



IHME

Measuring what matters

Home

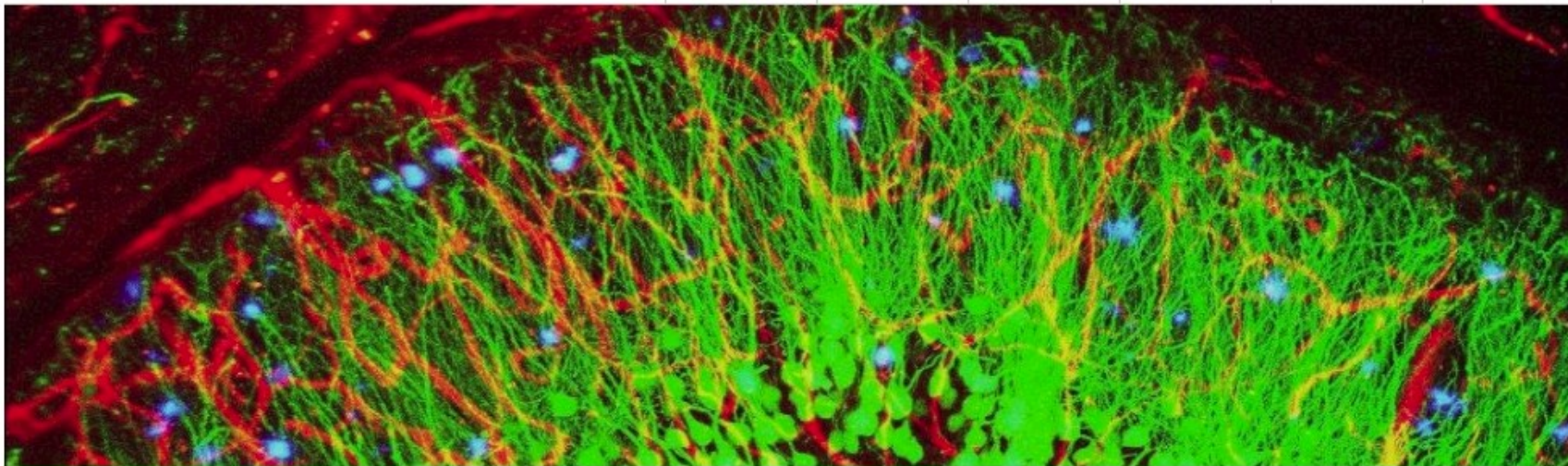
Results

News &
Events

Projects

Get
Involved

About



New neurology studies a 'wakeup call' for global health

Photo by Alvin Gogineni, Genentech.



<http://www.healthdata.org/>

<https://vizhub.healthdata.org/epi/>



Online Tool: The Institute for Health Metrics and Evaluation



IHME

Measuring what matters

Home

Results

News &
Events

Projects

Get
Involved

About

Data Visualizations

RESULTS

GBD Results Tool

Data Visualizations

Country Profiles

US Health

Policy Reports

Research Articles

Infographics

Topics

Data & Tools

Topics

- Any -

Date published

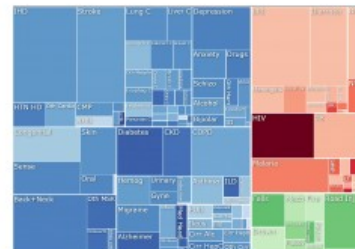
-Year



► Advanced

Apply

Reset



NOVEMBER 8, 2018

GBD Compare

Data Visualization

[Learn more](#)

Analyze updated data about the world's health levels and trends from 1990 to 2017 in this interactive tool using estimates from the Global Burden of Disease (GBD) study. Use treemaps, maps, arrow diagrams, and other charts to compare causes and risks within a country (now at the US state-level), compare countries with regions or the world, and explore patterns and trends by country, age, and gender. Drill from a global view into specific details.

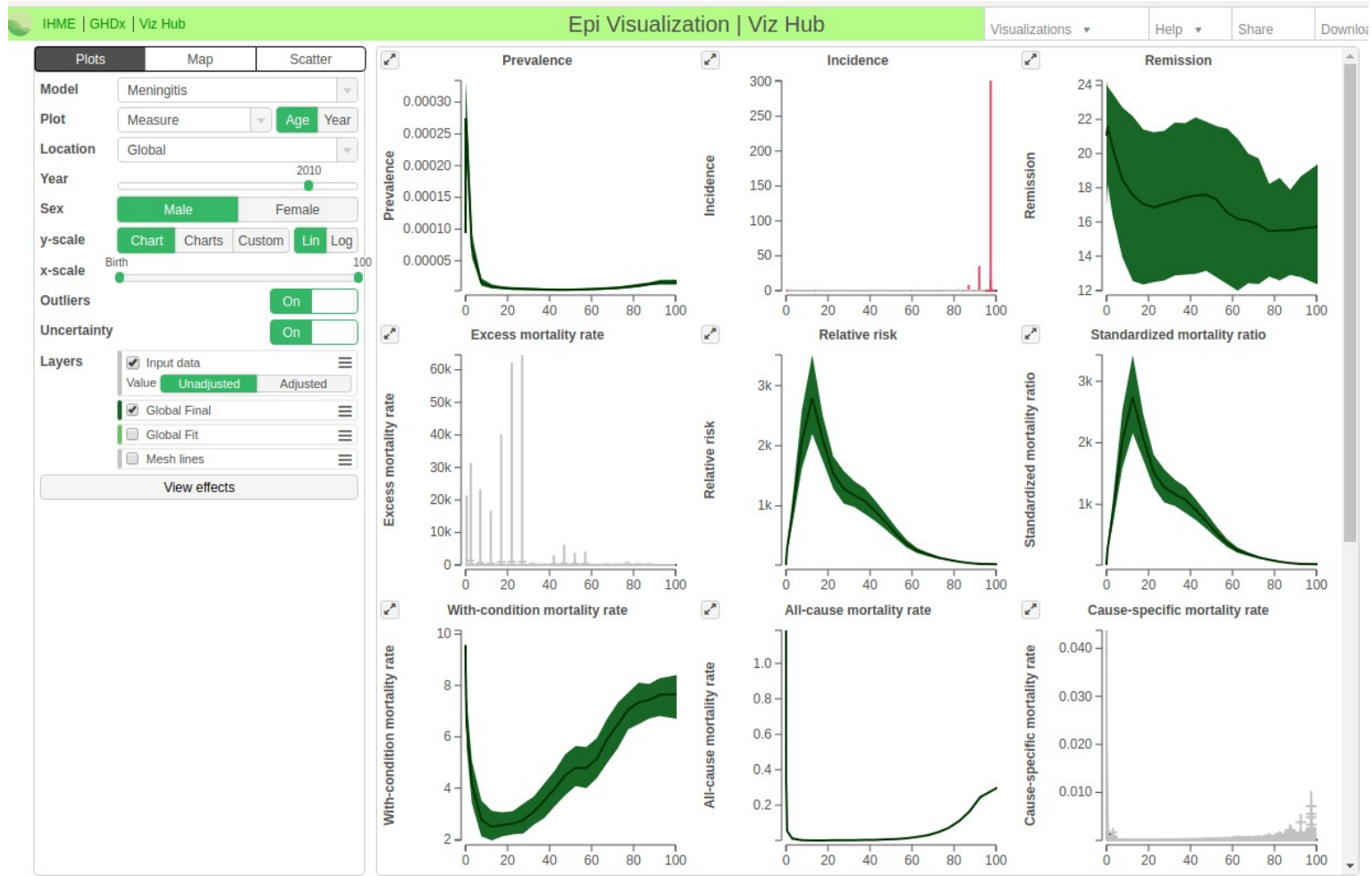
Visualize data on seemingly any topic of health

<http://www.healthdata.org/>

<https://vizhub.healthdata.org/epi/>



Online Tool: The Institute for Health Metrics and Evaluation

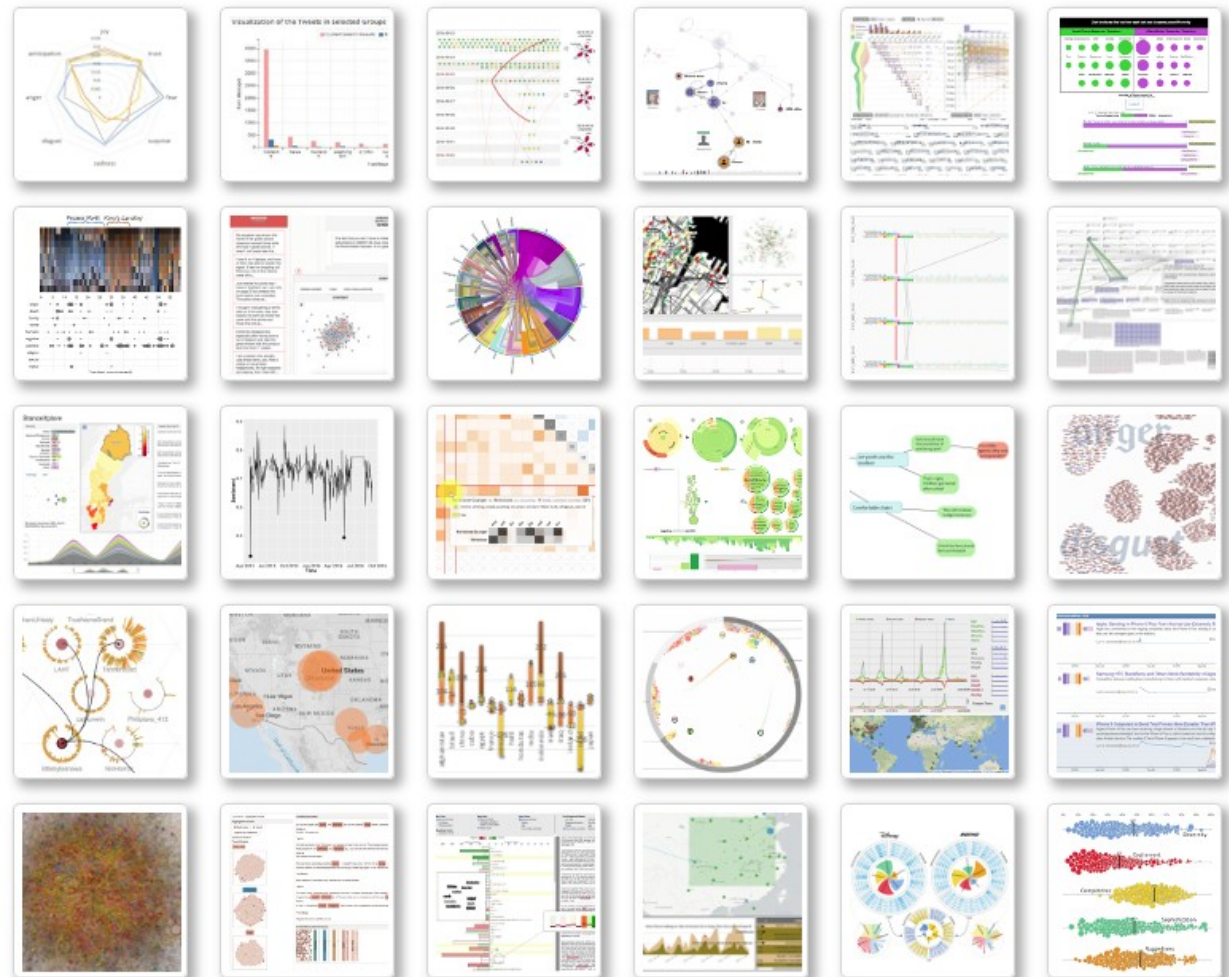


<https://vizhub.healthdata.org/epi/>

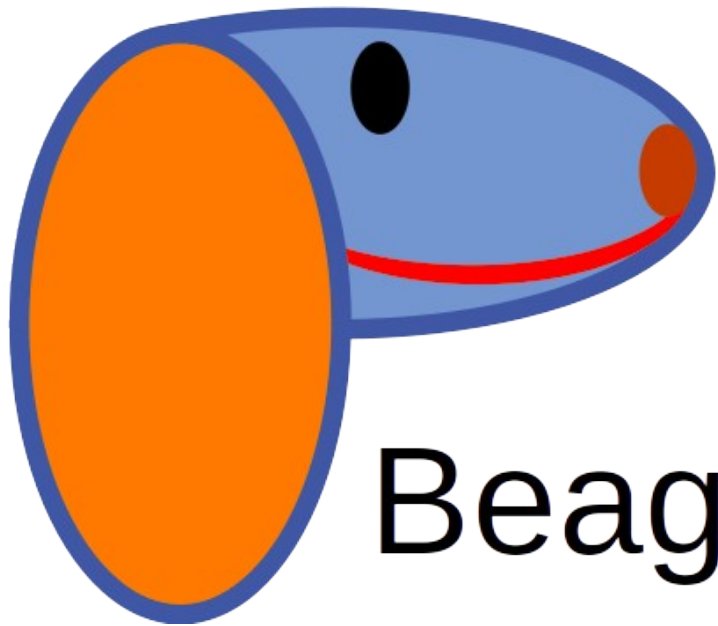
Visualizing Schemes are still being developed

- To find out about new work in visualizing analytics, check out the **SentimentVis Browser** at <http://sentimentvis.lnu.se/>

A Visual Survey
of Sentiment
Visualization
Techniques:
Have a look at
all the different
ways to determine
sentiment in text!



Find Relationships in the Literature



BeagleTM



A text mining tool for developing visual and interactive relationship networks from PubMed article information.

GitHub:

<https://github.com/developmentAC/beagleTM>

Slides:

https://www.cs.allegheny.edu/sites/obonhamcarter/slides/beagleTM_ficc.pdf



Text Analysis:

Sentiment of Content

- The determination of a text's "message" or "mood" based on the actual individual *words*.
- How good, how bad is the writer feeling about some topic?
- Is a body of text describing some idea where many of the words are emotionally charged with some type of feeling?
- Sentiment analysis is able to determine what the general feeling is behind some written work.



Tweet Text Analysis



sentiment viz

Tweet Sentiment Visualization

pleasant
high confidence



unpleasant
low confidence

https://www.csc2.ncsu.edu/faculty/healey/tweet_viz/tweet_app/

What Is This Tool?

- User-entered keywords are parsed in the tweets of the day.
- Tweets are presented using several different visualization techniques. Each technique is designed to highlight different aspects of the tweets and their sentiment.
- The sentiment tab visualizes where tweets lie in an emotional scatterplot with pleasure and arousal on its horizontal and vertical axes.
- The spatial distribution of the tweets summarizes their overall sentiment.
- The number of queries per minute is limited...



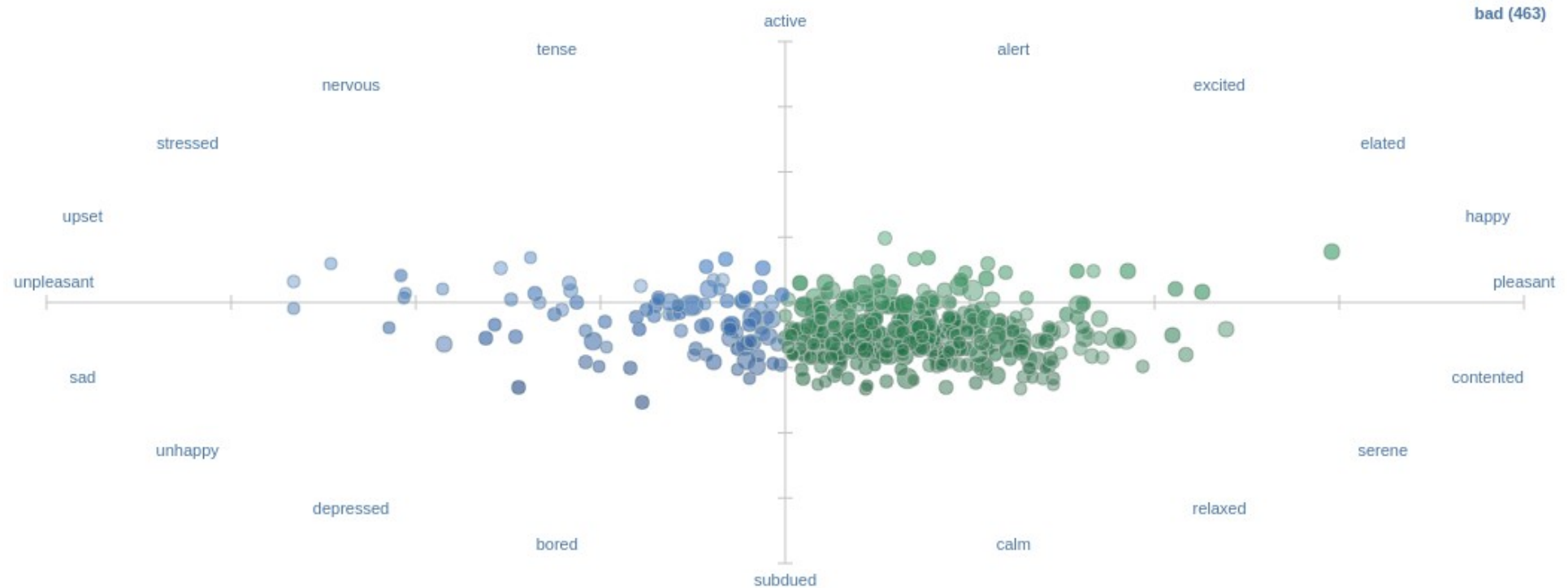
sentiment viz

Tweet Sentiment Visualization

https://www.csc2.ncsu.edu/faculty/healey/tweet_viz/tweet_app/

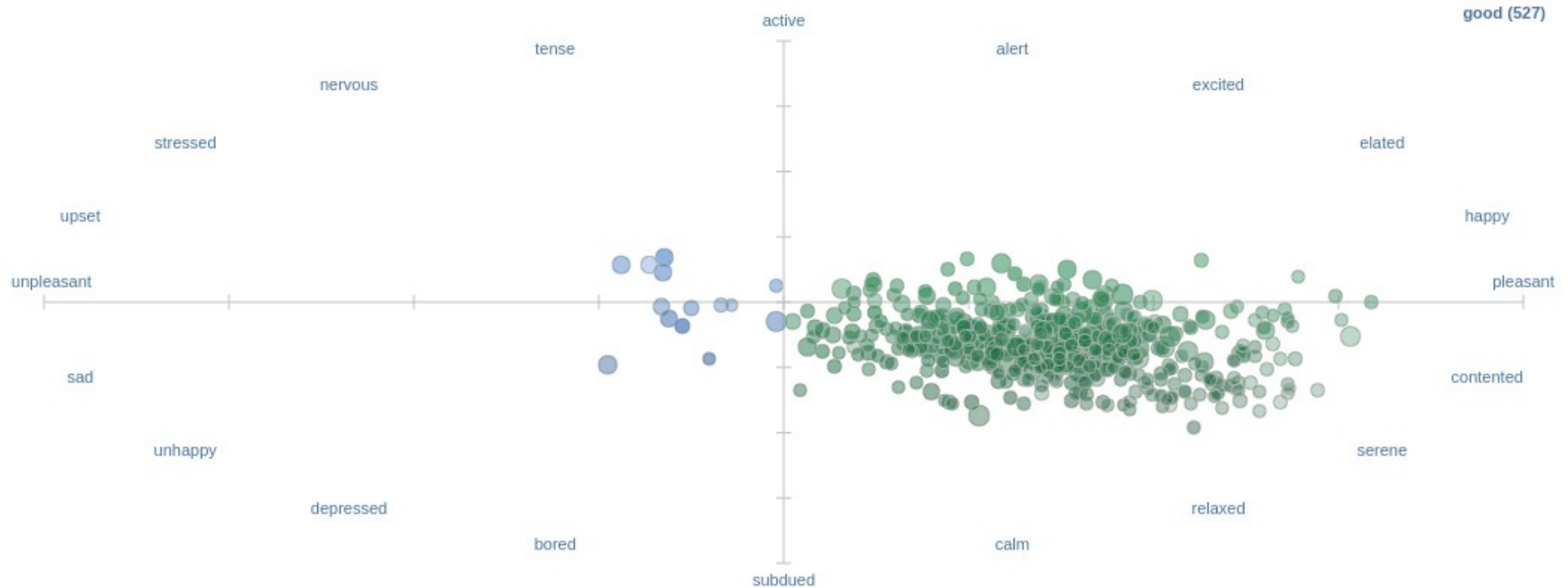


The word, “Bad”





The word, “Good”



Click around on the web site to discover new ways of viewing data.

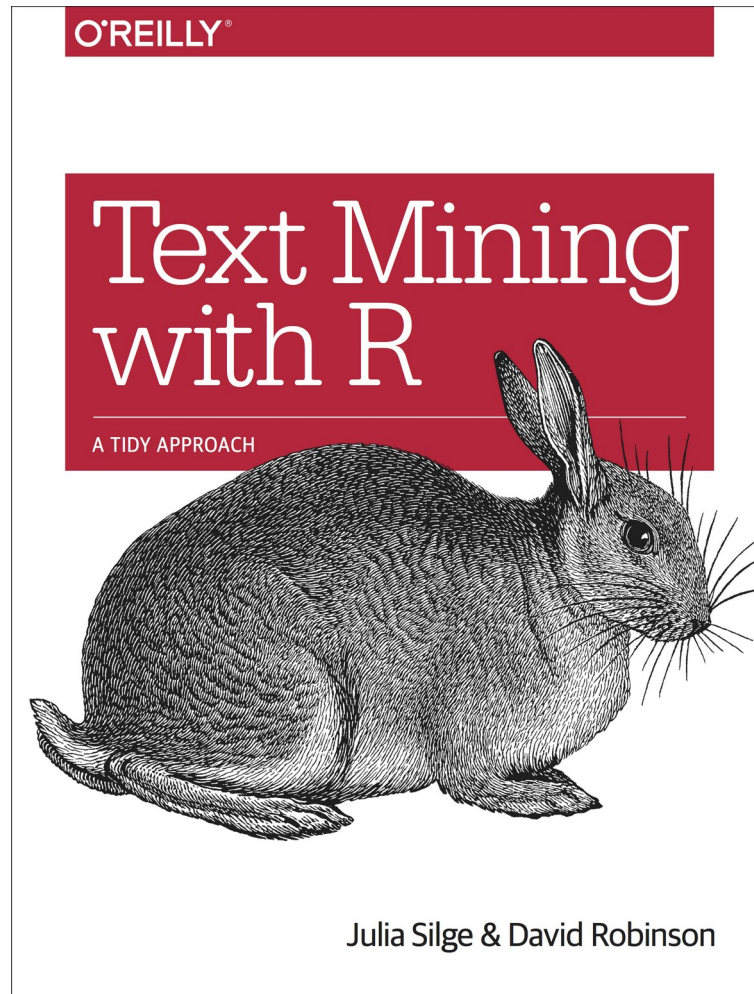


Another Type of Analysis...





Please Follow the Book



This slide material
below has been
taken from Silge *et al.*

Chapter: 2
*Sentiment analysis with
tidy data*

[https://www.tidytextmining.com/
sentiment.html](https://www.tidytextmining.com/sentiment.html)



Packages and Libraries

```
# install.packages("janeaustenr")
```

```
# install.packages("stringr")
```

```
rm(list = ls())
```

```
library(janeaustenr)
```

```
library(dplyr)
```

```
library(stringr)
```

```
library(tidyverse)
```

All following code is provided in:
sandbox/textAnalysis.r



Data: Jane Austen's Text

- Jane Austen's 6 completed, published novels from the *janeaustenr* package.
 - Sense & Sensibility
 - Pride & Prejudice
 - Mansfield Park
 - Emma
 - Northanger Abbey
 - Persuasion

Research Question

- Jane Austen's written work:

How many *Bad (pessimistic)* words did she use?

How many *Good (optimistic)* words did she use?





The *Sentiments* dataset

```
#install.packages("tidytext")  
library(tidytext)  
sentiments
```

```
## # A tibble: 27,314 × 4  
##           word sentiment  lexicon score  
##           <chr>      <chr>    <chr> <int>  
## 1      abacus      trust     nrc    NA  
## 2    abandon      fear     nrc    NA  
## 3    abandon negative     nrc    NA  
## 4    abandon  sadness     nrc    NA  
## 5  abandoned      anger     nrc    NA  
## 6  abandoned      fear     nrc    NA  
## 7  abandoned negative     nrc    NA
```




Three general-purpose lexicons

- **AFINN** from Finn Årup Nielsen,
 - assigns words with a score that runs between -5 and 5, with negative scores indicating negative sentiment and positive scores indicating positive sentiment
- **bing** from Bing Liu and collaborators,
 - categorizes words in a binary fashion into positive and negative categories
- **nrc** from Saif Mohammad and Peter Turney
 - categorizes words in a binary fashion (“yes”/“no”) into categories of positive, negative, anger, anticipation, disgust, fear, joy, sadness, surprise, and trust.
- Used to determine the general mood of words.
- Lexicons are based on unigrams, (i.e., single words).
- Words are assigned scores for positive/negative sentiment,
- Emotions: joy, anger, sadness and etc.



Sentiments: **afinn**

```
install.packages("textdata")  
get_sentiments("afinn")
```

You might need
to install another
package.

```
> 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
```

Returns
a score
for each word
[-5, 5]
(Bad to Good)



Sentiments: **nrc**

```
install.packages("textdata")  
get_sentiments("nrc")
```

```
> 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
```

You might need
to install another
package.

Returns
a *synonym*
for each word



Sentiments: **bing**

```
install.packages("textdata")  
get_sentiments("bing")
```

```
> 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
```

You might need
to install another
package.

Returns
“Positive”
or
“Negative”
for words



Wrangling Book Data

```
original_books <- austen_books() %>%  
  group_by(book) %>%  
  mutate(linenumber = row_number(),  
         chapter = cumsum(str_detect(text, regex("^chapter [\\  
divxlc]", ignore_case = TRUE)))) %>%  
  ungroup()  
  
# words from all novels  
View(original_books)
```




Chapter Words

- The words in the order that they appear in the text.
- Note the first line is the title of the book.

```
## # A tibble: 73,422 x 4
##   text                book                linenumber chapter
##   <chr>              <fctr>                <int>    <int>
## 1 SENSE AND SENSIBILITY Sense & Sensibility      1        0
## 2 ""                Sense & Sensibility      2        0
## 3 by Jane Austen    Sense & Sensibility      3        0
## 4 ""                Sense & Sensibility      4        0
## 5 (1811)            Sense & Sensibility      5        0
## 6 ""                Sense & Sensibility      6        0
## 7 ""                Sense & Sensibility      7        0
## 8 ""                Sense & Sensibility      8        0
## 9 ""                Sense & Sensibility      9        0
## 10 CHAPTER 1        Sense & Sensibility     10        1
## # ... with 73,412 more rows
```



Unnesting Book Words

We need the words in list (un-nested) to work with them.

```
tidy_books <- original_books %>%  
  unnest_tokens(word, text) #make a list of  
  words from the paragraphs
```

```
View(tidy_books)
```



Unnested Words

```
## # A tibble: 725,055 x 4
##   book                linenumber chapter word
##   <fctr>              <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
```

When words are in one-word-per-row format,
manipulation with tidy tools like *dplyr* is possible



Stop Words

- Remove *stop words*: words which do not add any distinguishing information to a body of text.
 - Contractions: hasn't, didn't won't
 - In-betweens: been, is, had, having

```
data("stop_words")
```

```
View(stop_words)
```

```
cleaned_books <- tidy_books %>% anti_join(stop_words)
```

```
# anti_join() returns all rows from x where there are not  
# matching values in y, keeping just columns from x.
```



Counting Common Words Across All Books

```
cleaned_books %>%  
  count(word, sort = TRUE)
```

```
## # A tibble: 13,914 x 2  
##   word      n  
##   <chr> <int>  
## 1 miss    1855  
## 2 time    1337  
## 3 fanny    862  
## 4 dear     822  
## 5 lady     817  
## 6 sir      806  
## 7 day      797  
## 8 emma     787  
## 9 sister   727  
## 10 house   699  
## # ... with 13,904 more rows
```



Joy in Emma

- We will consider the common words having scores indicating that they are of **Joy**, according to the *nrc* lexicon in the novel, Emma

```
#install.packages("textdata")  
library(textdata)  
# Note: enter '1', if prompted  
nrcjoy <- get_sentiments("nrc") %>%  
  filter(sentiment == "joy")  
tidy_books %>%  
  filter(book == "Emma") %>%  
  semi_join(nrcjoy) %>%  
  count(word, sort = TRUE)
```




Oh Joy ...

```
tidy_books %>%  
  filter(book == "Emma") %>%  
  semi_join(nrcjoy) %>%  
  count(word, sort = TRUE)
```

We find counts
of the *joy* words in
the novel, Emma

```
## # 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  
## # ... with 293 more rows
```



How Does Sentiment Change? (In each novel?)

```
library(tidyr)

bing <- get_sentiments("bing")

# move line by line of book, find difference in
# sentiments to "score" each line

janeaustensentiment <- tidy_books %>%
  inner_join(bing) %>%
  count(book, index = linenumbers %/% 80, sentiment)
  %>% spread(sentiment, n, fill = 0) %>%
  mutate(sentiment = positive - negative)
```



What Most Common Positive and Negative Words?

- Count the common positive words across the books.

```
bing_word_counts <- tidy_books %>%  
  inner_join(bing) %>%  
  count(word, sentiment, sort = TRUE) %>%  
  ungroup()
```

```
View(bing_word_counts)
```



Such Positivity ...

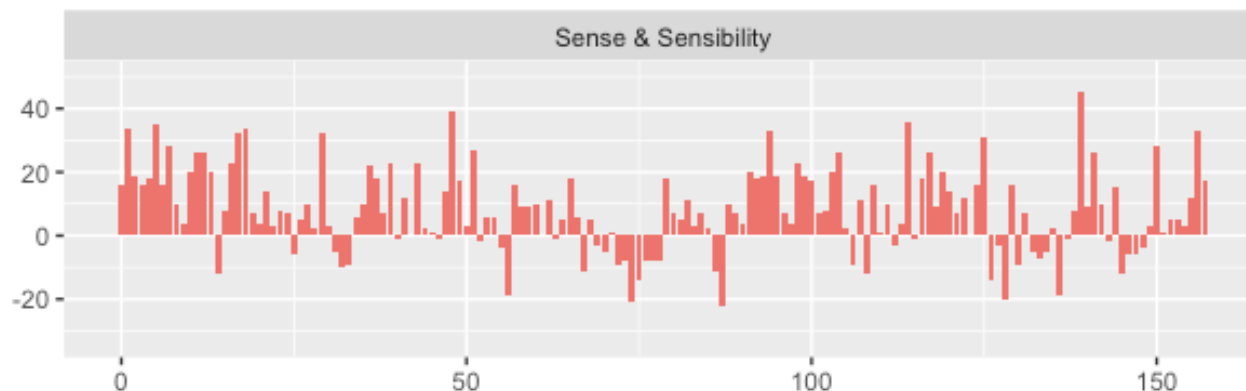
```
View(bing_word_counts)
```

```
## # A tibble: 2,585 x 3
##   word      sentiment      n
##   <chr>    <chr>      <int>
## 1 miss     negative    1855
## 2 well     positive    1523
## 3 good     positive    1380
## 4 great    positive     981
## 5 like     positive     725
## 6 better   positive     639
## 7 enough   positive     613
## 8 happy    positive     534
## 9 love     positive     495
## 10 pleasure positive     462
## # ... with 2,575 more rows
```

Plot the Good and Bad Words Across Each Book

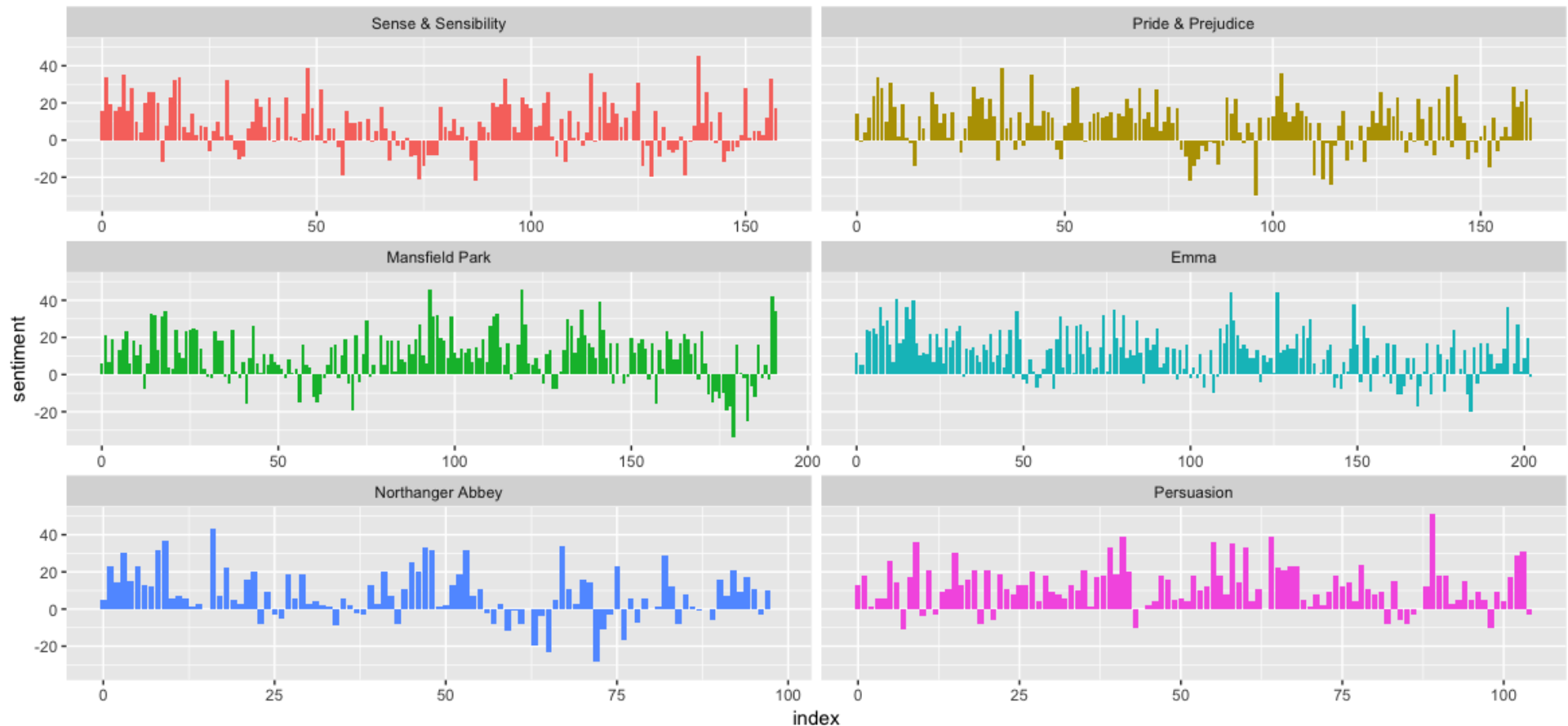
- # plot the sentiment in each book

```
ggplot(janeaustensentiment, aes(index,  
sentiment, fill = book)) + geom_bar(stat =  
"identity", show.legend = FALSE) +  
facet_wrap(~book, ncol = 2, scales = "free_x")
```





Plot the Good and Bad Words Across Each Book



An optimistic writer: there appears to be a similar pattern of optimistic / pessimistic word usage across all her books!

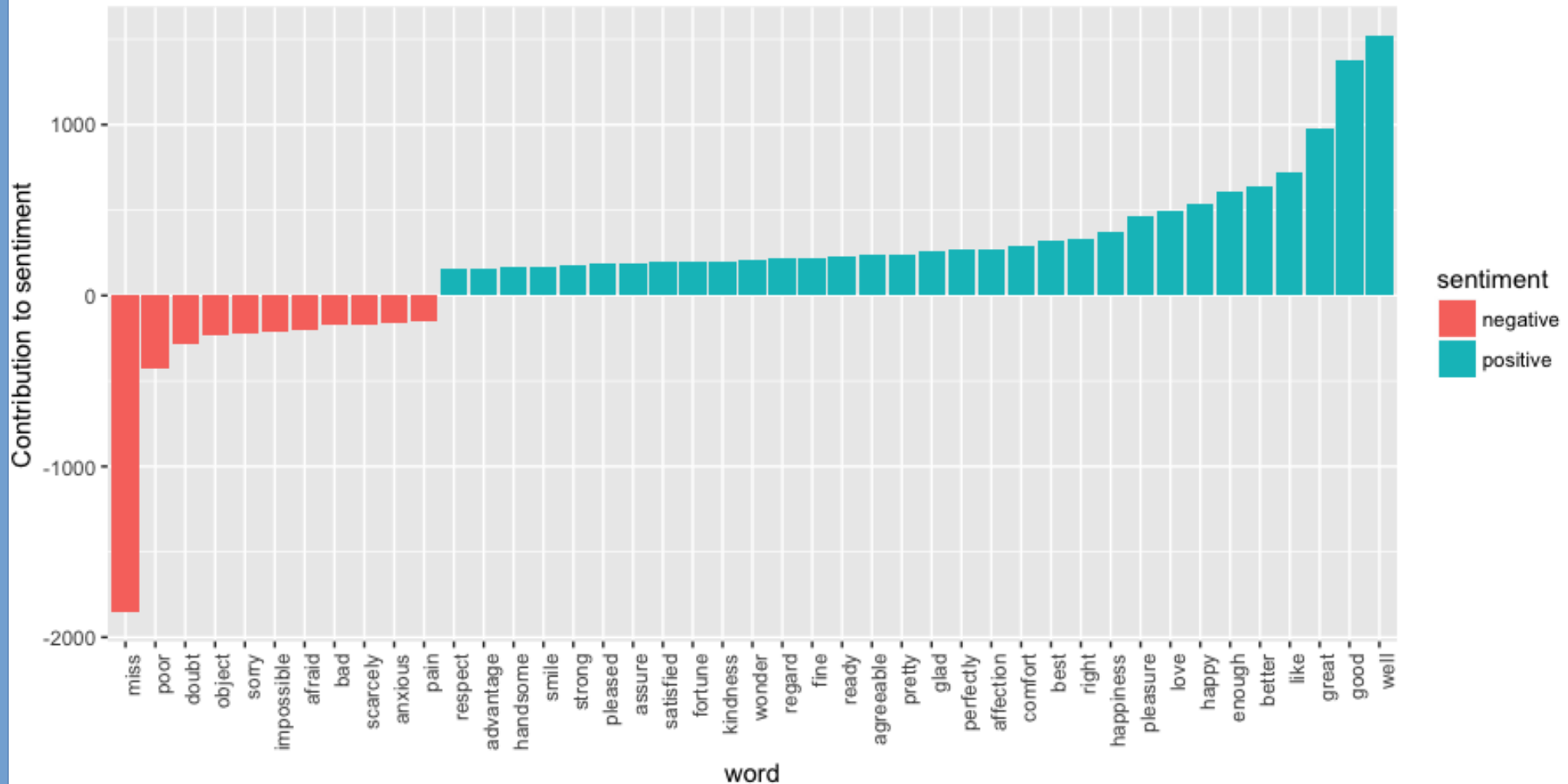
Plot of The Common Positive and Negative Words

- Plot the common positive words across the books.

```
bing_word_counts %>%  
  filter(n > 150) %>%  
  mutate(n = ifelse(sentiment == "negative", -n, n)) %>%  
  mutate(word = reorder(word, n)) %>%  
  ggplot(aes(word, n, fill = sentiment)) +  
  geom_bar(stat = "identity") +  
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +  
  ylab("Contribution to sentiment")
```



Plot of Positive and Negative Sentiment Words



And Now. Back to the Research Question!

- Jane Austen's written work:

How many *Bad* (pessimistic) words did she use?

How many *Good* (optimistic) words did she use?



Get list of Positive and Negative Sentiments

```
bing_word_counts <- tidy_books %>%  
  inner_join(get_sentiments("bing")) %>%  
  count(word, sentiment, sort = TRUE)  
%>%  
  ungroup()  
  
bing_word_counts
```

```
> bing_word_counts  
# A tibble: 2,585 x 3  
  word      sentiment      n  
  <chr>    <chr>    <int>  
1 miss      negative   1855  
2 well      positive   1523  
3 good      positive   1380  
4 great     positive    981  
5 like      positive    725  
6 better    positive    639  
7 enough    positive    613  
8 happy     positive    534  
9 love      positive    495  
10 pleasure positive    462  
# ... with 2,575 more rows
```



What are the Sentiments of Words?

```
bing_word_counts <- tidy_books %>%  
  inner_join(get_sentiments("bing")) %>%  
  count(word, sentiment, sort = TRUE) %>%  
  ungroup()
```

```
bing_word_counts
```

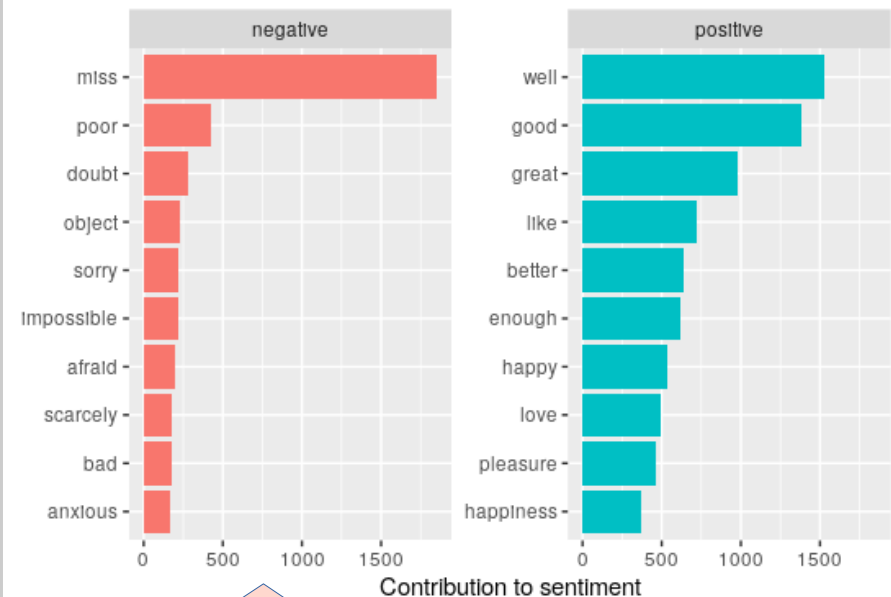
Each word has an associated sentiment. Each sentiment is represented by a number of words in the lexicon. Here we note the number of words that may be associated to each of the sentiments.

```
> bing_word_counts  
# A tibble: 2,585 x 3  
  word      sentiment      n  
  <chr>    <chr>    <int>  
1 miss      negative    1855  
2 well      positive    1523  
3 good      positive    1380  
4 great     positive     981  
5 like      positive     725  
6 better    positive     639  
7 enough    positive     613  
8 happy     positive     534  
9 love      positive     495  
10 pleasure positive     462  
# ... with 2,575 more rows
```



Visually Shown, The Sentiment Words

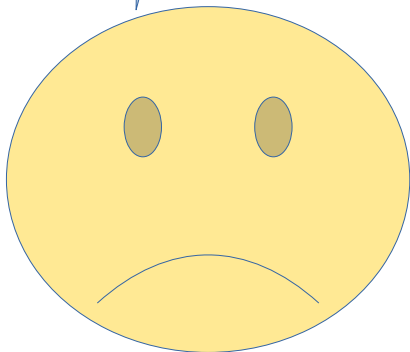
```
bing_word_counts %>%  
  group_by(sentiment) %>%  
  top_n(10) %>%  
  ungroup() %>%  
  mutate(word = reorder(word, n)) %>%  
  ggplot(aes(word, n, fill = sentiment)) +  
  geom_col(show.legend = FALSE) +  
  facet_wrap(~sentiment, scales = "free_y") +  
  labs(y = "Contribution to sentiment",  
       x = NULL) +  
  coord_flip()
```



Gimme the top ten, and then show me how many lexicon words are associated to that sentiment.



Many more words associated to “miss”
(pessimistic max) than “well” (optimistic max)



Could “well” also have other types of optimistic uses
as well? And **Good**? Why not **Great**!



Topic Modelling:

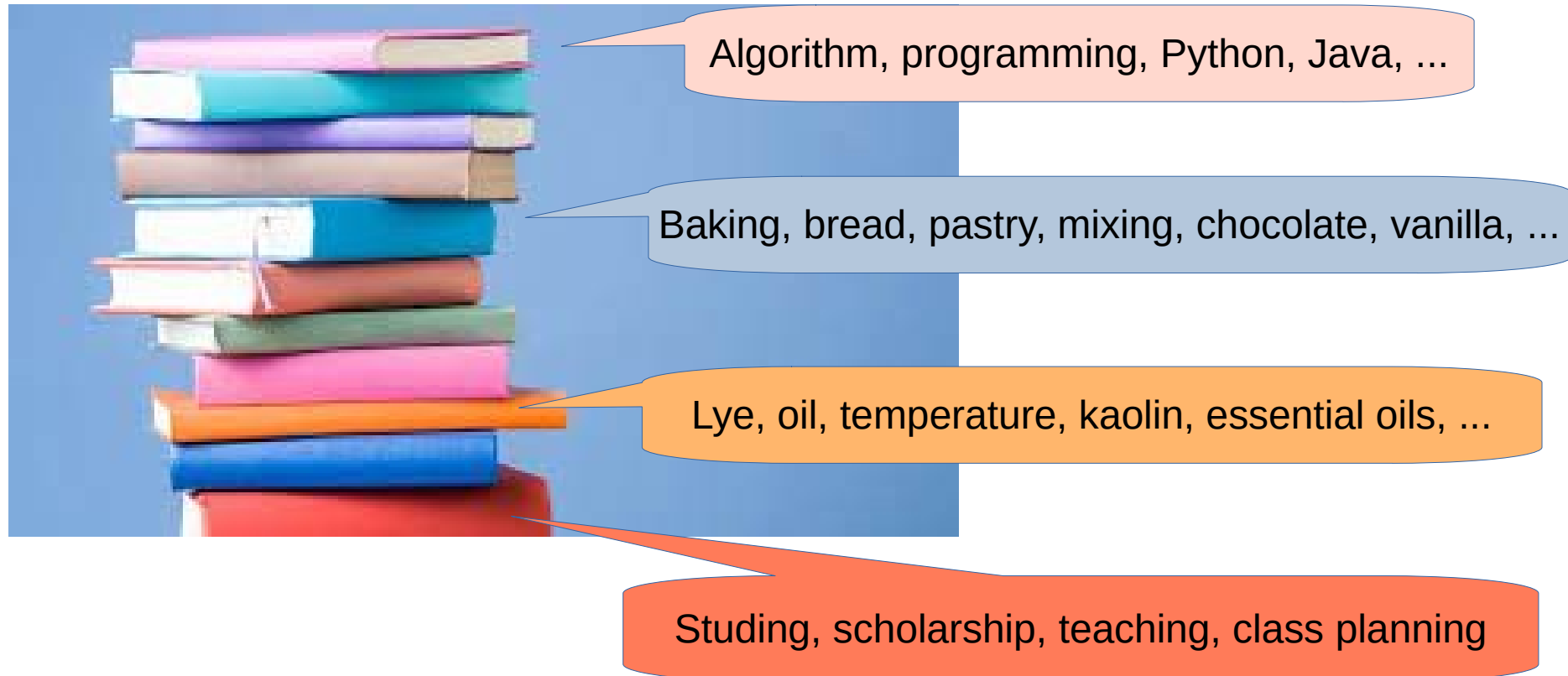
Latent Dirichlet allocation (LDA)

- Document are a mixture of topics, and each topic as a mixture of words.
- Word usage can be used used to describe something “telling” about the topic of the document.
- Example: cooking books contain words about cooking, computer science books use terms about computers ...
- More about this at link:
<https://www.tidytextmining.com/topicmodeling.html>



Topic Modelling:

Latent Dirichlet allocation (LDA)



- We can determine the topic of a book by the most commonly used words, sentiment analysis gives you the over-all feeling of the text.

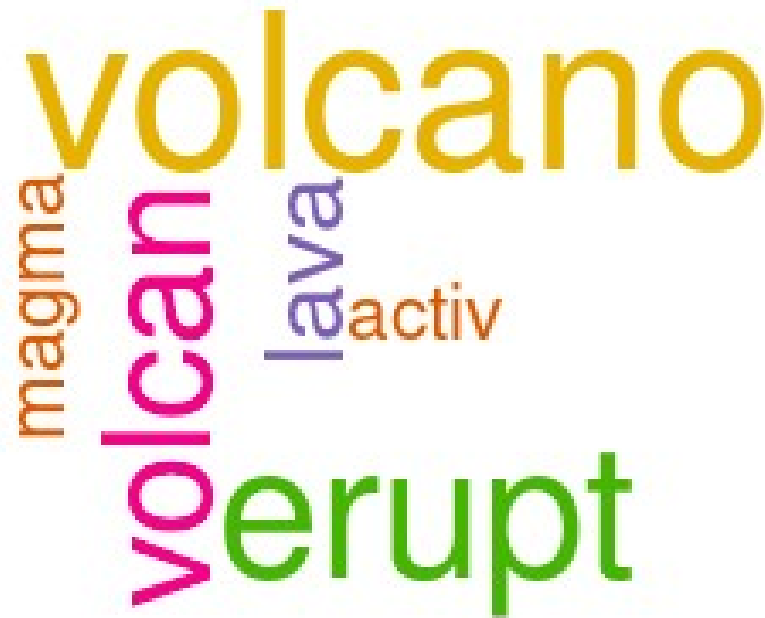
Make your own word clouds: [sandbox/wordCloudDemo.r](https://sandbox.wordclouddemo.com)

Make your own word clouds: [sandbox/wordCloudDemo.ru](https://sandbox.wordCloudDemo.ru)



Topic Modelling:

WikiPage about Volcanoes



Make your own word clouds: [sandbox/wordCloudDemo.r](https://sandbox.wordCloudDemo.r)