Text as Data

Justin Grimmer

Associate Professor
Department of Political Science
Stanford University

October 2nd, 2014
Classification via Dictionary Methods

1) Task

   a) Categorize documents into predetermined categories
   b) Measure documents association with predetermined categories

2) Objective function:

\[ f(\theta, X_i) = \sum_{j=1}^{N} \theta_j X_{ij} \]

where:

- \( \theta = (\theta_1, \theta_2, ..., \theta_N) \) are word weights
- \( X_i = (X_{i1}, X_{i2}, ..., X_{iN}) \) count the occurrence of each corresponding word in document

3) Optimization

\( \rightarrow \) predetermined word list, no task specific optimization

4) Validation (Model checking)

\( \rightarrow \) weight (model) checking, replication of hand coding, face validity
Classification via Dictionary Methods

1) Task
   a) Categorize documents into predetermined categories

2) Objective function:
   \[ f(\theta, X_i) = \sum_{j=1}^{N} \theta_j X_{ij} \]
   where:
   - \( \theta = (\theta_1, \theta_2, ..., \theta_N) \) are word weights
   - \( X_i = (X_{i1}, X_{i2}, ..., X_{iN}) \) count the occurrence of each corresponding word in document

3) Optimization
   \( \rightarrow \) predetermined word list, no task specific optimization

4) Validation (Model checking)
   \( \rightarrow \) weight (model) checking, replication of hand coding, face validity
Classification via Dictionary Methods

1) Task
   a) Categorize documents into predetermined categories
   b) Measure documents association with predetermined categories

2) Objective function:
   \[ f(\theta, X_i) = \sum_{j=1}^{N} \theta_j X_{ij} \sum_{j=1}^{N} X_{ij} \]
   where:
   - \( \theta = (\theta_1, \theta_2, \ldots, \theta_N) \) are word weights
   - \( X_i = (X_{i1}, X_{i2}, \ldots, X_{iN}) \) count the occurrence of each corresponding word in document

3) Optimization
   \( \rightarrow \) predetermined word list, no task specific optimization

4) Validation (Model checking)
   \( \rightarrow \) weight (model) checking, replication of hand coding, face validity
Classification via Dictionary Methods

1) Task
   a) Categorize documents into predetermined categories
   b) Measure documents association with predetermined categories

2) Objective function:

\[ f(\theta, X_i) = \sum_{j=1}^{N} \theta_j X_{ij} / \sum_{j=1}^{N} X_{ij} \]

where:
- \( \theta = (\theta_1, \theta_2, \ldots, \theta_N) \) are word weights
- \( X_i = (X_i^1, X_i^2, \ldots, X_i^N) \) count the occurrence of each corresponding word in document

3) Optimization
   \( \rightarrow \) predetermined word list, no task specific optimization

4) Validation (Model checking)
   \( \rightarrow \) weight (model) checking, replication of hand coding, face validity
Classification via Dictionary Methods

1) Task
   a) Categorize documents into predetermined categories
   b) Measure documents association with predetermined categories

2) Objective function:

\[
f(\theta, X_i) = \frac{\sum_{j=1}^{N} \theta_j X_{ij}}{\sum_{j=1}^{N} X_{ij}}
\]

where:

- \(\theta = (\theta_1, \theta_2, ..., \theta_N)\) are word weights
- \(X_i = (X_{i1}, X_{i2}, ..., X_{iN})\) count the occurrence of each corresponding word in document \(i\)
Classification via Dictionary Methods

1) Task
   a) Categorize documents into predetermined categories
   b) Measure documents association with predetermined categories

2) Objective function:

\[ f(\theta, X_i) = \frac{\sum_{j=1}^{N} \theta_j X_{ij}}{\sum_{j=1}^{N} X_{ij}} \]

where:

- \( \theta = (\theta_1, \theta_2, ..., \theta_N) \) are word weights
- \( X_i = (X_{i1}, X_{i2}, ..., X_{iN}) \) count the occurrence of each corresponding word in document

3) Optimization
   \( \Rightarrow \) predetermined word list, no task specific optimization

4) Validation (Model checking)
   \( \Rightarrow \) weight (model) checking, replication of hand coding, face validity
Classification via Dictionary Methods

1) Task
   a) Categorize documents into predetermined categories
   b) Measure documents association with predetermined categories

2) Objective function:

\[ f(\theta, X_i) = \frac{\sum_{j=1}^{N} \theta_j X_{ij}}{\sum_{j=1}^{N} X_{ij}} \]

where:
   - \( \theta = (\theta_1, \theta_2, \ldots, \theta_N) \) are word weights

Optimization \( \rightarrow \) predetermined word list, no task specific optimization

Validation (Model checking) \( \rightarrow \) weight (model) checking, replication of hand coding, face validity
Classification via Dictionary Methods

1) Task
   a) Categorize documents into predetermined categories
   b) Measure documents association with predetermined categories

2) Objective function:

\[
f(\theta, X_i) = \frac{\sum_{j=1}^{N} \theta_j X_{ij}}{\sum_{j=1}^{N} X_{ij}}
\]

where:
- \( \theta = (\theta_1, \theta_2, \ldots, \theta_N) \) are word weights
- \( X_i = (X_{i1}, X_{i2}, \ldots, X_{iN}) \) count the occurrence of each corresponding word in document \( i \)
Classification via Dictionary Methods

1) Task
   a) Categorize documents into predetermined categories
   b) Measure documents association with predetermined categories

2) Objective function:

\[ f(\theta, X_i) = \frac{\sum_{j=1}^{N} \theta_j X_{ij}}{\sum_{j=1}^{N} X_{ij}} \]

where:
- \( \theta = (\theta_1, \theta_2, \ldots, \theta_N) \) are word weights
- \( X_i = (X_{i1}, X_{i2}, \ldots, X_{iN}) \) count the occurrence of each corresponding word in document \( i \)

3) Optimization \( \rightarrow \) predetermined word list, no task specific optimization

4) Validation (Model checking) \( \rightarrow \) weight (model) checking, replication of hand coding, face validity
Classification via Dictionary Methods

1) Task
   a) Categorize documents into predetermined categories
   b) Measure documents association with predetermined categories

2) Objective function:

\[ f(\theta, X_i) = \frac{\sum_{j=1}^{N} \theta_j X_{ij}}{\sum_{j=1}^{N} X_{ij}} \]

where:
- \( \theta = (\theta_1, \theta_2, \ldots, \theta_N) \) are word weights
- \( X_i = (X_{i1}, X_{i2}, \ldots, X_{iN}) \) count the occurrence of each corresponding word in document \( i \)

3) Optimization \( \mapsto \) predetermined word list, no task specific optimization

4) Validation (Model checking) \( \mapsto \) weight (model) checking, replication of hand coding, face validity
Word Weights: Separating Classes

General Classification Goal: Place documents into categories
Word Weights: Separating Classes

General Classification Goal: Place documents into categories

How To Do Classification?

- Dictionaries:
  - Rely on Humans
  - Humans to identify words that associate with classes
  - Measure how well words separate (positive/negative, emotional, ...)

- Supervised Classification Methods (Later in the Quarter):
  - Rely on statistical models
  - Given set of coded documents, statistical relationship between classes/words
  - Statistical measures of separation

Key point: this is the same task
Word Weights: Separating Classes

General Classification Goal: Place documents into categories

How To Do Classification?

- Dictionaries:
  - Rely on Humans: humans to identify words that associate with classes
  - Measure how well words separate (positive/negative, emotional, ...)

- Supervised Classification Methods (Later in the Quarter):
  - Rely on statistical models
  - Given set of coded documents, statistical relationship between classes/words
  - Statistical measures of separation

Key point: this is the same task
Word Weights: Separating Classes

General Classification Goal: Place documents into categories

How To Do Classification?

- Dictionaries:
  - Rely on Humans—to identify words that associate with classes

Supervised Classification Methods (Later in the Quarter):

- Rely on statistical models
- Given set of coded documents, statistical relationship between classes/words
- Statistical measures of separation

Key point: this is the same task
Word Weights: Separating Classes

General Classification Goal: Place documents into categories

How To Do Classification?

- Dictionaries:
  - Rely on Humans to identify words that associate with classes
  - Measure how well words separate (positive/negative, emotional, ...)

Supervised Classification Methods (Later in the Quarter):

- Rely on statistical models
- Given set of coded documents, statistical relationship between classes/words
- Statistical measures of separation

Key point: this is the same task
Word Weights: Separating Classes

General Classification Goal: Place documents into categories

How To Do Classification?

- Dictionaries:
  - Rely on Humans\(\mapsto\) humans to identify words that associate with classes
  - Measure how well words separate (positive/negative, emotional, ...)

- Supervised Classification Methods (Later in the Quarter):
Word Weights: Separating Classes

General Classification Goal: Place documents into categories

How To Do Classification?

- **Dictionaries:**
  - Rely on Humans to identify words that associate with classes
  - Measure how well words separate (positive/negative, emotional, ...)

- **Supervised Classification Methods (Later in the Quarter):**
  - Rely on statistical models
Word Weights: Separating Classes

General Classification Goal: Place documents into categories

How To Do Classification?

- **Dictionaries:**
  - Rely on Humans to identify words that associate with classes
  - Measure how well words separate (positive/negative, emotional, ...)

- **Supervised Classification Methods (Later in the Quarter):**
  - Rely on statistical models
  - Given set of coded documents, statistical relationship between classes/words
Word Weights: Separating Classes

General Classification Goal: Place documents into categories

How To Do Classification?

- Dictionaries:
  - Rely on Humans to identify words that associate with classes
  - Measure how well words separate (positive/negative, emotional, ...)

- Supervised Classification Methods (Later in the Quarter):
  - Rely on statistical models
  - Given set of coded documents, statistical relationship between classes/words
  - Statistical measures of separation
Word Weights: Separating Classes

General Classification Goal: Place documents into categories

How To Do Classification?

- Dictionaries:
  - Rely on Humans→ humans to identify words that associate with classes
  - Measure how well words separate (positive/negative, emotional, ...)

- Supervised Classification Methods (Later in the Quarter):
  - Rely on statistical models
  - Given set of coded documents, statistical relationship between classes/words
  - Statistical measures of separation

Key point: this is the same task
Types of Classification Problems

**Topic:** What is this text about?

- Policy area of legislation
  - Agriculture, Crime, Environment, ...
- Campaign agendas
  - Abortion, Campaign, Finance, Taxing, ...
- Sentiment: What is said in this text? [Public Opinion]
  - Positions on legislation
    - Support, Ambiguous, Oppose
  - Positions on Court Cases
    - Agree with Court, Disagree with Court
- Style/Tone: How is it said?
  - Taunting in floor statements
    - Partisan Taunt, Intra party taunt, Agency taunt, ...
  - Negative campaigning
    - Negative ad, Positive ad
Types of Classification Problems

**Topic:** What is this text about?
- Policy area of legislation
  ⇒ \{Agriculture, Crime, Environment, ...\}
- Campaign agendas
  ⇒ \{Abortion, Campaign, Finance, Taxing, ...\}
Types of Classification Problems

**Topic:** What is this text about?
- Policy area of legislation
  ⇒ {Agriculture, Crime, Environment, ...}
- Campaign agendas
  ⇒ {Abortion, Campaign, Finance, Taxing, ...}

**Sentiment:** What is said in this text? [Public Opinion]
Types of Classification Problems

**Topic:** What is this text about?
- Policy area of legislation
  ⇒ {Agriculture, Crime, Environment, ...}
- Campaign agendas
  ⇒ {Abortion, Campaign, Finance, Taxing, ...}

**Sentiment:** What is said in this text? [Public Opinion]
- Positions on legislation
  ⇒ {Support, Ambiguous, Oppose}
- Positions on Court Cases
  ⇒ {Agree with Court, Disagree with Court}
- Liberal/Conservative Blog Posts
  ⇒ {Liberal, Middle, Conservative, No Ideology Expressed}
Types of Classification Problems

**Topic:** What is this text about?
- Policy area of legislation
  ⇒ {Agriculture, Crime, Environment, ...}
- Campaign agendas
  ⇒ {Abortion, Campaign, Finance, Taxing, ...}

**Sentiment:** What is said in this text? [Public Opinion]
- Positions on legislation
  ⇒ {Support, Ambiguous, Oppose}
- Positions on Court Cases
  ⇒ {Agree with Court, Disagree with Court}
- Liberal/Conservative Blog Posts
  ⇒ {Liberal, Middle, Conservative, No Ideology Expressed}

**Style/Tone:** How is it said?
Types of Classification Problems

**Topic:** What is this text about?
- Policy area of legislation
  ⇒ \{ Agriculture, Crime, Environment, ... \}
- Campaign agendas
  ⇒ \{ Abortion, Campaign, Finance, Taxing, ... \}

**Sentiment:** What is said in this text? [Public Opinion]
- Positions on legislation
  ⇒ \{ Support, Ambiguous, Oppose \}
- Positions on Court Cases
  ⇒ \{ Agree with Court, Disagree with Court \}
- Liberal/Conservative Blog Posts
  ⇒ \{ Liberal, Middle, Conservative, No Ideology Expressed \}

**Style/Tone:** How is it said?
- Taunting in floor statements
  ⇒ \{ Partisan Taunt, Intra party taunt, Agency taunt, ... \}
- Negative campaigning
  ⇒ \{ Negative ad, Positive ad\}
Pre-existing word weights $\rightarrow$ Dictionaries
Pre-existing word weights $\Rightarrow$ Dictionaries

DICTION

DICTION is a computer-aided text analysis program for Windows® and Mac® that uses a series of dictionaries to search a passage for five semantic features—Activity, Optimism, Certainty, Realism and Commonality—as well as thirty-five sub-features. DICTION uses predefined dictionaries and can use up to thirty custom dictionaries built with words that the user has defined, such as topical or negative words, for particular research needs.
Pre-existing word weights → Dictionaries

DICTION

DICTION 7, now with *Power Mode*, can read a variety of text formats and can accept a large number of files within a single project. Projects containing over 1000 files are analyzed using *power analysis* for enhanced speed and reporting efficiency, with results automatically exported to .csv-formatted spreadsheet file.
Pre-existing word weights $\rightarrow$ Dictionaries

DICTION

On an average computer, DICTION can process over 20,000 passages in about five minutes. DICTION requires 4.9 MB of memory and 38.4 MB of hard disk space.
Pre-existing word weights $\rightarrow$ Dictionaries

DICTION

"provides both social scientific and humanistic understandings"
—Don Waisanen, Baruch College
Pre-existing word weights → Dictionaries

DICTION

DICTION 7 for Mac (Educational) ($219.00)

This is the educational edition of DICTION Version 7 for Mac. You purchase on the following page.
WHAT YEAR IS IT
Dictionary Methods

Many Dictionary Methods (like DICTION)
Dictionary Methods

Many Dictionary Methods (like DICTION)

1) Proprietary
Dictionary Methods

Many Dictionary Methods (like DICTION)

1) Proprietary \rightarrow \text{wrapped in GUI}

Justin Grimmer (Stanford University)
Dictionary Methods

Many Dictionary Methods (like DICTION)

1) Proprietary wrapped in GUI
2) Basic tasks:
Dictionary Methods

Many Dictionary Methods (like DICTION)

1) Proprietary wrapped in GUI
2) Basic tasks:
   a) Count words
Dictionary Methods

Many Dictionary Methods (like DICTION)

1) Proprietary \( \rightarrow \) wrapped in GUI
2) Basic tasks:
   a) Count words
   b) Weighted counts of words
Dictionary Methods

Many Dictionary Methods (like DICTION)

1) Proprietary wrapped in GUI
2) Basic tasks:
   a) Count words
   b) Weighted counts of words
   c) Some graphics
Dictionary Methods

Many Dictionary Methods (like DICTION)

1) Proprietary wrapped in GUI
2) Basic tasks:
   a) Count words
   b) Weighted counts of words
   c) Some graphics
3) Pricey inexplicably
DICTION

---

Certain, Uncertain ≈ 10,000 words

Applies DICTION to a wide array of political texts

Examine specific periods of American political history

Justin Grimmer (Stanford University)

Text as Data

October 2nd, 2014
DICTION

- { Certain, Uncertain }

Applies DICTION to a wide array of political texts
Examine specific periods of American political history

Justin Grimmer (Stanford University)
DICTION

- \{ Certain, Uncertain \}
  , \{ Optimistic, Pessimistic \}
DICTION

- \{ Certain, Uncertain \}
  , \{ Optimistic, Pessimistic \}
- \approx 10,000\text{ words}
DICTION

- \{ Certain, Uncertain \}, \{ Optimistic, Pessimistic \}
- \approx 10,000 \text{ words}

Applies DICTION to a wide array of political texts
DICTION

- { Certain, Uncertain }
  , { Optimistic, Pessimistic }
- ≈ 10,000 words
Applies DICTION to a wide array of political texts
Examine specific periods of American political history
Other Dictionaries

1) General Inquirer Database
   (http://www.wjh.harvard.edu/~inquirer/)
   - Stone, P.J., Dumphy, D.C., and Ogilvie, D.M. (1966)
     The General Inquirer: A Computer Approach to Content Analysis
     - {Positive, Negative}
     - 3627 negative and positive word strings
     - Workhorse for classification across many domains/papers

2) Linguistic Inquiry Word Count (LIWC)
   - Creation process:
     1) Generate word list for categories
        ⇝ "We drew on common emotion rating scales...Roget's Thesaurus...standard English dictionaries. [then]
        brainstorming sessions among 3-6 judges were held" to generate other words
     2) Judge round
        ⇝ (a) Does the word belong? (b) What other categories might it belong to?
     - {Positive emotion, Negative emotion}
     - 2300 words grouped into 70 classes
     - Harvard-IV-4
     - Affective Norms for English Words (we'll discuss this more later)

...
Other Dictionaries

1) General Inquirer Database
   (http://www.wjh.harvard.edu/~inquirer/)

2) Linguistic Inquiry Word Count (LIWC)
   - Creation process:
     1) Generate word list for categories
        ⇝ We drew on common emotion rating scales...Roget's Thesaurus...standard English dictionaries. [then] brain-storming sessions among 3-6 judges were held to generate other words
     2) Judge round
        (a) Does the word belong? (b) What other categories might it belong to?
   - 2300 words grouped into 70 classes
   - Harvard-IV-4
   - Affective Norms for English Words (we'll discuss this more later)
Other Dictionaries

1) General Inquirer Database
   (http://www.wjh.harvard.edu/~inquirer/)
   - { Positive, Negative }

2) Linguistic Inquiry Word Count (LIWC)
   - Creation process:
     1) Generate word list for categories
     2) Judge round
        (a) Does the word belong?
        (b) What other categories might it belong to?
   - { Positive emotion, Negative emotion }
   - 2300 words grouped into 70 classes
   - Harvard-IV-4
   - Affective Norms for English Words (we'll discuss this more later)
Other Dictionaries

1) General Inquirer Database
   (http://www.wjh.harvard.edu/~inquirer/ )
   - { Positive, Negative }
   - 3627 negative and positive word strings

2) Linguistic Inquiry Word Count (LIWC)
   - Creation process:
     1) Generate word list for categories
        ⇝ “We drew on common emotion rating scales...Roget’s Thesaurus...standard English dictionaries. [then] brainstorming sessions among 3-6 judges were held” to generate other words
     2) Judge round
        (a) Does the word belong? (b) What other categories might it belong to?
   - { Positive emotion, Negative emotion }
   - 2300 words grouped into 70 classes
   - Harvard-IV-4
   - Affective Norms for English Words (we’ll discuss this more later)
Other Dictionaries

1) General Inquirer Database
   (http://www.wjh.harvard.edu/~inquirer/ )
   - { Positive, Negative }
   - 3627 negative and positive word strings
   - Workhorse for classification across many domains/papers
Other Dictionaries

1) General Inquirer Database
   (http://www.wjh.harvard.edu/~inquirer/)
   - \{ Positive, Negative \}
   - 3627 negative and positive word strings
   - Workhorse for classification across many domains/papers

2) Linguistic Inquiry Word Count (LIWC)

Justin Grimmer (Stanford University)
Other Dictionaries

1) General Inquirer Database
   (http://www.wjh.harvard.edu/~inquirer/ )
   - { Positive, Negative }
   - 3627 negative and positive word strings
   - Workhorse for classification across many domains/papers

2) Linguistic Inquiry Word Count (LIWC)
   - Creation process:
Other Dictionaries

1) General Inquirer Database
   (http://www.wjh.harvard.edu/~inquirer/)  
   - { Positive, Negative }  
   - 3627 negative and positive word strings  
   - Workhorse for classification across many domains/papers

2) Linguistic Inquiry Word Count (LIWC)
   - Creation process:
     1) Generate word list for categories “We drew on common emotion rating scales...Roget’s Thesaurus...standard English dictionaries. [then] brain-storming sessions among 3-6 judges were held” to generate other words
Other Dictionaries

1) General Inquirer Database
   (http://www.wjh.harvard.edu/~inquirer/)
   - \{ Positive, Negative \}
   - 3627 negative and positive word strings
   - Workhorse for classification across many domains/papers

2) Linguistic Inquiry Word Count (LIWC)
   - Creation process:
     1) Generate word list for categories
        “We drew on common emotion rating scales...Roget’s Thesaurus...standard English dictionaries. [then] brain-storming sessions among 3-6 judges were held” to generate other words
     2) Judge round
        (a) Does the word belong? (b) What other categories might it belong to?
Other Dictionaries

1) General Inquirer Database
   (http://www.wjh.harvard.edu/~inquirer/)
   - { Positive, Negative }
   - 3627 negative and positive word strings
   - Workhorse for classification across many domains/papers

2) Linguistic Inquiry Word Count (LIWC)
   - Creation process:
     1) Generate word list for categories
        “We drew on common emotion rating scales...Roget’s Thesaurus...standard English dictionaries. [then] brain-storming sessions among 3-6 judges were held” to generate other words
     2) Judge round
        (a) Does the word belong? (b) What other categories might it belong to?
   - { Positive emotion, Negative emotion }
Other Dictionaries

1) General Inquirer Database
   (http://www.wjh.harvard.edu/~inquirer/)
   - { Positive, Negative }
   - 3627 negative and positive word strings
   - Workhorse for classification across many domains/papers

2) Linguistic Inquiry Word Count (LIWC)
   - Creation process:
     1) Generate word list for categories “We drew on common emotion rating scales...Roget’s Thesaurus...standard English dictionaries. [then] brain-storming sessions among 3-6 judges were held” to generate other words
     2) Judge round (a) Does the word belong? (b) What other categories might it belong to?
   - { Positive emotion, Negative emotion }
   - 2300 words grouped into 70 classes
Other Dictionaries

1) General Inquirer Database
   (http://www.wjh.harvard.edu/~inquirer/)
   - \{ Positive, Negative \}
   - 3627 negative and positive word strings
   - Workhorse for classification across many domains/papers

2) Linguistic Inquiry Word Count (LIWC)
   - Creation process:
     1) Generate word list for categories
        “We drew on common emotion rating scales...Roget’s Thesaurus...standard English dictionaries. [then] brain-storming sessions among 3-6 judges were held” to generate other words
     2) Judge round
        (a) Does the word belong? (b) What other categories might it belong to?
        - \{ Positive emotion, Negative emotion \}
        - 2300 words grouped into 70 classes
   - Harvard-IV-4
Other Dictionaries

1) General Inquirer Database
   (http://www.wjh.harvard.edu/~inquirer/)
   - { Positive, Negative }
   - 3627 negative and positive word strings
   - Workhorse for classification across many domains/papers

2) Linguistic Inquiry Word Count (LIWC)
   - Creation process:
     1) Generate word list for categories "We drew on common emotion rating scales...Roget’s Thesaurus...standard English dictionaries. [then] brain-storming sessions among 3-6 judges were held” to generate other words
     2) Judge round (a) Does the word belong? (b) What other categories might it belong to?
        - { Positive emotion, Negative emotion }
        - 2300 words grouped into 70 classes
   - Harvard-IV-4
   - Affective Norms for English Words (we’ll discuss this more later)
Other Dictionaries

1) General Inquirer Database
   (http://www.wjh.harvard.edu/~inquirer/)
     Inquirer: A Computer Approach to Content Analysis
   - \{ Positive, Negative \}
   - 3627 negative and positive word strings
   - Workhorse for classification across many domains/papers

2) Linguistic Inquiry Word Count (LIWC)
   - Creation process:
     1) Generate word list for categories \rightarrow \text{“We drew on common emotion}
        rating scales...Roget’s Thesaurus...standard English dictionaries. [then]
        brain-storming sessions among 3-6 judges were held” to generate other
        words
     2) Judge round \rightarrow (a) Does the word belong? (b) What other categories
        might it belong to?
        - \{ Positive emotion, Negative emotion \}
        - 2300 words grouped into 70 classes
   - Harvard-IV-4
   - Affective Norms for English Words (we’ll discuss this more later)
   - ...
Generating New Words

Three ways to create dictionaries (non-exhaustive):

- Statistical methods
- Manual generation
- Careful thought (prayer? epiphanies? divine intervention?) about useful words
- Populations of people who are surprisingly willing to perform ill-defined tasks
  a) Undergraduates: Pizza → Research Output
  b) Mechanical turkers
  - Example: {Happy, Unhappy}
  - Ask turkers: how happy is elevator, car, pretty, young
  Output as dictionary
Generating New Words

Three ways to create dictionaries (non-exhaustive):
  - Statistical methods
  - Manual generation
  - Careful thought (prayer? epiphanies? divine intervention?) about useful words
    - Populations of people who are surprisingly willing to perform ill-defined tasks
      a) Undergraduates: Pizza → Research Output
      b) Mechanical turkers
        - Example: \{Happy, Unhappy\}
        - Ask turkers: how happy is elevator, car, pretty, young
        - Output as dictionary
Generating New Words

Three ways to create dictionaries (non-exhaustive):
- Statistical methods
- Manual generation
  - Example: {Happy, Unhappy}
  - Ask turkers: how happy is elevator, car, pretty, young
  Output as dictionary
Generating New Words

Three ways to create dictionaries (non-exhaustive):

- Statistical methods

  next Tuesday

- Manual generation

  - Careful thought (prayer? epiphanies? divine intervention?) about useful words

- Populations of people who are surprisingly willing to perform ill-defined tasks

  a) Undergraduates: Pizza → Research Output

  b) Mechanical turkers:

     - Example: {Happy, Unhappy}

     - Ask turkers: how happy is elevator, car, pretty, young

     Output as dictionary
Generating New Words

Three ways to create dictionaries (non-exhaustive):

- Statistical methods
- Manual generation
  - Careful thought (prayer? epiphanies? divine intervention?) about useful words
- Populations of people who are surprisingly willing to perform ill-defined tasks
Generating New Words

Three ways to create dictionaries (non-exhaustive):

- Statistical methods
- Manual generation
  - Careful thought (prayer? epiphanies? divine intervention?) about useful words
- Populations of people who are surprisingly willing to perform ill-defined tasks
  a) Undergraduates: Pizza → Research Output
Generating New Words

Three ways to create dictionaries (non-exhaustive):

- Statistical methods
- Manual generation
  - Careful thought (prayer? epiphanies? divine intervention?) about useful words
- Populations of people who are surprisingly willing to perform ill-defined tasks
  a) Undergraduates: Pizza \rightarrow Research Output
  b) Mechanical turkers
Three ways to create dictionaries (non-exhaustive):

- Statistical methods
  - next Tuesday
- Manual generation
  - Careful thought (prayer? epiphanies? divine intervention?) about useful words
- Populations of people who are surprisingly willing to perform ill-defined tasks
  a) Undergraduates: Pizza → Research Output
  b) Mechanical turkers
     - Example: \{ Happy, Unhappy \}
Generating New Words

Three ways to create dictionaries (non-exhaustive):

- Statistical methods → next Tuesday
- Manual generation
  - Careful thought (prayer? epiphanies? divine intervention?) about useful words
- Populations of people who are surprisingly willing to perform ill-defined tasks
  a) Undergraduates: Pizza → Research Output
  b) Mechanical turkers
     - Example: { Happy, Unhappy }
     - Ask turkers: how happy is
Generating New Words

Three ways to create dictionaries (non-exhaustive):

- Statistical methods
- Manual generation
  - Careful thought (prayer? epiphanies? divine intervention?) about useful words
- Populations of people who are surprisingly willing to perform ill-defined tasks
  a) Undergraduates: Pizza → Research Output
  b) Mechanical turkers
    - Example: { Happy, Unhappy }
Generating New Words

Three ways to create dictionaries (non-exhaustive):

- **Statistical methods**
  - next Tuesday
- **Manual generation**
  - Careful thought (prayer? epiphanies? divine intervention?) about useful words
- **Populations of people who are surprisingly willing to perform ill-defined tasks**
  a) Undergraduates: Pizza → Research Output
  b) Mechanical turkers
    - Example: \{ Happy, Unhappy \}
    - Ask turkers: how happy is elevator, car, pretty, young
    Output as dictionary
Applying Methods to Documents

Applying the model:

- Vector of word counts: $X_i = (X_{i1}, X_{i2}, ..., X_{iK}, (i = 1, ..., N))$
- Weights attached to words: $\theta = (\theta_1, \theta_2, ..., \theta_K)$
- $\theta_k \in \{0, 1\}$
- $\theta_k \in \{-1, 0, 1\}$
- $\theta_k \in \{-2, -1, 0, 1, 2\}$
- $\theta_k \in \mathbb{R}$

For each document $i$, calculate score for document $Y_i = \sum_{k=1}^{K} \theta_k X_{ik} / \sum_{k=1}^{K} X_{ik}$

$Y_i \approx \text{continuous} \xrightarrow{\text{Classification}}$

$Y_i > 0 \Rightarrow \text{Positive Category}$

$Y_i < 0 \Rightarrow \text{Negative Category}$

$Y_i \approx 0 \Rightarrow \text{Ambiguous}$
Applying Methods to Documents
Applying the model:
- Vector of word counts: \( \mathbf{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iK}, (i = 1, \ldots, N) \)
Applying Methods to Documents

Applying the model:

- Vector of word counts: \( \mathbf{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iK}), (i = 1, \ldots, N) \)
- Weights attached to words \( \boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_K) \)
Applying Methods to Documents

Applying the model:
- Vector of word counts: $\mathbf{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iK}, (i = 1, \ldots, N)$
- Weights attached to words $\mathbf{\theta} = (\theta_1, \theta_2, \ldots, \theta_K)$
  - $\theta_k \in \{0, 1\}$
Applying Methods to Documents

Applying the model:

- Vector of word counts: \( \mathbf{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iK}, (i = 1, \ldots, N) \)
- Weights attached to words \( \boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_K) \)
  - \( \theta_k \in \{0, 1\} \)
  - \( \theta_k \in \{-1, 0, 1\} \)

For each document \( i \) calculate score for document

\[
Y_i = \sum_{k=1}^{K} \theta_k X_{ik} 
\]

\[
Y_i \approx \text{continuous} \Rightarrow \text{Classification}
\]

\[
Y_i > 0 \Rightarrow \text{Positive Category}
\]

\[
Y_i < 0 \Rightarrow \text{Negative Category}
\]

\[
Y_i \approx 0 \Rightarrow \text{Ambiguous}
\]
Applying Methods to Documents

Applying the model:
- Vector of word counts: \( \mathbf{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iK}), (i = 1, \ldots, N) \)
- Weights attached to words \( \boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_K) \)
  - \( \theta_k \in \{0, 1\} \)
  - \( \theta_k \in \{-1, 0, 1\} \)
  - \( \theta_k \in \{-2, -1, 0, 1, 2\} \)
Applying Methods to Documents

Applying the model:
- Vector of word counts: $X_i = (X_{i1}, X_{i2}, \ldots, X_{iK}, \ (i = 1, \ldots, N)$
- Weights attached to words $\theta = (\theta_1, \theta_2, \ldots, \theta_K)$
  - $\theta_k \in \{0, 1\}$
  - $\theta_k \in \{-1, 0, 1\}$
  - $\theta_k \in \{-2, -1, 0, 1, 2\}$
  - $\theta_k \in \mathbb{R}$

For each document $i$ calculate score for document $Y_i = \sum_{K}^{k=1} \theta_k X_{ik} / \sum_{K}^{k=1} X_k$

$Y_i \approx \text{continuous} \rightarrow \text{Classification}$

$Y_i > 0 \Rightarrow \text{Positive Category}$

$Y_i < 0 \Rightarrow \text{Negative Category}$

$Y_i \approx 0 \Rightarrow \text{Ambiguous}$
Applying Methods to Documents

Applying the model:

- Vector of word counts: \( X_i = (X_{i1}, X_{i2}, \ldots, X_{iK}, (i = 1, \ldots, N) \)
- Weights attached to words \( \theta = (\theta_1, \theta_2, \ldots, \theta_K) \)
  - \( \theta_k \in \{0, 1\} \)
  - \( \theta_k \in \{-1, 0, 1\} \)
  - \( \theta_k \in \{-2, -1, 0, 1, 2\} \)
  - \( \theta_k \in \mathbb{R} \)

For each document \( i \) calculate score for document
Applying Methods to Documents

Applying the model:

- Vector of word counts: \( \mathbf{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iK}, (i = 1, \ldots, N) \)
- Weights attached to words \( \mathbf{\theta} = (\theta_1, \theta_2, \ldots, \theta_K) \)
  - \( \theta_k \in \{0, 1\} \)
  - \( \theta_k \in \{-1, 0, 1\} \)
  - \( \theta_k \in \{-2, -1, 0, 1, 2\} \)
  - \( \theta_k \in \mathbb{R} \)

For each document \( i \) calculate score for document

\[
Y_i = \frac{\sum_{k=1}^{K} \theta_k X_{ik}}{\sum_{k=1}^{K} X_k}
\]
Applying Methods to Documents

Applying the model:
- Vector of word counts: \( \mathbf{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iK}, (i = 1, \ldots, N) \)
- Weights attached to words \( \mathbf{\theta} = (\theta_1, \theta_2, \ldots, \theta_K) \)
  - \( \theta_k \in \{0, 1\} \)
  - \( \theta_k \in \{-1, 0, 1\} \)
  - \( \theta_k \in \{-2, -1, 0, 1, 2\} \)
  - \( \theta_k \in \mathbb{R} \)

For each document \( i \) calculate score for document

\[
Y_i = \frac{\sum_{k=1}^{K} \theta_k X_{ik}}{\sum_{k=1}^{K} X_k}
\]

\[
Y_i = \frac{\mathbf{\theta}^\prime \mathbf{X}_i}{\mathbf{X}_i^\prime 1}
\]
Applying Methods to Documents

Applying the model:
- Vector of word counts: \( \mathbf{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iK}, (i = 1, \ldots, N) \)
- Weights attached to words \( \mathbf{\theta} = (\theta_1, \theta_2, \ldots, \theta_K) \)
  - \( \theta_k \in \{0, 1\} \)
  - \( \theta_k \in \{-1, 0, 1\} \)
  - \( \theta_k \in \{-2, -1, 0, 1, 2\} \)
  - \( \theta_k \in \mathbb{R} \)

For each document \( i \) calculate score for document

\[
Y_i = \frac{\sum_{k=1}^{K} \theta_k X_{ik}}{\sum_{k=1}^{K} X_k}
\]

\[
Y_i = \frac{\mathbf{\theta}' \mathbf{X}_i}{\mathbf{X}_i' \mathbf{1}}
\]

\( Y_i \approx \text{continuous} \leadsto \text{Classification} \)
Applying Methods to Documents

Applying the model:
- Vector of word counts: \( \mathbf{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iK}, (i = 1, \ldots, N) \)
- Weights attached to words \( \mathbf{\theta} = (\theta_1, \theta_2, \ldots, \theta_K) \)
  - \( \theta_k \in \{0, 1\} \)
  - \( \theta_k \in \{-1, 0, 1\} \)
  - \( \theta_k \in \{-2, -1, 0, 1, 2\} \)
  - \( \theta_k \in \mathbb{R} \)

For each document \( i \) calculate score for document

\[
Y_i = \frac{\sum_{k=1}^{K} \theta_k X_{ik}}{\sum_{k=1}^{K} X_k}
\]

\[
Y_i = \frac{\mathbf{\theta}' \mathbf{X}_i}{\mathbf{X}_i' \mathbf{1}}
\]

\( Y_i \approx \text{continuous} \leadsto \text{Classification} \)

\( Y_i > 0 \Rightarrow \text{Positive Category} \)
Applying Methods to Documents

Applying the model:
- Vector of word counts: $\mathbf{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iK}, (i = 1, \ldots, N)$
- Weights attached to words $\theta = (\theta_1, \theta_2, \ldots, \theta_K)$
  - $\theta_k \in \{0, 1\}$
  - $\theta_k \in \{-1, 0, 1\}$
  - $\theta_k \in \{-2, -1, 0, 1, 2\}$
  - $\theta_k \in \mathbb{R}$

For each document $i$ calculate score for document

$$Y_i = \frac{\sum_{k=1}^{K} \theta_k X_{ik}}{\sum_{k=1}^{K} X_k}$$

$$Y_i = \frac{\theta' \mathbf{X}_i}{\mathbf{X}_i' \mathbf{1}}$$

$Y_i \approx$ continuous $\leadsto$ Classification

$Y_i > 0 \Rightarrow$ Positive Category

$Y_i < 0 \Rightarrow$ Negative Category
Applying Methods to Documents

Applying the model:
- Vector of word counts: \( \mathbf{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iK}, (i = 1, \ldots, N) \)
- Weights attached to words \( \mathbf{\theta} = (\theta_1, \theta_2, \ldots, \theta_K) \)
  - \( \theta_k \in \{0, 1\} \)
  - \( \theta_k \in \{-1, 0, 1\} \)
  - \( \theta_k \in \{-2, -1, 0, 1, 2\} \)
  - \( \theta_k \in \mathbb{R} \)

For each document \( i \) calculate score for document

\[
Y_i = \frac{\sum_{k=1}^{K} \theta_k X_{ik}}{\sum_{k=1}^{K} X_k}
\]

\[
Y_i = \frac{\mathbf{\theta}^\top \mathbf{X}_i}{\mathbf{X}_i \mathbf{1}}
\]

\( Y_i \approx \text{continuous} \sim \text{Classification} \)

\( Y_i > 0 \Rightarrow \text{Positive Category} \)

\( Y_i < 0 \Rightarrow \text{Negative Category} \)

\( Y_i \approx 0 \text{ Ambiguous} \)
Applying a Dictionary to Press Releases
Applying a Dictionary to Press Releases

Applying a Dictionary to Press Releases

- Dictionary from Neal Caren’s website ~ Theresa Wilson, Janyce Wiebe, and Paul Hoffman’s dictionary
Applying a Dictionary to Press Releases

- Dictionary from Neal Caren’s website \(\sim\) Theresa Wilson, Janyce Wiebe, and Paul Hoffman’s dictionary
- Create positive/negative score for press releases.
Applying a Dictionary to Press Releases

- Dictionary from Neal Caren’s website ⇝ Theresa Wilson, Janyce Wiebe, and Paul Hoffman’s dictionary
- Create positive/negative score for press releases.

Python code and press releases
Examining Positive and Negative Statements in Press Releases

Least positive members of Congress:
1) Dan Burton, 2008
2) Nancy Pelosi, 2007
3) Mike Pence, 2007
4) John Boehner, 2009
5) Jeff Flake, (basically all years)
6) Eric Cantor, 2009
7) Tom Price, 2010

Legislators who are more extreme ⇝ less positive in press releases
Examining Positive and Negative Statements in Press Releases

Least positive members of Congress:

1) Dan Burton, 2008
2) Nancy Pelosi, 2007
3) Mike Pence, 2007
4) John Boehner, 2009
5) Jeff Flake, (basically all years)
6) Eric Cantor, 2009
7) Tom Price, 2010

Legislators who are more extreme ⇝ less positive in press releases
Examining Positive and Negative Statements in Press Releases

Least positive members of Congress:

1) Dan Burton, 2008
Examining Positive and Negative Statements in Press Releases

Least positive members of Congress:
1) Dan Burton, 2008
2) Nancy Pelosi, 2007
Examining Positive and Negative Statements in Press Releases

Least positive members of Congress:

1) Dan Burton, 2008
2) Nancy Pelosi, 2007
3) Mike Pence 2007
Examining Positive and Negative Statements in Press Releases

Least positive members of Congress:

1) Dan Burton, 2008
2) Nancy Pelosi, 2007
3) Mike Pence, 2007
4) John Boehner, 2009
Examining Positive and Negative Statements in Press Releases

Least positive members of Congress:

1) Dan Burton, 2008
2) Nancy Pelosi, 2007
3) Mike Pence 2007
4) John Boehner, 2009
5) Jeff Flake, (basically all years)
Examining Positive and Negative Statements in Press Releases

Least positive members of Congress:

1) Dan Burton, 2008
2) Nancy Pelosi, 2007
3) Mike Pence 2007
4) John Boehner, 2009
5) Jeff Flake, (basically all years)
6) Eric Cantor, 2009
Examining Positive and Negative Statements in Press Releases

Least positive members of Congress:

1) Dan Burton, 2008
2) Nancy Pelosi, 2007
3) Mike Pence 2007
4) John Boehner, 2009
5) Jeff Flake, (basically all years)
6) Eric Cantor, 2009
7) Tom Price, 2010
Examining Positive and Negative Statements in Press Releases

Least positive members of Congress:

1) Dan Burton, 2008
2) Nancy Pelosi, 2007
3) Mike Pence 2007
4) John Boehner, 2009
5) Jeff Flake, (basically all years)
6) Eric Cantor, 2009
7) Tom Price, 2010

Legislators who are more extreme \(\rightarrow\) less positive in press releases
Examining Positive and Negative Statements in Press Releases
Examining Positive and Negative Statements in Press Releases

- Credit Claiming press release: 9.1 percentage points “more positive” than a non-credit claiming press release
Examining Positive and Negative Statements in Press Releases

- Credit Claiming press release: 9.1 percentage points “more positive” than a non-credit claiming press release
- Anti-spending press release: 10.6 percentage points “less positive” than a non-anti spending press release
Examining Positive and Negative Statements in Press Releases
Examining Positive and Negative Statements in Press Releases
Examining Positive and Negative Statements in Press Releases

![Scatter plot showing the relationship between Anti Spending and Positive statements.](image)
Dictionary methods are context invariant
Methodological Issues/Problems with Dictionaries

Dictionary methods are context invariant

- No optimization step $\Rightarrow$ same word weights regardless of texts

Just because dictionaries provide measures labeled “positive” or “negative” it doesn’t mean they are accurate measures in your text (!!!)

Validation

Justin Grimmer (Stanford University)
Dictionary methods are context invariant

- No optimization step $\Rightarrow$ same word weights regardless of texts
- Optimization $\Rightarrow$ incorporate information specific to context
Dictionary methods are context invariant

- No optimization step $\Rightarrow$ same word weights regardless of texts
- Optimization $\Rightarrow$ incorporate information specific to context
- Without optimization $\Rightarrow$ unclear about dictionaries performance
Methodological Issues/Problems with Dictionaries

Dictionary methods are context invariant

- No optimization step $\Rightarrow$ same word weights regardless of texts
- Optimization $\Rightarrow$ incorporate information specific to context
- Without optimization $\Rightarrow$ unclear about dictionaries performance

Just because dictionaries provide measures labeled “positive” or “negative” it doesn’t mean they are accurate measures in your text (!!!!)
Methodological Issues/Problems with Dictionaries

Dictionary methods are context invariant

- No optimization step \(\rightarrow\) same word weights regardless of texts
- Optimization\(\rightarrow\) incorporate information specific to context
- Without optimization\(\rightarrow\) unclear about dictionaries performance

Just because dictionaries provide measures labeled “positive” or “negative” it doesn’t mean they are accurate measures in your text (!!!!)

Validation
Validation

Classification Validity:

- Training: build dictionary on subset of documents with known labels
- Test: apply dictionary method to other documents with known labels
- Requires hand coded documents
- Hand coded documents useful for other reasons
- Is the classification scheme well defined for your texts?
- Can humans accomplish the coding task?
- Is the dictionary your using appropriate?

Replicate classification exercise

- How well does our method perform on held out documents?
- Why held out?

Over fitting

- Using off-the-shelf dictionary: all labeled documents to test
- Supervised learning classification: (Cross) validation
Validation

Classification Validity:
- **Training:** build dictionary on subset of documents with known labels

- **Test:** apply dictionary method to other documents with known labels

- Requires hand coded documents

- Hand coded documents useful for other reasons

- Is the classification scheme well defined for your texts?

- Can humans accomplish the coding task?

- Is the dictionary your using appropriate?

Replicate classification exercise
- How well does our method perform on held out documents?

- Why held out?

Over fitting
- Using off-the-shelf dictionary: all labeled documents to test
- Supervised learning classification: (Cross)validation
Validation

Classification Validity:
- **Training**: build dictionary on subset of documents with known labels
- **Test**: apply dictionary method to other documents with known labels

- Requires hand coded documents
- Hand coded documents useful for other reasons
- Is the classification scheme well defined for your texts?
- Can humans accomplish the coding task?
- Is the dictionary your using appropriate?

Replicate classification exercise

- How well does our method perform on held out documents?
- Why held out?

Over fitting

- Using off-the-shelf dictionary: all labeled documents to test
- Supervised learning classification: (Cross)validation
Validation

Classification Validity:
- **Training**: build dictionary on subset of documents with known labels
- **Test**: apply dictionary method to other documents with known labels
- Requires hand coded documents

- Hand coded documents useful for other reasons
- Is the classification scheme well defined for your texts?
- Can humans accomplish the coding task?
- Is the dictionary your using appropriate?

Replicate classification exercise
- How well does our method perform on held out documents?
- Why held out?

Over fitting
- Using off-the-shelf dictionary: all labeled documents to test
- Supervised learning classification: (Cross)validation
Validation

Classification Validity:
- **Training**: build dictionary on subset of documents with known labels
- **Test**: apply dictionary method to other documents with known labels
- Requires hand coded documents
- Hand coded documents useful for other reasons

Replicate classification exercise
- How well does our method perform on held out documents?
- Why held out?

Over fitting
- Using off-the-shelf dictionary: all labeled documents to test
- Supervised learning classification: (Cross)validation
Validation

Classification Validity:
- **Training**: build dictionary on subset of documents with known labels
- **Test**: apply dictionary method to other documents with known labels
- Requires hand coded documents
- Hand coded documents useful for other reasons
  - Is the classification scheme well defined for your texts?

Replicate classification exercise
- How well does our method perform on held out documents?
- Why held out?

Over fitting
- Using off-the-shelf dictionary: all labeled documents to test
- Supervised learning classification: (Cross)validation
Validation

Classification Validity:
- **Training**: build dictionary on subset of documents with known labels
- **Test**: apply dictionary method to other documents with known labels
- Requires hand coded documents
- Hand coded documents useful for other reasons
  - Is the classification scheme well defined for your texts?
  - Can humans accomplish the coding task?
Validation

Classification Validity:
- **Training**: build dictionary on subset of documents with known labels
- **Test**: apply dictionary method to other documents with known labels
- Requires hand coded documents
- Hand coded documents useful for other reasons
  - Is the classification scheme well defined for your texts?
  - Can humans accomplish the coding task?
  - Is the dictionary you're using appropriate?
Validation

Classification Validity:
- **Training**: build dictionary on subset of documents with known labels
- **Test**: apply dictionary method to other documents with known labels
- Requires hand coded documents
- Hand coded documents useful for other reasons
  - Is the classification scheme well defined for your texts?
  - Can humans accomplish the coding task?
  - Is the dictionary your using appropriate?

**Replicate** classification exercise
Validation

Classification Validity:
- **Training**: build dictionary on subset of documents with known labels
- **Test**: apply dictionary method to other documents with known labels
- Requires hand coded documents
- Hand coded documents useful for other reasons
  - Is the classification scheme well defined for your texts?
  - Can humans accomplish the coding task?
  - Is the dictionary your using appropriate?

**Replicate** classification exercise
- How well does our method perform on **held out** documents?
Validation

Classification Validity:
- **Training**: build dictionary on subset of documents with known labels
- **Test**: apply dictionary method to other documents with known labels
- Requires hand coded documents
- Hand coded documents useful for other reasons
  - Is the classification scheme well defined for your texts?
  - Can humans accomplish the coding task?
  - Is the dictionary your using appropriate?

*Replicate* classification exercise
- How well does our method perform on held out documents?
- Why held out?
Validation

Classification Validity:
- **Training**: build dictionary on subset of documents with known labels
- **Test**: apply dictionary method to other documents with known labels
- Requires hand coded documents
- Hand coded documents useful for other reasons
  - Is the classification scheme well defined for your texts?
  - Can humans accomplish the coding task?
  - Is the dictionary your using appropriate?

Replicate classification exercise
- How well does our method perform on held out documents?
- Why held out? **Over fitting**
Validation

Classification Validity:
- **Training**: build dictionary on subset of documents with known labels
- **Test**: apply dictionary method to other documents with known labels
- Requires hand coded documents
- Hand coded documents useful for other reasons
  - Is the classification scheme well defined for your texts?
  - Can humans accomplish the coding task?
  - Is the dictionary you're using appropriate?

Replicate classification exercise
- How well does our method perform on held out documents?
- Why held out? Over fitting
- Using off-the-shelf dictionary: all labeled documents to test
Validation

Classification Validity:
- **Training**: build dictionary on subset of documents with known labels
- **Test**: apply dictionary method to other documents with known labels
- Requires hand coded documents
- Hand coded documents useful for other reasons
  - Is the classification scheme well defined for your texts?
  - Can humans accomplish the coding task?
  - Is the dictionary your using appropriate?

**Replicate** classification exercise
- How well does our method perform on held out documents?
- Why held out? **Over fitting**
- Using off-the-shelf dictionary: all labeled documents to test
- Supervised learning classification: (Cross)validation
Hand Coding: A Brief Digression

Humans should be able to classify documents into the categories you want the machine to classify them in

- This is hard
  - Ambiguity in language
  - Limited working memory
  - Ambiguity in classification rules

A procedure for training coders:
1) Coding rules
2) Apply to new texts
3) Assess coder agreement (we’ll discuss more in a few weeks)
4) Using information and discussion, revise coding rules
Hand Coding: A Brief Digression

Humans should be able to classify documents into the categories you want the machine to classify them in

- This is hard
Hand Coding: A Brief Digression

Humans should be able to classify documents into the categories you want the machine to classify them in

- This is hard
- Why?
Hand Coding: A Brief Digression

Humans should be able to classify documents into the categories you want the machine to classify them in

- This is hard
- Why?
  - Ambiguity in language
Hand Coding: A Brief Digression

Humans should be able to classify documents into the categories you want the machine to classify them in

- This is hard
- Why?
  - Ambiguity in language
  - Limited working memory
Hand Coding: A Brief Digression

Humans should be able to classify documents into the categories you want the machine to classify them in

- This is hard
- Why?
  - Ambiguity in language
  - Limited working memory
  - Ambiguity in classification rules
Hand Coding: A Brief Digression

Humans should be able to classify documents into the categories you want the machine to classify them in

- This is hard
- Why?
  - Ambiguity in language
  - Limited working memory
  - Ambiguity in classification rules
- A procedure for training coders:
Hand Coding: A Brief Digression

Humans should be able to classify documents into the categories you want the machine to classify them in

- This is hard
- Why?
  - Ambiguity in language
  - Limited working memory
  - Ambiguity in classification rules
- A procedure for training coders:
  1) Coding rules
  2) Apply to new texts
  3) Assess coder agreement (we’ll discuss more in a few weeks)
  4) Using information and discussion, revise coding rules
## Assessing Classification

### Measures of classification performance

<table>
<thead>
<tr>
<th>Guess</th>
<th>Actual Label</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Liberal</td>
<td>True Liberal</td>
<td>False Liberal</td>
</tr>
<tr>
<td>Conservative</td>
<td>False Conservative</td>
<td>True Conservative</td>
</tr>
</tbody>
</table>

Accuracy = \( \frac{\text{True Liberal} + \text{True Conservative}}{\text{True Liberal} + \text{True Conservative} + \text{False Liberal} + \text{False Conservative}} \)

Precision (Liberal) = \( \frac{\text{True Liberal}}{\text{True Liberal} + \text{False Liberal}} \)

Recall (Liberal) = \( \frac{\text{True Liberal}}{\text{True Liberal} + \text{False Conservative}} \)

F1 (Liberal) = \( \frac{2 \times \text{Precision (Liberal)} \times \text{Recall (Liberal)}}{\text{Precision (Liberal)} + \text{Recall (Liberal)}} \)

Under reported for dictionary classification
Assessing Classification

Measures of classification performance

<table>
<thead>
<tr>
<th>Guess</th>
<th>Actual Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>Liberal</td>
<td>True Liberal</td>
</tr>
<tr>
<td></td>
<td>False Liberal</td>
</tr>
<tr>
<td>Conservative</td>
<td>True Conservative</td>
</tr>
<tr>
<td></td>
<td>False Conservative</td>
</tr>
</tbody>
</table>

Accuracy = \[
\frac{TrueLib + TrueCons}{TrueLib + TrueCons + FalseLib + FalseCons}\]
Assessing Classification

Measures of classification performance

<table>
<thead>
<tr>
<th>Guess</th>
<th>Actual Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>Liberal</td>
<td>True Liberal</td>
</tr>
<tr>
<td></td>
<td>False Liberal</td>
</tr>
<tr>
<td>Conservative</td>
<td>True Conservative</td>
</tr>
</tbody>
</table>

\[
\text{Accuracy} = \frac{\text{TrueLib} + \text{TrueCons}}{\text{TrueLib} + \text{TrueCons} + \text{FalseLib} + \text{FalseCons}}
\]

\[
\text{Precision}_{\text{Liberal}} = \frac{\text{True Liberal}}{\text{True Liberal} + \text{False Liberal}}
\]
### Assessing Classification

#### Measures of classification performance

<table>
<thead>
<tr>
<th>Guess</th>
<th>Actual Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>Liberal</td>
<td>True Liberal</td>
</tr>
<tr>
<td>Conservative</td>
<td>False Conservative</td>
</tr>
</tbody>
</table>

- **Accuracy**
  
  \[
  \text{Accuracy} = \frac{\text{TrueLib} + \text{TrueCons}}{\text{TrueLib} + \text{TrueCons} + \text{FalseLib} + \text{FalseCons}}
  \]

- **Precision\textsubscript{Liberal}**
  
  \[
  \text{Precision\textsubscript{Liberal}} = \frac{\text{True Liberal}}{\text{True Liberal} + \text{False Liberal}}
  \]

- **Recall\textsubscript{Liberal}**
  
  \[
  \text{Recall\textsubscript{Liberal}} = \frac{\text{True Liberal}}{\text{True Liberal} + \text{False Conservative}}
  \]
Assessing Classification

Measures of classification performance

<table>
<thead>
<tr>
<th>Guess</th>
<th>Actual Label</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Liberal</td>
<td>Conservative</td>
</tr>
<tr>
<td>Liberal</td>
<td>True Liberal</td>
<td>False Liberal</td>
</tr>
<tr>
<td>Conservative</td>
<td>False Conservative</td>
<td>True Conservative</td>
</tr>
</tbody>
</table>

Accuracy = \frac{TrueLib + TrueCons}{TrueLib + TrueCons + FalseLib + FalseCons}  

Precision\text{Liberal} = \frac{TrueLib}{TrueLib + FalseLib} 

Recall\text{Liberal} = \frac{TrueLib}{TrueLib + FalseCons} 

F\text{Liberal} = \frac{2 \times \text{Precision}\text{Liberal} \times \text{Recall}\text{Liberal}}{\text{Precision}\text{Liberal} + \text{Recall}\text{Liberal}}
## Assessing Classification

### Measures of classification performance

<table>
<thead>
<tr>
<th>Guess</th>
<th>Actual Label</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Liberal</td>
<td>True Liberal</td>
<td>False Liberal</td>
<td></td>
</tr>
<tr>
<td>Conservative</td>
<td>False Conservative</td>
<td>True Conservative</td>
<td></td>
</tr>
</tbody>
</table>

### Formulas

**Accuracy**

\[
\text{Accuracy} = \frac{\text{TrueLib} + \text{TrueCons}}{\text{TrueLib} + \text{TrueCons} + \text{FalseLib} + \text{FalseCons}}
\]

**Precision\text{Liberal}**

\[
\text{Precision}_{\text{Liberal}} = \frac{\text{True Liberal}}{\text{True Liberal} + \text{False Liberal}}
\]

**Recall\text{Liberal}**

\[
\text{Recall}_{\text{Liberal}} = \frac{\text{True Liberal}}{\text{True Liberal} + \text{False Conservative}}
\]

**F\text{Liberal}**

\[
F_{\text{Liberal}} = \frac{2 \text{Precision}_{\text{Liberal}} \cdot \text{Recall}_{\text{Liberal}}}{\text{Precision}_{\text{Liberal}} + \text{Recall}_{\text{Liberal}}}
\]

*Under reported for dictionary classification*
What about continuous measures?

- Necessarily more complicated
- Go back to hand coding exercise
- Imagine asking undergraduates to rate a document on a continuous scale (0-100)
- Difficult to create classifications with agreement
- Precisely the point
  - merly creating a gold standard is hard, let alone computer classification

- Lower level classification
  - label phrases and then aggregate

- Modifiable areal unit problem in texts
  - aggregating destroys information, conclusion may depend on level of aggregation
What about continuous measures?

Necessarily more complicated
What about continuous measures?

Necessarily more complicated

- Go back to hand coding exercise
What about continuous measures?

Necessarily more complicated

- Go back to hand coding exercise
- Imagine asking undergraduates to rate document on a continuous scale (0-100)

Lower level classification

⇝

Modifiable areal unit problem in texts

⇝

aggregating destroys information, conclusion may depend on level of aggregation
What about continuous measures?

Necessarily more complicated

- Go back to hand coding exercise
- Imagine asking undergraduates to rate document on a continuous scale (0-100)
- Difficult to create classifications with agreement
What about continuous measures?

Necessarily more complicated

- Go back to hand coding exercise
- Imagine asking undergraduates to rate document on a continuous scale (0-100)
- **Difficult** to create classifications with agreement
- **Precisely** the point⇒ merely creating a gold standard is hard, let alone computer classification
What about continuous measures?

Necessarily more complicated

- Go back to hand coding exercise
- Imagine asking undergraduates to rate document on a continuous scale (0-100)
- Difficult to create classifications with agreement
- Precisely the point\rightarrow merely creating a gold standard is hard, let alone computer classification

Lower level classification
What about continuous measures?

Necessarily more complicated

- Go back to hand coding exercise
- Imagine asking undergraduates to rate document on a continuous scale (0-100)
- Difficult to create classifications with agreement
- Precisely the point: merely creating a gold standard is hard, let alone computer classification

Lower level classification: label phrases and then aggregate
What about continuous measures?

Necessarily more complicated

- Go back to hand coding exercise
- Imagine asking undergraduates to rate document on a continuous scale (0-100)
- **Difficult** to create classifications with agreement
- **Precisely** the point → merely creating a gold standard is hard, let alone computer classification

Lower level classification → label phrases and then aggregate
Modifiable areal unit problem in texts
What about continuous measures?

Necessarily more complicated

- Go back to hand coding exercise
- Imagine asking undergraduates to rate document on a continuous scale (0-100)
- **Difficult** to create classifications with agreement
- **Precisely** the point \(\Rightarrow\) merely creating a gold standard is hard, let alone computer classification

**Lower level classification** \(\Rightarrow\) label phrases and then aggregate

Modifiable areal unit problem in texts \(\Rightarrow\) aggregating destroys information, conclusion may depend on level of aggregation
Validation, Dictionaries from other Fields

Accounting Research: measure tone of 10-K reports

Previous state of art: Harvard-IV-4 Dictionary applied to texts

Loughran and McDonald (2011): Financial Documents are Different, polysemes

- Negative words in Harvard, Not Negative in Accounting:
  - tax, cost, capital, board, liability, foreign, cancer, crude (oil), tire

- 73% of Harvard negative words in this set (!!!!!)

- Not Negative Harvard, Negative in Accounting:
  - felony, litigation, restated, misstatement, and unanticipated

Context Matters
Validation, Dictionaries from other Fields

Accounting Research: measure tone of 10-K reports
Validation, Dictionaries from other Fields

Accounting Research: measure tone of 10-K reports
   - tone matters (\$)

Previous state of art: Harvard-IV-4 Dictionary applied to texts
Loughran and McDonald (2011): Financial Documents are Different, polysemes
   - Negative words in Harvard, Not Negative in Accounting:
      - tax, cost, capital, board, liability, foreign, cancer, crude (oil), tire
   - 73% of Harvard negative words in this set (!!!!!)
   - Not Negative Harvard, Negative in Accounting:
      - felony, litigation, restated, misstatement, and unanticipated

Context Matters
Validation, Dictionaries from other Fields

Accounting Research: measure tone of 10-K reports
  - tone matters ($)

Previous state of art: Harvard-IV-4 Dictionary applied to texts
  - Negative words in Harvard, Not Negative in Accounting:
    - tax, cost, capital, board, liability, foreign, cancer, crude(oil), tire
  - 73% of Harvard negative words in this set

  - Not Negative Harvard, Negative in Accounting:
    - felony, litigation, restated, misstatement, and unanticipated

Context Matters
Validation, Dictionaries from other Fields

Accounting Research: measure tone of 10-K reports

- tone matters ($)

Previous state of art: Harvard-IV-4 Dictionary applied to texts

Loughran and McDonald (2011): Financial Documents are Different, polysemes
Validation, Dictionaries from other Fields

Accounting Research: measure **tone** of 10-K reports
  - **tone** matters ($)

Previous state of art: Harvard-IV-4 Dictionary applied to texts
Loughran and McDonald (2011): Financial Documents are Different, polysemes
  - Negative words in Harvard, Not Negative in Accounting:
Validation, Dictionaries from other Fields

Accounting Research: measure tone of 10-K reports
  - tone matters ($)

Previous state of art: Harvard-IV-4 Dictionary applied to texts
Loughran and McDonald (2011): Financial Documents are Different, polysemes
  - Negative words in Harvard, Not Negative in Accounting:
    tax, cost, capital, board, liability, foreign, cancer, crude(oil), tire
Accounting Research: measure tone of 10-K reports
  - tone matters ($) 
Previous state of art: Harvard-IV-4 Dictionary applied to texts
Loughran and McDonald (2011): Financial Documents are Different, polysemes
  - Negative words in Harvard, Not Negative in Accounting: tax, cost, capital, board, liability, foreign, cancer, crude(oil), tire 
  - 73% of Harvard negative words in this set(!!!!!!)
Validation, Dictionaries from other Fields

Accounting Research: measure tone of 10-K reports
  - tone matters ($)

Previous state of art: Harvard-IV-4 Dictionary applied to texts
Loughran and McDonald (2011): Financial Documents are Different, polysemses
  - Negative words in Harvard, Not Negative in Accounting:
    tax, cost, capital, board, liability, foreign, cancer, crude(oil), tire
  - 73% of Harvard negative words in this set(!!!!!)
  - Not Negative Harvard, Negative in Accounting:
Validation, Dictionaries from other Fields

Accounting Research: measure tone of 10-K reports
- tone matters ($)

Previous state of art: Harvard-IV-4 Dictionary applied to texts
Loughran and McDonald (2011): Financial Documents are Different, polysemes
- Negative words in Harvard, Not Negative in Accounting:
tax, cost, capital, board, liability, foreign, cancer, crude (oil), tire
- 73% of Harvard negative words in this set(!!!!!)
- Not Negative Harvard, Negative in Accounting:
  felony, litigation, restated, misstatement, and unanticipated
Measuring Happiness

- Quantifying Happiness: How happy is society?
- How Happy is a Song?
- Blog posts?
- Facebook posts? (Gross National Happiness)

Use Dictionary Methods

Justin Grimmer (Stanford University)

Text as Data

October 2nd, 2014 20 / 23
Measuring Happiness

- Quantifying Happiness: How happy is society?

- How Happy is a Song?
- Blog posts?
- Facebook posts? (Gross National Happiness)

Use Dictionary Methods

Justin Grimmer (Stanford University)
Measuring Happiness

- Quantifying Happiness: How happy is society?
- How Happy is a Song?
Measuring Happiness

- Quantifying Happiness: How happy is society?
- How Happy is a Song?
- Blog posts?
Measuring Happiness

- Quantifying Happiness: How happy is society?
- How Happy is a Song?
- Blog posts?
- Facebook posts? (Gross National Happiness)
Measuring Happiness

- Quantifying Happiness: How happy is society?
- How Happy is a Song?
- Blog posts?
- Facebook posts? (Gross National Happiness)

Use Dictionary Methods
Measuring Happiness

Dodds and Danforth (2009): Use a dictionary method to measure happiness

- Affective Norms for English Words (ANEW)
- Bradley and Lang 1999: 1034 words, Affective reaction to words
- On a scale of 1-9 how happy does this word make you?
  - Happy: triumphant (8.82)/paradise (8.72)/ love (8.72)
  - Neutral: street (5.22)/ paper (5.20)/ engine (5.20)
  - Unhappy: cancer (1.5)/funeral (1.39)/ rape (1.25) /suicide (1.25)

Happiness for text

\[
\text{Happiness}_i = \frac{\sum_{k=1}^{K} \theta_k X_{ik}}{\sum_{k=1}^{K} X_{ik}}
\]

Justin Grimmer (Stanford University)
Measuring Happiness

Dodds and Danforth (2009): Use a dictionary method to measure happiness
  - Affective Norms for English Words (ANEW)

Happiness \( i \) (with word \( j \) having happiness \( \theta_j \) and document \( X_{ij} \))

\[
Happiness_i = \frac{\sum_{k=1}^{K} \theta_k X_{ik}}{\sum_{k=1}^{K} X_{ik}}
\]
Measuring Happiness

Dodds and Danforth (2009): Use a dictionary method to measure happiness
  - Affective Norms for English Words (ANEW)
  - Bradley and Lang 1999: 1034 words, Affective reaction to words

\[
\text{Happiness}_i = \sum_{k=1}^{K} \theta_k X_{ik} / \sum_{k=1}^{K} X_{ik}
\]
Measuring Happiness

Dodds and Danforth (2009): Use a dictionary method to measure happiness

- Affective Norms for English Words (ANEW)
- Bradley and Lang 1999: 1034 words, Affective reaction to words
  - On a scale of 1-9 how happy does this word make you?
Measuring Happiness

Dodds and Danforth (2009): Use a dictionary method to measure happiness

- Affective Norms for English Words (ANEW)
- Bradley and Lang 1999: 1034 words, Affective reaction to words
  - On a scale of 1-9 how happy does this word make you?
    Happy: triumphant (8.82)/paradise (8.72)/ love (8.72)
    Neutral: street (5.22)/ paper (5.20)/ engine (5.20)
    Unhappy: cancer (1.5)/funeral (1.39)/ rape (1.25) /suicide (1.25)
Measuring Happiness

Dodds and Danforth (2009): Use a dictionary method to measure happiness

- **Affective Norms for English Words (ANEW)**
- Bradley and Lang 1999: 1034 words, Affective reaction to words
  - On a scale of 1-9 how happy does this word make you?
    - **Happy**: triumphant (8.82)/paradise (8.72)/ love (8.72)
    - **Neutral**: street (5.22)/ paper (5.20)/ engine (5.20)
  - **Unhappy**: cancer (1.5)/funeral (1.39)/ rape (1.25)/ suicide (1.25)
Measuring Happiness

Dodds and Danforth (2009): Use a dictionary method to measure happiness

- Affective Norms for English Words (ANEW)
- Bradley and Lang 1999: 1034 words, Affective reaction to words
  - On a scale of 1-9 how happy does this word make you?
    Happy: triumphant (8.82)/paradise (8.72)/ love (8.72)
    Neutral: street (5.22)/ paper (5.20)/ engine (5.20)
    Unhappy: cancer (1.5)/funeral (1.39)/ rape (1.25) /suicide (1.25)
Measuring Happiness

Dodds and Danforth (2009): Use a dictionary method to measure happiness

- Affective Norms for English Words (ANEW)
- Bradley and Lang 1999: 1034 words, Affective reaction to words
  - On a scale of 1-9 how happy does this word make you?
    - Happy: triumphant (8.82)/paradise (8.72)/ love (8.72)
    - Neutral: street (5.22)/ paper (5.20)/ engine (5.20)
    - Unhappy: cancer (1.5)/funeral (1.39)/ rape (1.25) /suicide (1.25)

- HAPPINESS for text \( i \) (with word \( j \) having happiness \( \theta_j \) and document frequency \( X_{ij} \))
Measuring Happiness

Dodds and Danforth (2009): Use a dictionary method to measure happiness

- **Affective Norms for English Words (ANEW)**
- Bradley and Lang 1999: 1034 words, Affective reaction to words
  - On a scale of 1-9 how happy does this word make you?
    - **Happy**: triumphant (8.82)/ paradise (8.72)/ love (8.72)
    - **Neutral**: street (5.22)/ paper (5.20)/ engine (5.20)
    - **Unhappy**: cancer (1.5)/ funeral (1.39)/ rape (1.25)/ suicide (1.25)

- **Happiness** for text $i$ (with word $j$ having happiness $\theta_j$ and document frequency $X_{ij}$)

$$\text{Happiness}_i = \frac{\sum_{k=1}^{K} \theta_k X_{ik}}{\sum_{k=1}^{K} X_{ik}}$$
Homework Hints:
One approach: write a for loop searching for words in dictionary (caution: is dictionary stemmed?)

Happiest Song on Thriller?
P.Y.T. (Pretty Young Thing) (This is the right answer!)

Justin Grimmer (Stanford University)

Lyrics for Michael Jackson’s Billie Jean

“She was more like a beauty queen from a movie scene."

And mother always told me, be careful who you love.

And be careful of what you do ’cause the lie becomes the truth.

Billie Jean is not my lover, She’s just a girl who claims that I am the one.

<table>
<thead>
<tr>
<th>ANEW words</th>
<th>$v_k$</th>
<th>$f_k$</th>
</tr>
</thead>
<tbody>
<tr>
<td>k=1. love</td>
<td>8.72</td>
<td>1</td>
</tr>
<tr>
<td>2. mother</td>
<td>8.39</td>
<td>1</td>
</tr>
<tr>
<td>3. baby</td>
<td>8.22</td>
<td>3</td>
</tr>
<tr>
<td>4. beauty</td>
<td>7.82</td>
<td>1</td>
</tr>
<tr>
<td>5. truth</td>
<td>7.80</td>
<td>1</td>
</tr>
<tr>
<td>6. people</td>
<td>7.33</td>
<td>2</td>
</tr>
<tr>
<td>7. strong</td>
<td>7.11</td>
<td>2</td>
</tr>
<tr>
<td>8. young</td>
<td>6.89</td>
<td>1</td>
</tr>
<tr>
<td>9. girl</td>
<td>6.87</td>
<td>4</td>
</tr>
<tr>
<td>10. movie</td>
<td>6.86</td>
<td>1</td>
</tr>
<tr>
<td>11. perfume</td>
<td>6.76</td>
<td>1</td>
</tr>
<tr>
<td>12. queen</td>
<td>6.44</td>
<td>1</td>
</tr>
<tr>
<td>13. name</td>
<td>5.55</td>
<td>1</td>
</tr>
<tr>
<td>14. lie</td>
<td>2.79</td>
<td>1</td>
</tr>
</tbody>
</table>

$$v_{text} = \frac{\sum v_k f_k}{\sum f_k}$$

$\Rightarrow v_{Billie Jean} = 7.1$

$\Rightarrow v_{Thriller} = 6.3$

$\Rightarrow v_{Michael Jackson} = 6.4$
Homework Hints: One approach: write a for loop searching for words in dictionary (caution: is dictionary stemmed?)
Homework Hints: One approach: write a for loop searching for words in dictionary (caution: is dictionary stemmed?)

Happiest Song on Thriller?
Homework Hints: One approach: write a for loop searching for words in dictionary (caution: is dictionary stemmed?)

Happiest Song on Thriller?

P.Y.T. (Pretty Young Thing) (This is the right answer!)
Happiness in Society

![Graph showing the trend of happiness in society over the years from 1960 to 2000. The graph plots 'valence (v)' on the y-axis and 'year' on the x-axis. The trend shows a general decline in happiness over time.](image)
Happiness in Society

![Graph showing the change in valence (v) for different genres over time.]

- Gospel/Soul (6.91)
- Pop (6.69)
- Reggae (6.40)
- Rock (6.27)
- Rap/Hip-Hop (6.01)
- Punk (5.61)
- Metal/Industrial (5.10)
Dictionary Methods

Today: Classification via Dictionaries
Next week: Separating Words and the Geometry of Text
Good luck on the homework!