Text as Data

Justin Grimmer

Professor
Department of Political Science
Stanford University

May 24th, 2019

Discovery and Measurement

What is the research process? (Grimmer, Roberts, and Stewart 2019)

1) Discovery: a hypothesis or view of the world
2) Measurement according to some organization
3) Causal Inference: effect of some intervention

Text as data methods assist at each stage of research process

Measurement

Types of Classification Problems

Topic: What is this text about?

Types of Classification Problems

Topic: What is this text about?

- Policy area of legislation
  $\Rightarrow$ {Agriculture, Crime, Environment, ...}
- Campaign agendas
  $\Rightarrow$ {Abortion, Campaign, Finance, Taxing, ... }

Types of Classification Problems

Topic: What is this text about?

- Policy area of legislation
  $\Rightarrow$ {Agriculture, Crime, Environment, ...}
- Campaign agendas
  $\Rightarrow$ {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
  $\Rightarrow$ {Agriculture, Crime, Environment, ...}
- Campaign agendas
  $\Rightarrow$ {Abortion, Campaign, Finance, Taxing, ... }

Sentiment: What is said in this text? [Public Opinion]

- Positions on legislation
  $\Rightarrow$ { Support, Ambiguous, Oppose }
- Positions on Court Cases
  $\Rightarrow$ { Agree with Court, Disagree with Court }
- Liberal/Conservative Blog Posts
  $\Rightarrow$ { 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

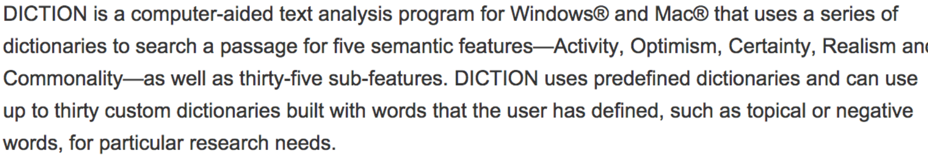
  ⇒ { Partisan Taunt, Intra party taunt, Agency taunt, ... }
- Negative campaigning

  ⇒ { Negative ad, Positive ad}

Pre-existing word weights$\rightsquigarrow$ Dictionaries

Pre-existing word weights$\rightsquigarrow$ Dictionaries

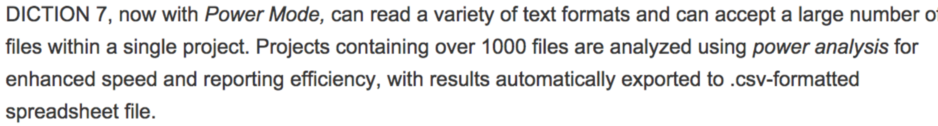
`DICTION`



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 an 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 $\rightsquigarrow$ Dictionaries

`DICTION`



DICTION 7, now with *Power Mode,* can read a variety of text formats and can accept a large number o 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.

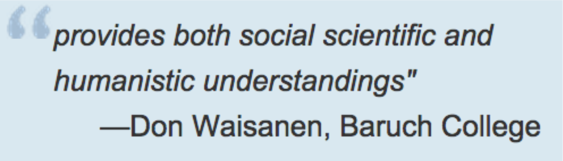
Pre-existing word weights⇝ Dictionaries

`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$\rightsquigarrow$ Dictionaries

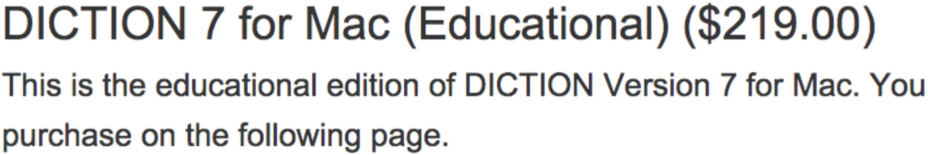
`DICTION`



> *provides both social scientific and humanistic understandings"*
>
> —Don Waisanen, Baruch College

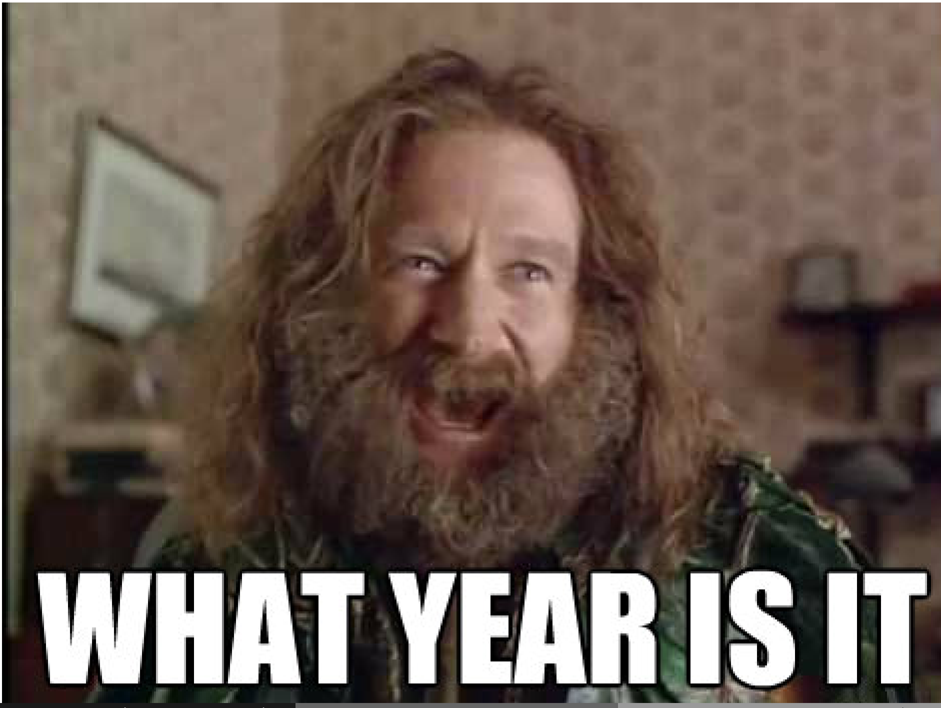
Pre-existing word weights⇝ Dictionaries

`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$\rightsquigarrow$ wrapped in GUI

Dictionary Methods

Many Dictionary Methods (like DICTION)

1) Proprietary$\rightsquigarrow$ wrapped in GUI
2) Basic tasks:

Dictionary Methods

Many Dictionary Methods (like DICTION)

1) Proprietary $\rightsquigarrow$ wrapped in GUI
2) Basic tasks:
    a) Count words

Dictionary Methods

Many Dictionary Methods (like DICTION)

1) Proprietary$\rightsquigarrow$ wrapped in GUI
2) Basic tasks:
    a) Count words
    b) Weighted counts of words

Dictionary Methods

Many Dictionary Methods (like DICTION)

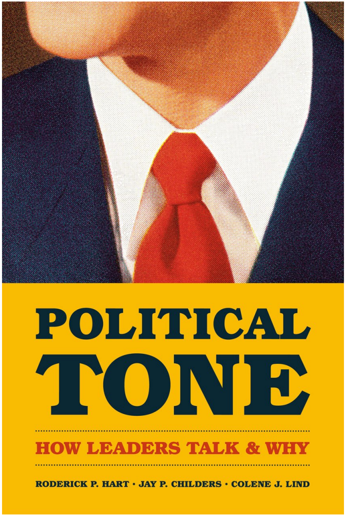
1) Proprietary⇝ wrapped in GUI
2) Basic tasks:
    a) Count words
    b) Weighted counts of words
    c) Some graphics

Dictionary Methods

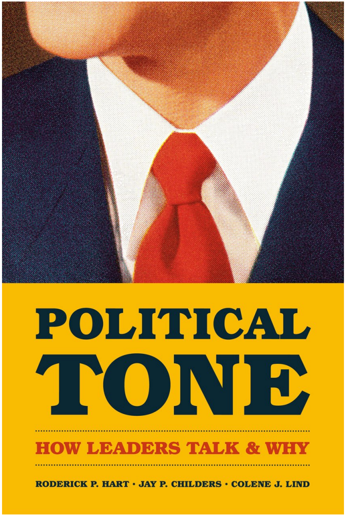
Many Dictionary Methods (like DICTION)

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

DICTION

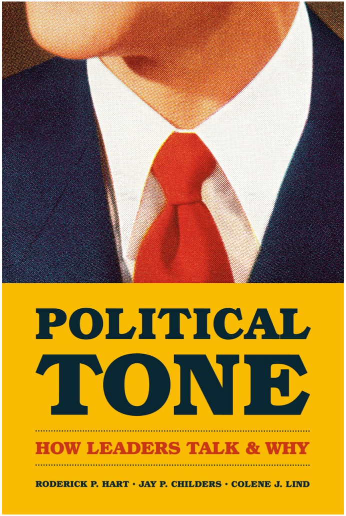

POLITICAL
TONE
HOW LEADERS TALK & WHY
RODERICK P. HART · JAY P. CHILDERS · COLENE J. LIND

DICTION



- { Certain, Uncertain }

DICTION



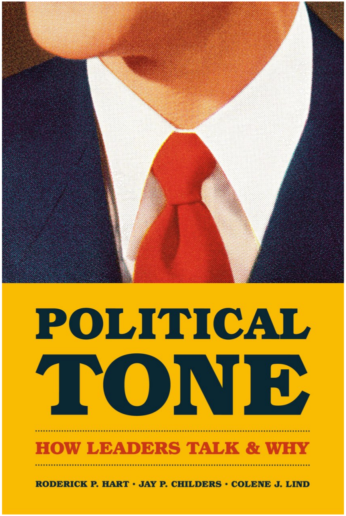
POLITICAL
TONE

HOW LEADERS TALK & WHY

RODERICK P. HART · JAY P. CHILDERS · COLENE J. LIND

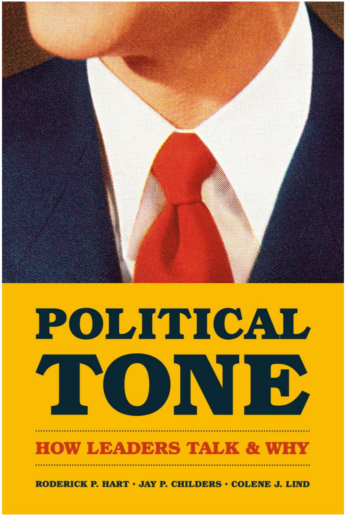
- { Certain, Uncertain }
  , { Optimistic, Pessimistic }

DICTION



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

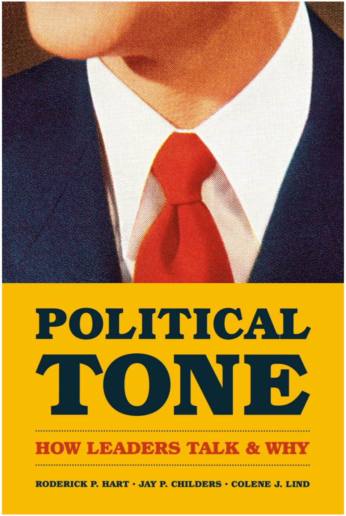
DICTION



- { Certain, Uncertain } , { Optimistic, Pessimistic }
- ≈ 10,000 words

Applies DICTION to a wide array of political texts

DICTION



- { Certain, Uncertain }
  , { Optimistic, Pessimistic }
- $\approx$ 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/`)

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*

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 }

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

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

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)

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:

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$\rightsquigarrow$ " 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/`)
    - 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] 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/`)
    - 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] 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/`)
    - 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] 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/`)
    - 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] 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/`)
    - 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] 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/`)
    - 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] 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)
- ...

Generating New Words

Three ways to create dictionaries (non-exhaustive):

Generating New Words

Three ways to create dictionaries (non-exhaustive):

- Statistical methods (Separating methods)

Generating New Words

Three ways to create dictionaries (non-exhaustive):

- Statistical methods (Separating methods)
- Manual generation

Generating New Words

Three ways to create dictionaries (non-exhaustive):

- Statistical methods (Separating methods)
- Manual generation
  - Careful thought (prayer? epiphanies? divine intervention?) about useful words

Generating New Words

Three ways to create dictionaries (non-exhaustive):

- Statistical methods (Separating 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 (Separating 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 (Separating 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

Generating New Words

Three ways to create dictionaries (non-exhaustive):

- Statistical methods (Separating 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
    - Example: { Happy, Unhappy }

Generating New Words

Three ways to create dictionaries (non-exhaustive):

- Statistical methods (Separating 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

Generating New Words

Three ways to create dictionaries (non-exhaustive):

- Statistical methods (Separating 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`

Generating New Words

Three ways to create dictionaries (non-exhaustive):

- Statistical methods (Separating 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
        - Example: { Happy, Unhappy }
        - Ask turkers: how happy is
          `elevator`, `car`, `pretty`, `young`
          Output as dictionary

Applying Methods to Documents

Applying the model:

Applying Methods to Documents

Applying the model:

- Vector of word counts: $\boldsymbol{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iK})$, $(i = 1, \ldots, N)$

Applying Methods to Documents

Applying the model:

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

Applying Methods to Documents

Applying the model:

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

Applying Methods to Documents

Applying the model:

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

Applying Methods to Documents

Applying the model:

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

Applying Methods to Documents

Applying the model:

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

For each document *i* calculate score for document

Applying Methods to Documents

Applying the model:

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

For each document *i* calculate score for document

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

Applying Methods to Documents

Applying the model:

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

For each document *i* calculate score for document

$$\begin{aligned} Y_i &= \frac{\sum_{k=1}^{K} \theta_k X_{ik}}{\sum_{k=1}^{K} X_k} \\ Y_i &= \frac{\boldsymbol{\theta}^{'} \boldsymbol{X}_i}{\boldsymbol{X}_i^{'} \mathbf{1}} \end{aligned}$$

Applying Methods to Documents

Applying the model:

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

For each document *i* calculate score for document

$$\begin{aligned} Y_i &= \frac{\sum_{k=1}^{K} \theta_k X_{ik}}{\sum_{k=1}^{K} X_k} \\ Y_i &= \frac{\boldsymbol{\theta}^{'} \boldsymbol{X}_i}{\boldsymbol{X}_i^{'} \mathbf{1}} \end{aligned}$$

$Y_i \approx$ continuous $\rightsquigarrow$ Classification

Applying Methods to Documents

Applying the model:

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

For each document *i* calculate score for document

$$\begin{aligned} Y_i &= \frac{\sum_{k=1}^{K} \theta_k X_{ik}}{\sum_{k=1}^{K} X_k} \\ Y_i &= \frac{\boldsymbol{\theta}^{'} \boldsymbol{X}_i}{\boldsymbol{X}_i^{'} \mathbf{1}} \end{aligned}$$

$Y_i \approx$ continuous $\rightsquigarrow$ Classification

$Y_i > 0 \Rightarrow$ Positive Category

Applying Methods to Documents

Applying the model:

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

For each document i calculate score for document

$$\begin{aligned} Y_i &= \frac{\sum_{k=1}^{K} \theta_k X_{ik}}{\sum_{k=1}^{K} X_k} \\ Y_i &= \frac{\boldsymbol{\theta}^{'} \boldsymbol{X}_i}{\boldsymbol{X}_i^{'} \mathbf{1}} \end{aligned}$$

$Y_i \approx$ continuous $\rightsquigarrow$ Classification

$Y_i > 0 \Rightarrow$ Positive Category

$Y_i < 0 \Rightarrow$ Negative Category

Applying Methods to Documents

Applying the model:

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

For each document *i* calculate score for document

$$\begin{aligned} Y_i &= \frac{\sum_{k=1}^{K} \theta_k X_{ik}}{\sum_{k=1}^{K} X_k} \\ Y_i &= \frac{\boldsymbol{\theta}^{'} \boldsymbol{X}_i}{\boldsymbol{X}_i^{'} \mathbf{1}} \end{aligned}$$

$Y_i \approx$ continuous $\rightsquigarrow$ Classification

$Y_i > 0 \Rightarrow$ Positive Category

$Y_i < 0 \Rightarrow$ Negative Category

$Y_i \approx 0$ Ambiguous

Applying a Dictionary to Press Releases

Applying a Dictionary to Press Releases

- Collection of 169,779 press releases (US House members 2005-2010)

Applying a Dictionary to Press Releases

- Collection of 169,779 press releases (US House members 2005-2010)
- Dictionary from Neal Caren's website $\rightsquigarrow$ Theresa Wilson, Janyce Wiebe, and Paul Hoffman's dictionary

Applying a Dictionary to Press Releases

- Collection of 169,779 press releases (US House members 2005-2010)
- Dictionary from Neal Caren's website ⇝ Theresa Wilson, Janyce Wiebe, and Paul Hoffman's dictionary
- Create positive/negative score for press releases.

Applying a Dictionary to Press Releases

- Collection of 169,779 press releases (US House members 2005-2010)
- Dictionary from Neal Caren's website $\rightsquigarrow$ 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

Examining Positive and Negative Statements in Press Releases

Least positive members of Congress:

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

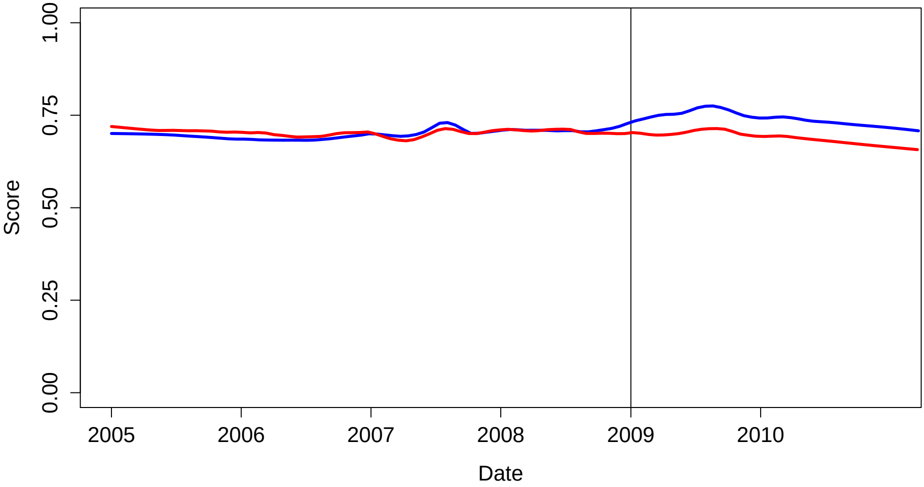
Examining Positive and Negative Statements in Press Releases

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 $\rightsquigarrow$ less positive in press releases

Examining Positive and Negative Statements in Press Releases



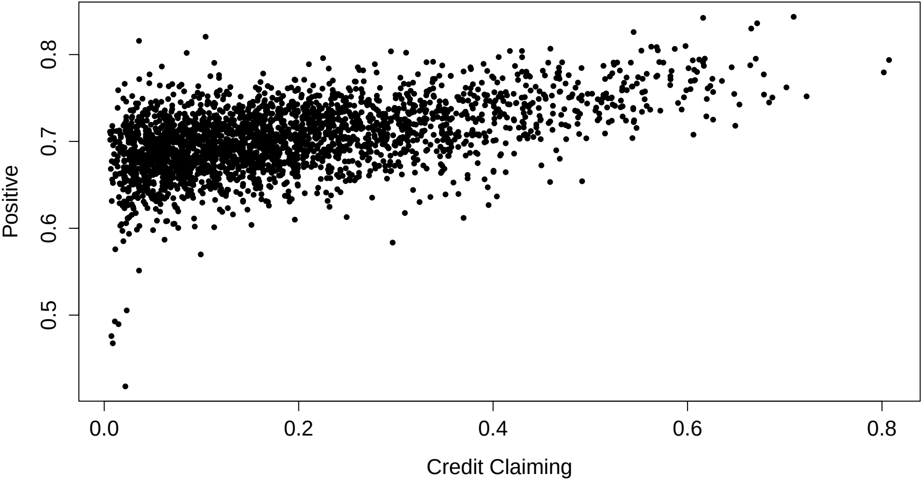
Examining Positive and Negative Statements in Press Releases

- Credit Claiming press release: 9.1 percentage points “more positive” than a non-credit claiming press release

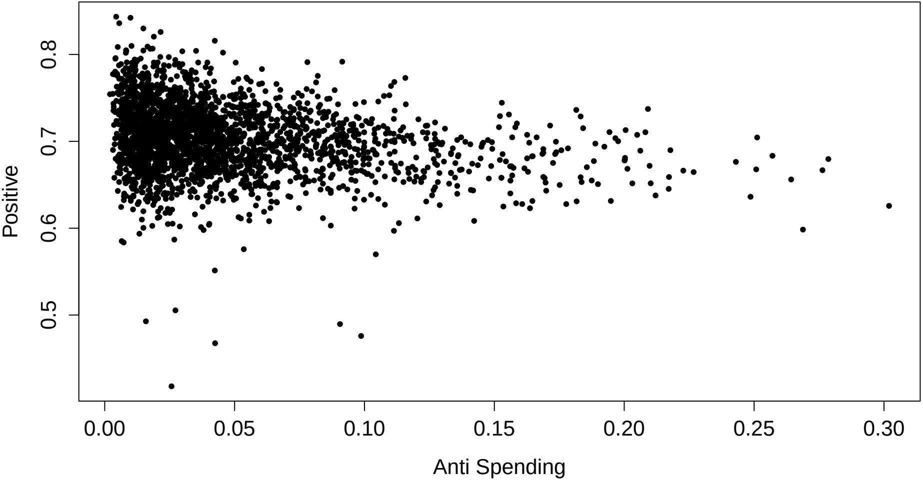
Examining Positive and Negative Statements in Press Releases

- 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

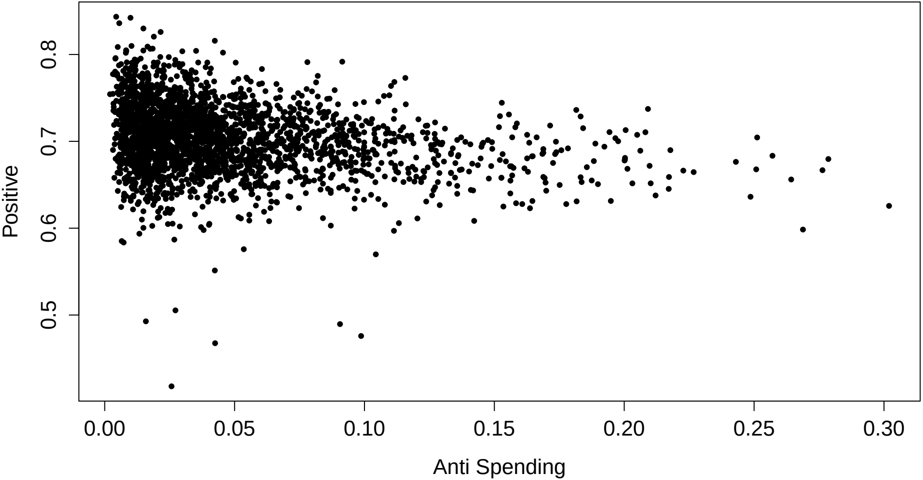
Examining Positive and Negative Statements in Press Releases



Examining Positive and Negative Statements in Press Releases



Examining Positive and Negative Statements in Press Releases



Methodological Issues/Problems with Dictionaries

Dictionary methods are context invariant

Methodological Issues/Problems with Dictionaries

Dictionary methods are context invariant

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

Methodological Issues/Problems with Dictionaries

Dictionary methods are context invariant

- No optimization step $\rightsquigarrow$ same word weights regardless of texts
- Optimization$\rightsquigarrow$ incorporate information specific to context

Methodological Issues/Problems with Dictionaries

Dictionary methods are context invariant

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

Methodological Issues/Problems with Dictionaries

Dictionary methods are context invariant

- No optimization step $\rightsquigarrow$ same word weights regardless of texts
- Optimization$\rightsquigarrow$ incorporate information specific to context
- Without optimization$\rightsquigarrow$ 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 ⇝ same word weights regardless of texts
- Optimization⇝ incorporate information specific to context
- Without optimization⇝ 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:

Validation

Classification Validity:

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

Validation

Classification Validity:

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

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

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

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?

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 your 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 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

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

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

	Actual Label	
Guess	Liberal	Conservative
Liberal	True Liberal	False Liberal
Conservative	False Conservative	True Conservative

Assessing Classification

Measures of classification performance

	Actual Label	
Guess	Liberal	Conservative
Liberal	True Liberal	False Liberal
Conservative	False Conservative	True Conservative

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

Assessing Classification

Measures of classification performance

	Actual Label	
Guess	Liberal	Conservative
Liberal	True Liberal	False Liberal
Conservative	False Conservative	True Conservative

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

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

Assessing Classification

Measures of classification performance

	Actual Label	
Guess	Liberal	Conservative
Liberal	True Liberal	False Liberal
Conservative	False Conservative	True Conservative

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

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

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

Assessing Classification

Measures of classification performance

	Actual Label	
Guess	Liberal	Conservative
Liberal	True Liberal	False Liberal
Conservative	False Conservative	True Conservative

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

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

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

$$F_{\text{Liberal}} = \frac{2\text{Precision}_{\text{Liberal}}\text{Recall}_{\text{Liberal}}}{\text{Precision}_{\text{Liberal}} + \text{Recall}_{\text{Liberal}}}$$

Assessing Classification

Measures of classification performance

	Actual Label	
Guess	Liberal	Conservative
Liberal	True Liberal	False Liberal
Conservative	False Conservative	True Conservative

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

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

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

$$F_{\text{Liberal}} = \frac{2\text{Precision}_{\text{Liberal}}\text{Recall}_{\text{Liberal}}}{\text{Precision}_{\text{Liberal}} + \text{Recall}_{\text{Liberal}}}$$

Under reported for dictionary classification

What about continuous measures?

$\rightsquigarrow$

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)

$\leadsto$

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 $\rightsquigarrow$ merely creating a gold standard is hard, let alone computer classification

$\rightsquigarrow$

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 $\rightsquigarrow$ merely creating a gold standard is hard, let alone computer classification

Lower level classification

$\rightsquigarrow$

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 $\rightsquigarrow$ merely creating a gold standard is hard, let alone computer classification

Lower level classification $\rightsquigarrow$ label phrases and then aggregate
Modifiable areal unit problem in texts $\rightsquigarrow$

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 ⇝ aggregating destroys information, conclusion may depend on level of aggregation

Validation, Dictionaries from other Fields

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 ($)

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

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`

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(!!!!!)

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:

Validation, Dictionaries from other Fields

Accounting Research: measure tone of 10-K reports

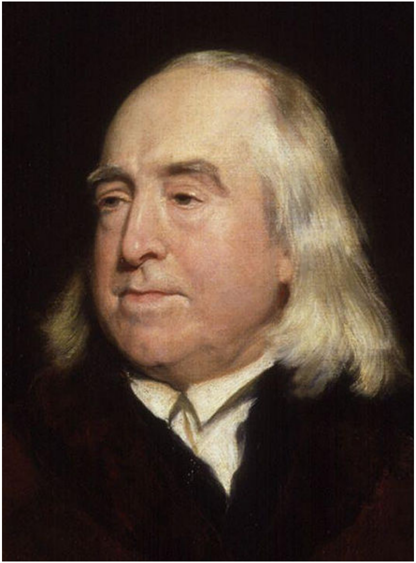
- tone matters ($)

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

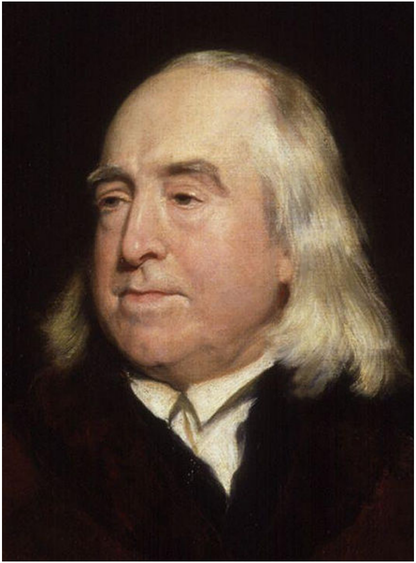
Loughran and McDonald (2011): Financial Documents are Different, polysemes

- 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, unanticipated`

Measuring Happiness

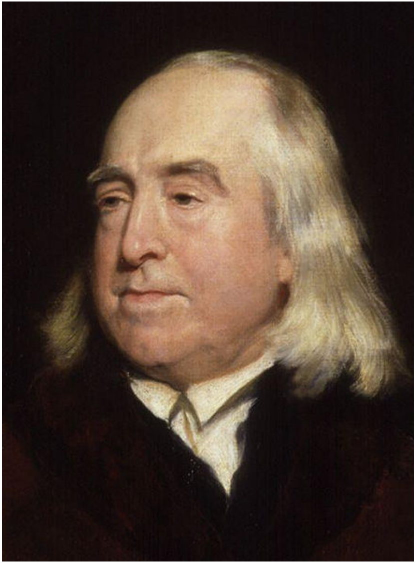


Measuring Happiness



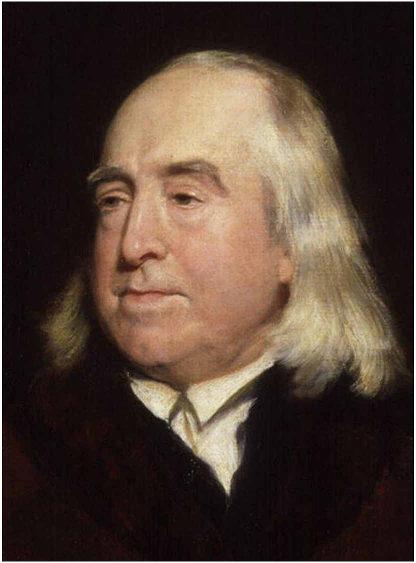
- Quantifying Happiness: How happy is society?

Measuring Happiness



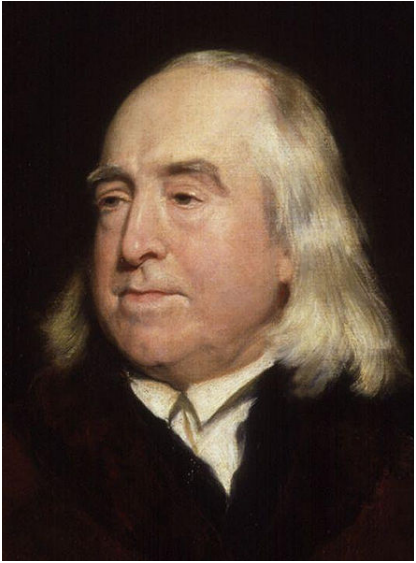
- Quantifying Happiness: How happy is society?
- How Happy is a Song?

Measuring Happiness



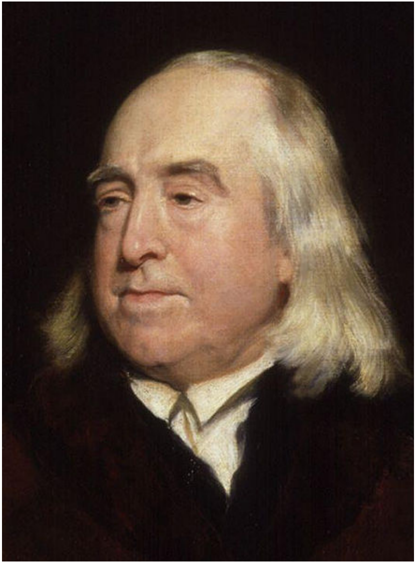
- Quantifying Happiness: How happy is society?
- How Happy is a Song?
- Blog posts?

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

Measuring Happiness

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

- Affective Norms for English Words (ANEW)

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

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)

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)

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 θ_j and document frequence 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 θ_j and document frequence X_{ij})

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

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.
⋮

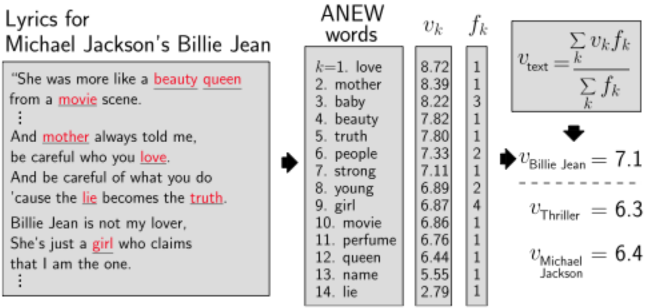
ANEW words	v_k	f_k
k=1. love	8.72	1
2. mother	8.39	1
3. baby	8.22	3
4. beauty	7.82	1
5. truth	7.80	1
6. people	7.33	2
7. strong	7.11	1
8. young	6.89	2
9. girl	6.87	4
10. movie	6.86	1
11. perfume	6.76	1
12. queen	6.44	1
13. name	5.55	1
14. lie	2.79	1

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

$$v_{\text{Billie Jean}} = 7.1$$

$$v_{\text{Thriller}} = 6.3$$

$$v_{\text{Michael Jackson}} = 6.4$$



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.
⋮

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

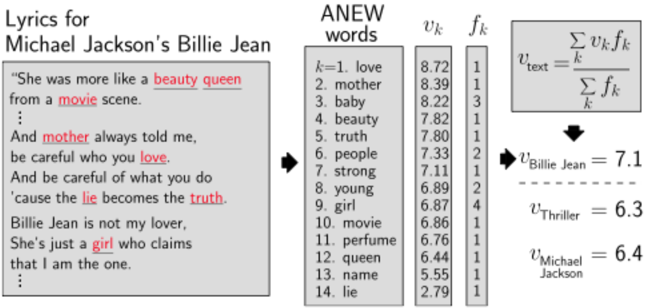
$$v_{\text{text}} = \frac{\sum_k v_k f_k}{\sum_k f_k}$$

$$v_{\text{Billie Jean}} = 7.1$$

$$v_{\text{Thriller}} = 6.3$$

$$v_{\text{Michael Jackson}} = 6.4$$

Homework Hints: One approach: write a `for` loop searching for words in dictionary (caution: is dictionary stemmed?)



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.
⋮

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

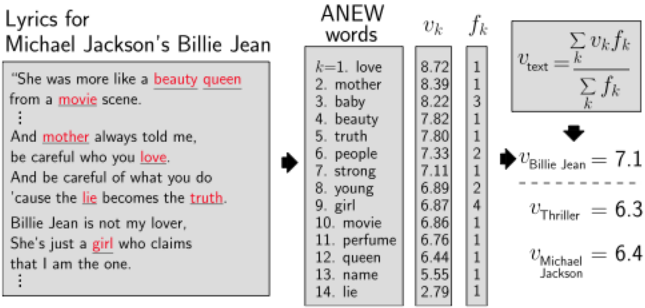
$$v_{\text{text}} = \frac{\sum_k v_k f_k}{\sum_k f_k}$$

$$v_{\text{Billie Jean}} = 7.1$$

$$v_{\text{Thriller}} = 6.3$$

$$v_{\text{Michael Jackson}} = 6.4$$

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



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.
⋮

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

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

$v_{\text{Billie Jean}} = 7.1$

$v_{\text{Thriller}} = 6.3$

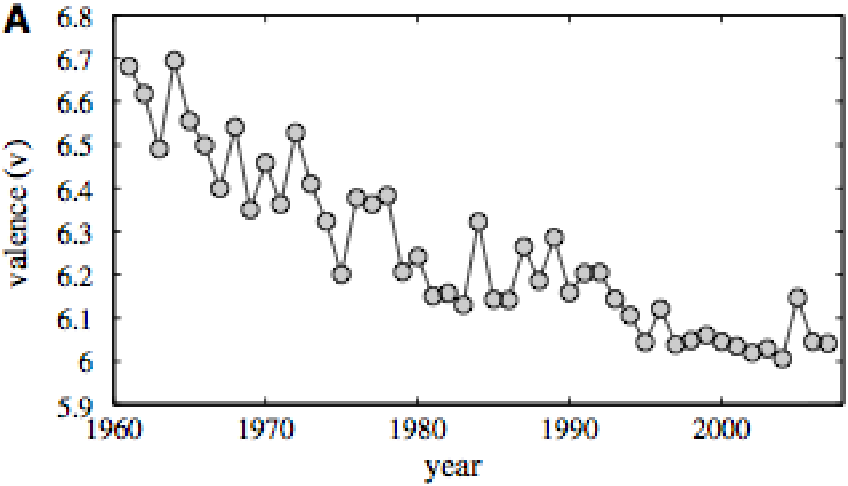
$v_{\text{Michael Jackson}} = 6.4$

Homework Hints: One approach: write a `for` loop searching for words in dictionary (caution: is dictionary stemmed?)

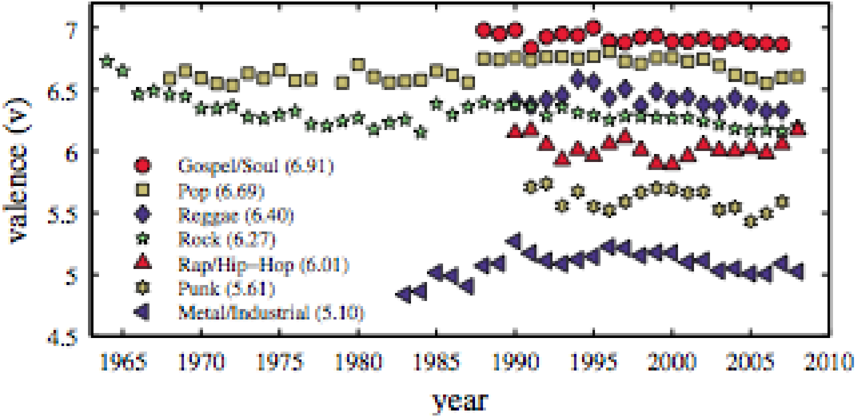
Happiest Song on Thriller?

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

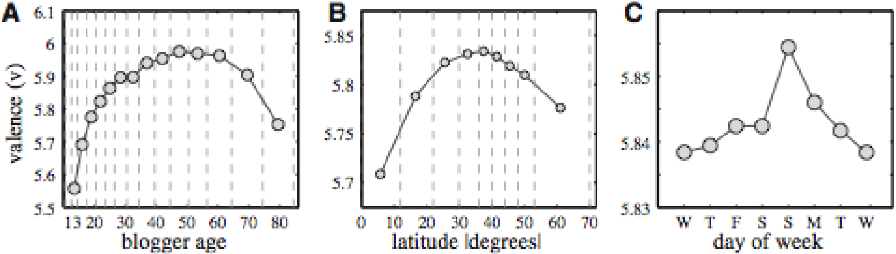
Happiness in Society



Happiness in Society



Happiness in Society



Supervised Learning

Supervised Learning

Supervised Methods:

Supervised Learning

Supervised Methods:

- Models for categorizing texts

Supervised Learning

Supervised Methods:

- Models for categorizing texts
    - Know (develop) categories before hand

Supervised Learning

Supervised Methods:

- Models for categorizing texts
    - Know (develop) categories before hand
    - Hand coding: assign documents to categories
    - Infer: new document assignment to categories (distribution of documents to categories)

Supervised Learning

Supervised Learning

- How to generate valid hand coding categories

Supervised Learning

- How to generate valid hand coding categories
    - Assessing coder performance
    - Assessing disagreement among coders
    - Evidence coders perform well

Supervised Learning

- How to generate valid hand coding categories
    - Assessing coder performance
    - Assessing disagreement among coders
    - Evidence coders perform well
- Supervised Learning Methods: Naive Bayes, LASSO (Ridge), ReadMe

Supervised Learning

- How to generate valid hand coding categories
    - Assessing coder performance
    - Assessing disagreement among coders
    - Evidence coders perform well
- Supervised Learning Methods: Naive Bayes, LASSO (Ridge), ReadMe
- Assessing Model Performance

Supervised Learning

- How to generate valid hand coding categories
  - Assessing coder performance
  - Assessing disagreement among coders
  - Evidence coders perform well
- Supervised Learning Methods: Naive Bayes, LASSO (Ridge), ReadMe
- Assessing Model Performance

Methods generalize beyond text

Components to Supervised Learning Method

Components to Supervised Learning Method

1) Set of categories

Components to Supervised Learning Method

1) Set of categories
    - Credit Claiming, Position Taking, Advertising
    - Positive Tone, Negative Tone
    - Pro-war, Ambiguous, Anti-war

Components to Supervised Learning Method

1) Set of categories
   - Credit Claiming, Position Taking, Advertising
   - Positive Tone, Negative Tone
   - Pro-war, Ambiguous, Anti-war
2) Set of hand-coded documents

Components to Supervised Learning Method

1) Set of categories
    - Credit Claiming, Position Taking, Advertising
    - Positive Tone, Negative Tone
    - Pro-war, Ambiguous, Anti-war
2) Set of hand-coded documents
    - Coding done by human coders
    - Training Set: documents we'll use to learn how to code
    - Validation Set: documents we'll use to learn how well we code

Components to Supervised Learning Method

1) Set of categories
    - Credit Claiming, Position Taking, Advertising
    - Positive Tone, Negative Tone
    - Pro-war, Ambiguous, Anti-war
2) Set of hand-coded documents
    - Coding done by human coders
    - Training Set: documents we'll use to learn how to code
    - Validation Set: documents we'll use to learn how well we code
3) Set of unlabeled documents

Components to Supervised Learning Method

1) Set of categories
    - Credit Claiming, Position Taking, Advertising
    - Positive Tone, Negative Tone
    - Pro-war, Ambiguous, Anti-war
2) Set of hand-coded documents
    - Coding done by human coders
    - Training Set: documents we'll use to learn how to code
    - Validation Set: documents we'll use to learn how well we code
3) Set of unlabeled documents
4) Method to extrapolate from hand coding to unlabeled documents

Three categories of documents

Hand labeled

- Training set (what we'll use to estimate model)
- Validation set (what we'll use to assess model)

Unlabeled

- Test set (what we'll use the model to categorize)

Label more documents than necessary to train model

Regression models

Suppose we have N documents, with each document i having label $y_i \in \{-1, 1\} \rightsquigarrow \{\text{not, credit claiming}\}$

Regression models

Suppose we have N documents, with each document i having label $y_i \in \{-1, 1\} \rightsquigarrow \{\text{not, credit claiming}\}$

We represent each document i is $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iJ})$.

Regression models

Suppose we have N documents, with each document i having label $y_i \in \{-1, 1\} \rightsquigarrow \{\text{not, credit claiming}\}$

We represent each document i is $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iJ})$.

$$f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) = \sum_{i=1}^{N} \left(y_i - \boldsymbol{\beta}^{'} \boldsymbol{x}_i\right)^2$$

Regression models

Suppose we have N documents, with each document i having label $y_i \in \{-1, 1\} \rightsquigarrow \{\text{not, credit claiming}\}$

We represent each document i is $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iJ})$.

$$
\begin{aligned}
f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) &= \sum_{i=1}^{N} \left(y_i - \boldsymbol{\beta}^{'} \boldsymbol{x}_i\right)^2 \\
\widehat{\boldsymbol{\beta}} &= \text{arg min}_{\boldsymbol{\beta}} \left\{ \sum_{i=1}^{N} \left(y_i - \boldsymbol{\beta}^{'} \boldsymbol{x}_i\right)^2 \right\}
\end{aligned}
$$

Regression models

Suppose we have N documents, with each document i having label $y_i \in \{-1, 1\} \rightsquigarrow \{\text{not, credit claiming}\}$
We represent each document i is $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iJ})$.

$$\begin{aligned} f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) &= \sum_{i=1}^{N} \left(y_i - \boldsymbol{\beta}^{'} \boldsymbol{x}_i \right)^2 \\ \widehat{\boldsymbol{\beta}} &= \text{arg min}_{\boldsymbol{\beta}} \left\{ \sum_{i=1}^{N} \left(y_i - \boldsymbol{\beta}^{'} \boldsymbol{x}_i \right)^2 \right\} \\ &= \left(\boldsymbol{X}^{'} \boldsymbol{X} \right)^{-1} \boldsymbol{X}^{'} \boldsymbol{Y} \end{aligned}$$

Regression models

Suppose we have N documents, with each document i having label $y_i \in \{-1, 1\} \rightsquigarrow \{\text{not, credit claiming}\}$

We represent each document i is $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iJ})$.

$$
\begin{aligned}
f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) &= \sum_{i=1}^{N} \left(y_i - \boldsymbol{\beta}^{'} \boldsymbol{x}_i\right)^2 \\
\widehat{\boldsymbol{\beta}} &= \text{arg min}_{\boldsymbol{\beta}} \left\{ \sum_{i=1}^{N} \left(y_i - \boldsymbol{\beta}^{'} \boldsymbol{x}_i\right)^2 \right\} \\
&= \left(\boldsymbol{X}^{'} \boldsymbol{X}\right)^{-1} \boldsymbol{X}^{'} \boldsymbol{Y}
\end{aligned}
$$

Problem:

Regression models

Suppose we have N documents, with each document i having label $y_i \in \{-1, 1\} \rightsquigarrow \{\text{not, credit claiming}\}$
We represent each document i is $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iJ})$.

$$
\begin{aligned}
f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) &= \sum_{i=1}^{N} \left(y_i - \boldsymbol{\beta}^{'} \boldsymbol{x}_i \right)^2 \\
\widehat{\boldsymbol{\beta}} &= \text{arg min}_{\boldsymbol{\beta}} \left\{ \sum_{i=1}^{N} \left(y_i - \boldsymbol{\beta}^{'} \boldsymbol{x}_i \right)^2 \right\} \\
&= \left(\boldsymbol{X}^{'} \boldsymbol{X} \right)^{-1} \boldsymbol{X}^{'} \boldsymbol{Y}
\end{aligned}
$$

Problem:

- J will likely be large (perhaps $J > N$)

Regression models

Suppose we have N documents, with each document i having label $y_i \in \{-1, 1\} \rightsquigarrow \{\text{not, credit claiming}\}$
We represent each document i is $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iJ})$.

$$
\begin{aligned}
f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) &= \sum_{i=1}^{N} \left(y_i - \boldsymbol{\beta}^{'} \boldsymbol{x}_i\right)^2 \\
\widehat{\boldsymbol{\beta}} &= \text{arg min}_{\boldsymbol{\beta}} \left\{ \sum_{i=1}^{N} \left(y_i - \boldsymbol{\beta}^{'} \boldsymbol{x}_i\right)^2 \right\} \\
&= \left(\boldsymbol{X}^{'}\boldsymbol{X}\right)^{-1} \boldsymbol{X}^{'}\boldsymbol{Y}
\end{aligned}
$$

Problem:

- J will likely be large (perhaps $J > N$)
- There many correlated variables

Regression models

Suppose we have N documents, with each document i having label $y_i \in \{-1, 1\} \rightsquigarrow \{\text{not, credit claiming}\}$
We represent each document i is $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iJ})$.

$$
\begin{aligned}
f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) &= \sum_{i=1}^{N} \left(y_i - \boldsymbol{\beta}^{'} \boldsymbol{x}_i\right)^2 \\
\widehat{\boldsymbol{\beta}} &= \text{arg min}_{\boldsymbol{\beta}} \left\{ \sum_{i=1}^{N} \left(y_i - \boldsymbol{\beta}^{'} \boldsymbol{x}_i\right)^2 \right\} \\
&= \left(\boldsymbol{X}^{'} \boldsymbol{X}\right)^{-1} \boldsymbol{X}^{'} \boldsymbol{Y}
\end{aligned}
$$

Problem:

- J will likely be large (perhaps $J > N$)
- There many correlated variables

Predictions will be variable

Mean Square Error

Suppose θ is some value of the true parameter

Mean Square Error

Suppose θ is some value of the true parameter
Bias:

Mean Square Error

Suppose θ is some value of the true parameter
Bias:

$$\text{Bias} = E[\widehat{\theta} - \theta]$$

Mean Square Error

Suppose θ is some value of the true parameter
Bias:

$$\text{Bias} = E[\widehat{\theta} - \theta]$$

We may care about average distance from truth

Mean Square Error

Suppose θ is some value of the true parameter
Bias:

$$\text{Bias} = E[\widehat{\theta} - \theta]$$

We may care about average distance from truth

$$\mathrm{E}[(\hat{\theta} - \theta)^2]$$

Mean Square Error

Suppose θ is some value of the true parameter
Bias:

$$\text{Bias} \quad = \quad E[\widehat{\theta} - \theta]$$

We may care about average distance from truth

$$\text{E}[(\hat{\theta} - \theta)^2] \quad = \quad E[\hat{\theta}^2] - 2\theta E[\hat{\theta}] + \theta^2$$

Mean Square Error

Suppose θ is some value of the true parameter
Bias:

$$\text{Bias} \quad = \quad E[\widehat{\theta} - \theta]$$

We may care about average distance from truth

$$\begin{aligned} \mathrm{E}[(\hat{\theta} - \theta)^2] &= E[\hat{\theta}^2] - 2\theta E[\hat{\theta}] + \theta^2 \\ &= E[\hat{\theta}^2] - E[\hat{\theta}]^2 + E[\hat{\theta}]^2 - 2\theta E[\hat{\theta}] + \theta^2 \end{aligned}$$

Mean Square Error

Suppose θ is some value of the true parameter
Bias:

$$\text{Bias} = E[\widehat{\theta} - \theta]$$

We may care about average distance from truth

$$\begin{aligned}
\text{E}[(\hat{\theta} - \theta)^2] &= E[\hat{\theta}^2] - 2\theta E[\hat{\theta}] + \theta^2 \\
&= E[\hat{\theta}^2] - E[\hat{\theta}]^2 + E[\hat{\theta}]^2 - 2\theta E[\hat{\theta}] + \theta^2 \\
&= E[\hat{\theta}^2] - E[\hat{\theta}]^2 + (E[\widehat{\theta} - \theta])^2
\end{aligned}$$

Mean Square Error

Suppose θ is some value of the true parameter
Bias:

$$\text{Bias} \quad = \quad E[\widehat{\theta} - \theta]$$

We may care about average distance from truth

$$\begin{aligned}
\text{E}[(\hat{\theta} - \theta)^2] &= E[\hat{\theta}^2] - 2\theta E[\hat{\theta}] + \theta^2 \\
&= E[\hat{\theta}^2] - E[\hat{\theta}]^2 + E[\hat{\theta}]^2 - 2\theta E[\hat{\theta}] + \theta^2 \\
&= E[\hat{\theta}^2] - E[\hat{\theta}]^2 + (E[\widehat{\theta} - \theta])^2 \\
&= \text{Var}(\hat{\theta}) + \text{Bias}^2
\end{aligned}$$

Mean Square Error

Suppose θ is some value of the true parameter
Bias:

$$\text{Bias} \quad = \quad E[\widehat{\theta} - \theta]$$

We may care about average distance from truth

$$\begin{aligned}
\mathrm{E}[(\hat{\theta} - \theta)^2] &= E[\hat{\theta}^2] - 2\theta E[\hat{\theta}] + \theta^2 \\
&= E[\hat{\theta}^2] - E[\hat{\theta}]^2 + E[\hat{\theta}]^2 - 2\theta E[\hat{\theta}] + \theta^2 \\
&= E[\hat{\theta}^2] - E[\hat{\theta}]^2 + (E[\widehat{\theta} - \theta])^2 \\
&= \mathrm{Var}(\hat{\theta}) + \mathrm{Bias}^2
\end{aligned}$$

To reduce MSE, we are willing to induce bias to decrease variance⇝ methods that shrink coefficeints toward zero

Ridge Regression

Penalty for model complexity

Ridge Regression

Penalty for model complexity

$$f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y})$$

Ridge Regression

Penalty for model complexity

$$f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) \quad = \quad \sum_{i=1}^{N}\left(y_i - \ \beta_0 - \sum_{j=1}^{J}\beta_j x_{ij}\right)^2$$

Ridge Regression

Penalty for model complexity

$$f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) = \sum_{i=1}^{N}\left(y_i - \beta_0 - \sum_{j=1}^{J}\beta_j x_{ij}\right)^2 + \underbrace{\lambda\sum_{j=1}^{J}\beta_j^2}_{\text{Penalty}}$$

Ridge Regression

Penalty for model complexity

$$f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) \;=\; \sum_{i=1}^{N}\left(y_i - \ \beta_0 - \sum_{j=1}^{J}\beta_j x_{ij}\right)^2 + \underbrace{\lambda\sum_{j=1}^{J}\beta_j^2}_{\text{Penalty}}$$

where:

Ridge Regression

Penalty for model complexity

$$f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) \quad = \quad \sum_{i=1}^{N} \left(y_i - \ \beta_0 - \sum_{j=1}^{J} \beta_j x_{ij} \right)^2 + \underbrace{\lambda \sum_{j=1}^{J} \beta_j^2}_{\text{Penalty}}$$

where:

- $\beta_0 \rightsquigarrow$ intercept

Ridge Regression

Penalty for model complexity

$$f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) = \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{J} \beta_j x_{ij} \right)^2 + \underbrace{\lambda \sum_{j=1}^{J} \beta_j^2}_{\text{Penalty}}$$

where:

- $\beta_0 \rightsquigarrow$ intercept
- $\lambda \rightsquigarrow$ penalty parameter

Ridge Regression

Penalty for model complexity

$$f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) \ = \ \sum_{i=1}^{N} \left(y_i - \ \beta_0 - \sum_{j=1}^{J} \beta_j x_{ij} \right)^2 + \underbrace{\lambda \sum_{j=1}^{J} \beta_j^2}_{\text{Penalty}}$$

where:

- $\beta_0 \rightsquigarrow$ intercept
- $\lambda \rightsquigarrow$ penalty parameter
- Standardized $\boldsymbol{X}$ (coefficients on same scale)

Ridge Regression$\rightsquigarrow$ Optimization

$$\boldsymbol{\beta}^{\text{Ridge}} = \text{arg min}_{\boldsymbol{\beta}} \left\{ f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) \right\}$$

Ridge Regression$\rightsquigarrow$ Optimization

$$\begin{aligned}
\boldsymbol{\beta}^{\text{Ridge}} &= \text{arg min}_{\boldsymbol{\beta}} \left\{ f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) \right\} \\
&= \text{arg min}_{\boldsymbol{\beta}} \left\{ \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{J} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{J} \beta_j^2 \right\}
\end{aligned}$$

Ridge Regression $\rightsquigarrow$ Optimization

$$\begin{aligned}
\boldsymbol{\beta}^{\text{Ridge}} &= \text{arg min}_{\boldsymbol{\beta}} \left\{ f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) \right\} \\
&= \text{arg min}_{\boldsymbol{\beta}} \left\{ \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{J} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{J} \beta_j^2 \right\} \\
&= \text{arg min}_{\boldsymbol{\beta}} \left\{ (\boldsymbol{Y} - \boldsymbol{X}^{'} \boldsymbol{\beta})^{'} (\boldsymbol{Y} - \boldsymbol{X}^{'} \boldsymbol{\beta}) + \lambda \boldsymbol{\beta}^{'} \boldsymbol{\beta} \right\}
\end{aligned}$$

Demean the data and set $\beta_0 = \bar{y} = \sum_{i=1}^{N} \frac{y_i}{N}$

Ridge Regression$\rightsquigarrow$ Optimization

$$
\begin{aligned}
\boldsymbol{\beta}^{\text{Ridge}} &= \text{arg min}_{\boldsymbol{\beta}} \left\{ f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) \right\} \\
&= \text{arg min}_{\boldsymbol{\beta}} \left\{ \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{J} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{J} \beta_j^2 \right\} \\
&= \text{arg min}_{\boldsymbol{\beta}} \left\{ (\boldsymbol{Y} - \boldsymbol{X}^{'}\boldsymbol{\beta})^{'}(\boldsymbol{Y} - \boldsymbol{X}^{'}\boldsymbol{\beta}) + \lambda \boldsymbol{\beta}^{'}\boldsymbol{\beta} \right\} \\
&= \left(\boldsymbol{X}^{'}\boldsymbol{X} + \lambda \boldsymbol{I}_J \right)^{-1} \boldsymbol{X}^{'}\boldsymbol{Y}
\end{aligned}
$$

Demean the data and set $\beta_0 = \bar{y} = \sum_{i=1}^{N} \frac{y_i}{N}$

Ridge Regression$\rightsquigarrow$ Optimization

$$
\begin{aligned}
\boldsymbol{\beta}^{\text{Ridge}} &= \text{arg min}_{\boldsymbol{\beta}} \left\{ f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) \right\} \\
&= \text{arg min}_{\boldsymbol{\beta}} \left\{ \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{J} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{J} \beta_j^2 \right\} \\
&= \text{arg min}_{\boldsymbol{\beta}} \left\{ (\boldsymbol{Y} - \boldsymbol{X}^{'} \boldsymbol{\beta})^{'} (\boldsymbol{Y} - \boldsymbol{X}^{'} \boldsymbol{\beta}) + \lambda \boldsymbol{\beta}^{'} \boldsymbol{\beta} \right\} \\
&= \left(\boldsymbol{X}^{'} \boldsymbol{X} + \lambda \boldsymbol{I}_J \right)^{-1} \boldsymbol{X}^{'} \boldsymbol{Y}
\end{aligned}
$$

Demean the data and set $\beta_0 = \bar{y} = \sum_{i=1}^{N} \frac{y_i}{N}$

Ridge Regression$\rightsquigarrow$ Intuition (1)

Suppose $\boldsymbol{X}^{'}\boldsymbol{X} = \boldsymbol{I}_J$.

Ridge Regression$\rightsquigarrow$ Intuition (1)

Suppose $\boldsymbol{X}^{'}\boldsymbol{X} = \boldsymbol{I}_{J}$.

$$\widehat{\beta} \;=\; \left(\boldsymbol{X}^{'}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^{'}\boldsymbol{Y}$$

Ridge Regression$\leadsto$ Intuition (1)

Suppose $\boldsymbol{X}^{'}\boldsymbol{X} = \boldsymbol{I}_J$.

$$\begin{aligned}\widehat{\beta} &= \left(\boldsymbol{X}^{'}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^{'}\boldsymbol{Y} \\ &= \boldsymbol{X}^{'}\boldsymbol{Y}\end{aligned}$$

Ridge Regression$\rightsquigarrow$ Intuition (1)

Suppose $\boldsymbol{X}^{'}\boldsymbol{X} = \boldsymbol{I}_J$.

$$\begin{aligned}
\widehat{\boldsymbol{\beta}} &= \left(\boldsymbol{X}^{'}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^{'}\boldsymbol{Y} \\
&= \boldsymbol{X}^{'}\boldsymbol{Y} \\
\boldsymbol{\beta}^{\text{ridge}} &= \left(\boldsymbol{X}^{'}\boldsymbol{X} + \lambda\boldsymbol{I}_J\right)^{-1}\boldsymbol{X}^{'}\boldsymbol{Y}
\end{aligned}$$

Ridge Regression$\rightsquigarrow$ Intuition (1)

Suppose $\boldsymbol{X}^{'}\boldsymbol{X} = \boldsymbol{I}_J$.

$$\begin{aligned}
\widehat{\boldsymbol{\beta}} &= \left(\boldsymbol{X}^{'}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^{'}\boldsymbol{Y} \\
&= \boldsymbol{X}^{'}\boldsymbol{Y} \\
\boldsymbol{\beta}^{\text{ridge}} &= \left(\boldsymbol{X}^{'}\boldsymbol{X} + \lambda\boldsymbol{I}_J\right)^{-1}\boldsymbol{X}^{'}\boldsymbol{Y} \\
&= \left(\boldsymbol{I}_j + \lambda\boldsymbol{I}_j\right)^{-1}\boldsymbol{X}^{'}\boldsymbol{Y}
\end{aligned}$$

Ridge Regression$\leadsto$ Intuition (1)

Suppose $\boldsymbol{X}^{'}\boldsymbol{X} = \boldsymbol{I}_J$.

$$\begin{aligned}
\widehat{\boldsymbol{\beta}} &= \left(\boldsymbol{X}^{'}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^{'}\boldsymbol{Y} \\
&= \boldsymbol{X}^{'}\boldsymbol{Y} \\
\boldsymbol{\beta}^{\text{ridge}} &= \left(\boldsymbol{X}^{'}\boldsymbol{X} + \lambda\boldsymbol{I}_J\right)^{-1}\boldsymbol{X}^{'}\boldsymbol{Y} \\
&= (\boldsymbol{I}_j + \lambda\boldsymbol{I}_j)^{-1}\boldsymbol{X}^{'}\boldsymbol{Y} \\
&= (\boldsymbol{I}_j + \lambda\boldsymbol{I}_j)^{-1}\widehat{\boldsymbol{\beta}}
\end{aligned}$$

Ridge Regression$\rightsquigarrow$ Intuition (1)

Suppose $\boldsymbol{X}^{'}\boldsymbol{X} = \boldsymbol{I}_J$.

$$\begin{aligned}
\widehat{\boldsymbol{\beta}} &= \left(\boldsymbol{X}^{'}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^{'}\boldsymbol{Y} \\
&= \boldsymbol{X}^{'}\boldsymbol{Y} \\
\boldsymbol{\beta}^{\text{ridge}} &= \left(\boldsymbol{X}^{'}\boldsymbol{X} + \lambda\boldsymbol{I}_J\right)^{-1}\boldsymbol{X}^{'}\boldsymbol{Y} \\
&= \left(\boldsymbol{I}_j + \lambda\boldsymbol{I}_j\right)^{-1}\boldsymbol{X}^{'}\boldsymbol{Y} \\
&= \left(\boldsymbol{I}_j + \lambda\boldsymbol{I}_j\right)^{-1}\widehat{\boldsymbol{\beta}} \\
\beta_j^{\text{Ridge}} &= \frac{\widehat{\beta}_j}{1+\lambda}
\end{aligned}$$

Ridge Regression$\rightsquigarrow$ Intuition (2)

$$
\begin{aligned}
\boldsymbol{\beta}_j & \sim \text{Normal}(0, \tau^2) \\
y_i & \sim \text{Normal}(\beta_0 + \boldsymbol{x}_i^{'}\boldsymbol{\beta}, \sigma^2)
\end{aligned}
$$

Ridge Regression$\rightsquigarrow$ Intuition (2)

$$\begin{aligned} \boldsymbol{\beta}_j &\sim \text{Normal}(0, \tau^2) \\ y_i &\sim \text{Normal}(\beta_0 + \boldsymbol{x}_i^{'}\boldsymbol{\beta}, \sigma^2) \end{aligned}$$

$$p(\boldsymbol{\beta}|\boldsymbol{X}, \boldsymbol{Y}) \quad \propto \quad \prod_{j=1}^{J} p(\beta_j) \prod_{i=1}^{N} p(y_i|\boldsymbol{x}_i, \boldsymbol{\beta})$$

Ridge Regression⇝ Intuition (2)

$$\begin{aligned}\boldsymbol{\beta}_j &\sim \text{Normal}(0, \tau^2)\\ y_i &\sim \text{Normal}(\beta_0 + \boldsymbol{x}_i^{'}\boldsymbol{\beta}, \sigma^2)\end{aligned}$$

$$\begin{aligned}p(\boldsymbol{\beta}|\boldsymbol{X}, \boldsymbol{Y}) &\propto \prod_{j=1}^{J} p(\beta_j) \prod_{i=1}^{N} p(y_i|\boldsymbol{x}_i, \boldsymbol{\beta})\\ &\propto \prod_{j=1}^{J} \frac{1}{\sqrt{2\pi}\tau} \exp\left(-\frac{\beta_j^2}{2\tau^2}\right) \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y_i - \beta_0 - \boldsymbol{x}_i^{'}\boldsymbol{\beta})^2}{2\sigma^2}\right)\end{aligned}$$

Ridge Regression $\rightsquigarrow$ Intuition (2)

$$\log p(\beta|\boldsymbol{X}, \boldsymbol{Y}) = -\sum_{j=1}^{J} \frac{\beta_j^2}{2\tau^2} - \sum_{i=1}^{N} \frac{(y_i - \beta_0 - \boldsymbol{x}^{'}\boldsymbol{\beta})^2}{2\sigma^2}$$

Ridge Regression$\rightsquigarrow$ Intuition (2)

$$\begin{aligned}
\log p(\boldsymbol{\beta}|\boldsymbol{X}, \boldsymbol{Y}) &= -\sum_{j=1}^{J} \frac{\beta_j^2}{2\tau^2} - \sum_{i=1}^{N} \frac{(y_i - \beta_0 - \boldsymbol{x}^{'}\boldsymbol{\beta})^2}{2\sigma^2} \\
-2\sigma^2 \log p(\boldsymbol{\beta}|\boldsymbol{X}, \boldsymbol{Y}) &= \sum_{i=1}^{N}(y_i - \beta_0 - \boldsymbol{x}^{'}\boldsymbol{\beta})^2 + \sum_{j=1}^{J} \frac{\sigma^2}{\tau^2}\beta_j^2
\end{aligned}$$

Ridge Regression$\rightsquigarrow$ Intuition (2)

$$\begin{aligned}
\log p(\boldsymbol{\beta}|\boldsymbol{X}, \boldsymbol{Y}) &= -\sum_{j=1}^{J} \frac{\beta_j^2}{2\tau^2} - \sum_{i=1}^{N} \frac{(y_i - \beta_0 - \boldsymbol{x}^{'}\boldsymbol{\beta})^2}{2\sigma^2} \\
-2\sigma^2 \log p(\boldsymbol{\beta}|\boldsymbol{X}, \boldsymbol{Y}) &= \sum_{i=1}^{N}(y_i - \beta_0 - \boldsymbol{x}^{'}\boldsymbol{\beta})^2 + \sum_{j=1}^{J} \frac{\sigma^2}{\tau^2}\beta_j^2
\end{aligned}$$

where:

Ridge Regression$\rightsquigarrow$ Intuition (2)

$$\begin{aligned}
\log p(\boldsymbol{\beta}|\boldsymbol{X}, \boldsymbol{Y}) &= -\sum_{j=1}^{J} \frac{\beta_j^2}{2\tau^2} - \sum_{i=1}^{N} \frac{(y_i - \beta_0 - \boldsymbol{x}^{'}\boldsymbol{\beta})^2}{2\sigma^2} \\
-2\sigma^2 \log p(\boldsymbol{\beta}|\boldsymbol{X}, \boldsymbol{Y}) &= \sum_{i=1}^{N}(y_i - \beta_0 - \boldsymbol{x}^{'}\boldsymbol{\beta})^2 + \sum_{j=1}^{J} \frac{\sigma^2}{\tau^2}\beta_j^2
\end{aligned}$$

where:

- $\lambda = \frac{\sigma^2}{\tau^2}$

Ridge Regression $\rightsquigarrow$ Intuition (3)

Definition

Suppose $\boldsymbol{X}$ is an $N \times J$ matrix. Then $\boldsymbol{X}$ can be written as:

$$\boldsymbol{X} = \underbrace{\boldsymbol{U}}_{N\times N} \underbrace{\boldsymbol{S}}_{N\times J} \underbrace{\boldsymbol{V}^{'}}_{J\times J}$$

Where:

$$\begin{aligned} \boldsymbol{U}^{'}\boldsymbol{U} &= \boldsymbol{I}_N \\ \boldsymbol{V}^{'}\boldsymbol{V} &= \boldsymbol{V}\boldsymbol{V}^{'} = \boldsymbol{I}_J \end{aligned}$$

$\boldsymbol{S}$ *contains* $\min(N, J)$ *singular values,* $\sqrt{\lambda_j} \geq 0$ *down the diagonal and then 0's for the remaining entries*

Ridge Regression $\rightsquigarrow$ Intuition (3)

Recall: PCA:

$$\frac{1}{N}\boldsymbol{X}^{'}\boldsymbol{X} = \underbrace{\boldsymbol{W}}_{\text{eigenvectors}} \begin{pmatrix} \lambda_1 & 0 & \ldots & 0 \\ 0 & \lambda_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \lambda_J \end{pmatrix} \underbrace{\boldsymbol{W}^{'}}_{\text{eigenvectors}}$$

Ridge Regression $\rightsquigarrow$ Intuition (3)

Recall: PCA:

$$\frac{1}{N}\boldsymbol{X}^{'}\boldsymbol{X} = \underbrace{\boldsymbol{W}}_{\text{eigenvectors}} \begin{pmatrix} \lambda_1 & 0 & \ldots & 0 \\ 0 & \lambda_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \lambda_J \end{pmatrix} \underbrace{\boldsymbol{W}^{'}}_{\text{eigenvectors}}$$

Using SVD:

Ridge Regression ⇝ Intuition (3)

Recall: PCA:

$$\frac{1}{N}\boldsymbol{X}^{'}\boldsymbol{X} = \underbrace{\boldsymbol{W}}_{\text{eigenvectors}} \begin{pmatrix} \lambda_1 & 0 & \ldots & 0 \\ 0 & \lambda_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \lambda_J \end{pmatrix} \underbrace{\boldsymbol{W}^{'}}_{\text{eigenvectors}}$$

Using SVD:

$$\frac{1}{N}\boldsymbol{X}^{'}\boldsymbol{X} \quad = \quad \boldsymbol{V}\boldsymbol{S}^{'}\underbrace{\left(\boldsymbol{U}^{'}\boldsymbol{U}\right)}_{\boldsymbol{I}_J}\boldsymbol{S}\boldsymbol{V}^{'}$$

Ridge Regression $\rightsquigarrow$ Intuition (3)

Recall: PCA:

$$\frac{1}{N}\boldsymbol{X}^{'}\boldsymbol{X} = \underbrace{\boldsymbol{W}}_{\text{eigenvectors}} \begin{pmatrix} \lambda_1 & 0 & \ldots & 0 \\ 0 & \lambda_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \lambda_J \end{pmatrix} \underbrace{\boldsymbol{W}^{'}}_{\text{eigenvectors}}$$

Using SVD:

$$\begin{aligned} \frac{1}{N}\boldsymbol{X}^{'}\boldsymbol{X} &= \boldsymbol{V}\boldsymbol{S}^{'} \underbrace{\left(\boldsymbol{U}^{'}\boldsymbol{U}\right)}_{\boldsymbol{I}_J} \boldsymbol{S}\boldsymbol{V}^{'} \\ &= \frac{1}{N}\boldsymbol{V}\boldsymbol{S}^{'}\boldsymbol{S}\boldsymbol{V}^{'} \end{aligned}$$

Ridge Regression $\rightsquigarrow$ Intuition (3)

Recall: PCA:

$$\frac{1}{N}\boldsymbol{X}^{'}\boldsymbol{X} = \underbrace{\boldsymbol{W}}_{\text{eigenvectors}} \begin{pmatrix} \lambda_1 & 0 & \ldots & 0 \\ 0 & \lambda_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \lambda_J \end{pmatrix} \underbrace{\boldsymbol{W}^{'}}_{\text{eigenvectors}}$$

Using SVD:

$$\begin{aligned} \frac{1}{N}\boldsymbol{X}^{'}\boldsymbol{X} &= \boldsymbol{V}\boldsymbol{S}^{'}\underbrace{\left(\boldsymbol{U}^{'}\boldsymbol{U}\right)}_{\boldsymbol{I}_J}\boldsymbol{S}\boldsymbol{V}^{'} \\ &= \frac{1}{N}\boldsymbol{V}\boldsymbol{S}^{'}\boldsymbol{S}\boldsymbol{V}^{'} \\ &= \underbrace{\boldsymbol{V}}_{\text{eigenvectors}} \begin{pmatrix} \lambda_1 & 0 & \ldots & 0 \\ 0 & \lambda_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \lambda_J \end{pmatrix} \underbrace{\boldsymbol{V}^{'}}_{\text{eigenvectors}} \end{aligned}$$

Ridge Regression ⇝ Intuition (3)

We can write the predicted values for a regular regression as

$$
\begin{aligned}
\hat{Y} &= \boldsymbol{X}\hat{\beta} \\
&= \boldsymbol{X}\left(\boldsymbol{X}^{'}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^{'}\boldsymbol{Y} \\
&= \boldsymbol{U}\boldsymbol{U}^{'}\boldsymbol{Y} = \sum_{j=1}^{J}\boldsymbol{u}_j\boldsymbol{u}_j^{'}\boldsymbol{Y}
\end{aligned}
$$

Ridge Regression $\rightsquigarrow$ Intuition (3)

We can write the predicted values for a regular regression as

$$\begin{aligned}\hat{Y} &= \boldsymbol{X}\hat{\boldsymbol{\beta}} \\ &= \boldsymbol{X}\left(\boldsymbol{X}^{'}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^{'}\boldsymbol{Y} \\ &= \boldsymbol{U}\boldsymbol{U}^{'}\boldsymbol{Y} = \sum_{j=1}^{J}\boldsymbol{u}_j\boldsymbol{u}_j^{'}\boldsymbol{Y}\end{aligned}$$

We can write $\boldsymbol{\beta}^{\text{ridge}}$ as

Ridge Regression $\rightsquigarrow$ Intuition (3)

We can write the predicted values for a regular regression as

$$\begin{aligned}\hat{Y} &= \boldsymbol{X}\hat{\boldsymbol{\beta}} \\ &= \boldsymbol{X}\left(\boldsymbol{X}^{'}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^{'}\boldsymbol{Y} \\ &= \boldsymbol{U}\boldsymbol{U}^{'}\boldsymbol{Y} = \sum_{j=1}^{J}\boldsymbol{u}_j\boldsymbol{u}_j^{'}\boldsymbol{Y}\end{aligned}$$

We can write $\boldsymbol{\beta}^{\text{ridge}}$ as

$$\hat{Y}^{\text{ridge}} = \boldsymbol{X}\left(\boldsymbol{X}^{'}\boldsymbol{X} + \lambda\boldsymbol{I}_J\right)^{-1}\boldsymbol{X}^{'}\boldsymbol{Y}$$

Ridge Regression $\rightsquigarrow$ Intuition (3)

We can write the predicted values for a regular regression as

$$\begin{aligned}\hat{Y} &= \boldsymbol{X}\hat{\boldsymbol{\beta}} \\ &= \boldsymbol{X}\left(\boldsymbol{X}^{'}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^{'}\boldsymbol{Y} \\ &= \boldsymbol{U}\boldsymbol{U}^{'}\boldsymbol{Y} = \sum_{j=1}^{J}\boldsymbol{u}_j\boldsymbol{u}_j^{'}\boldsymbol{Y}\end{aligned}$$

We can write $\boldsymbol{\beta}^{\text{ridge}}$ as

$$\begin{aligned}\hat{Y}^{\text{ridge}} &= \boldsymbol{X}\left(\boldsymbol{X}^{'}\boldsymbol{X} + \lambda\boldsymbol{I}_J\right)^{-1}\boldsymbol{X}^{'}\boldsymbol{Y} \\ &= \boldsymbol{U}\tilde{\boldsymbol{S}}\boldsymbol{U}^{'}\boldsymbol{Y}\end{aligned}$$

Where

$$\tilde{\boldsymbol{S}} = \left[\boldsymbol{S}(\boldsymbol{S}^{'}\boldsymbol{S} + \lambda\boldsymbol{I}_J)^{-1}\boldsymbol{S}\right]$$

Ridge Regression $\rightsquigarrow$ Intuition (3)

We can write the predicted values for a regular regression as

$$\begin{aligned}\hat{Y} &= \boldsymbol{X}\hat{\boldsymbol{\beta}} \\ &= \boldsymbol{X}\left(\boldsymbol{X}^{'}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^{'}\boldsymbol{Y} \\ &= \boldsymbol{U}\boldsymbol{U}^{'}\boldsymbol{Y} = \sum_{j=1}^{J}\boldsymbol{u}_j\boldsymbol{u}_j^{'}\boldsymbol{Y}\end{aligned}$$

We can write $\boldsymbol{\beta}^{\text{ridge}}$ as

$$\begin{aligned}\hat{Y}^{\text{ridge}} &= \boldsymbol{X}\left(\boldsymbol{X}^{'}\boldsymbol{X} + \lambda\boldsymbol{I}_J\right)^{-1}\boldsymbol{X}^{'}\boldsymbol{Y} \\ &= \boldsymbol{U}\tilde{\boldsymbol{S}}\boldsymbol{U}^{'}\boldsymbol{Y}\end{aligned}$$

Where

$$\tilde{\boldsymbol{S}} = \left[\boldsymbol{S}(\boldsymbol{S}^{'}\boldsymbol{S} + \lambda\boldsymbol{I}_J)^{-1}\boldsymbol{S}\right]$$

Which we can write as:

$$\hat{Y}^{\text{ridge}} = \sum_{j=1}^{J}\boldsymbol{u}_j\frac{\lambda_j}{\lambda_j + \lambda}\boldsymbol{u}_j^{'}\boldsymbol{Y}$$

Degrees of Freedom for Ridge

We will say that the degrees of freedom for Ridge regression with penalty λ is

$$\text{dof}(\lambda) = \sum_{j=1}^{J} \frac{\lambda_j}{\lambda_j + \lambda}$$

Lasso Regression Objective Function

Different Penalty for Model Complexity

$$f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) = \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{J} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{J} \underbrace{|\beta_j|}_{\text{Penalty}}$$

Lasso Regression Objective Function

Different Penalty for Model Complexity

$$f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) = \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{J} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{J} \underbrace{|\beta_j|}_{\text{Penalty}}$$

Lasso Regression Optimization

Definition

Coordinate Descent Algorithms:
Consider $g : \Re^J \rightarrow \Re$. *Our goal is to find* $\boldsymbol{x}^* \in \Re^J$ *such that* $g(\boldsymbol{x}^*) \leq g(\boldsymbol{x})$ *for all* $\boldsymbol{x} \in \Re$.
To find $\boldsymbol{x}^*$*:*
Until convergence: for each iteration t *and each coordinate* j

$$x_j^{t+1} = arg\ min_{x_j \in \Re} g(x_1^{t+1}, x_2^{t+1}, \ldots, x_{j-1}^{t+1}, x_j, x_{j+1}^{t}, \ldots, x_J^{t})$$

Lasso Regression Optimization: Coordinate Descent

$$\tilde{f}(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) = \frac{1}{2N}\sum_{i=1}^{N}\left(y_i - \beta_0 - \sum_{j=1}^{J}\beta_j x_{ij}\right)^2 + \lambda\sum_{j=1}^{J}|\beta_j|$$

Lasso Regression Optimization: Coordinate Descent

$$\tilde{f}(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) = \frac{1}{2N}\sum_{i=1}^{N}\left(y_i - \beta_0 - \sum_{j=1}^{J}\beta_j x_{ij}\right)^2 + \lambda\sum_{j=1}^{J}|\beta_j|$$

- Case 1: If $\beta_j = 0 \rightsquigarrow$ not differentiable. But $\beta_j = 0$

Lasso Regression Optimization: Coordinate Descent

$$\tilde{f}(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) \quad = \quad \frac{1}{2N}\sum_{i=1}^{N}\left(y_i - \beta_0 - \sum_{j=1}^{J}\beta_j x_{ij}\right)^2 + \lambda \sum_{j=1}^{J}|\beta_j|$$

- Case 1: If $\beta_j = 0 \rightsquigarrow$ not differentiable. But $\beta_j = 0$
- Case 2: If $\beta_j > (<)0 \rightsquigarrow$ differentiable $\rightsquigarrow$ differentiate and solve for β_j

Lasso Regression Optimization: Coordinate Descent

$$\tilde{f}(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) = \frac{1}{2N}\sum_{i=1}^{N}\left(y_i - \beta_0 - \sum_{j=1}^{J}\beta_j x_{ij}\right)^2 + \lambda\sum_{j=1}^{J}|\beta_j|$$

- Case 1: If $\beta_j = 0 \rightsquigarrow$ not differentiable. But $\beta_j = 0$
- Case 2: If $\beta_j > (<)0 \rightsquigarrow$ differentiable $\rightsquigarrow$ differentiate and solve for β_j

Define $\tilde{y}_i^j = \beta_0 + \sum_{l \neq j} x_{il}\beta_l$

Lasso Regression Optimization: Coordinate Descent

$$\tilde{f}(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) = \frac{1}{2N}\sum_{i=1}^{N}\left(y_i - \beta_0 - \sum_{j=1}^{J}\beta_j x_{ij}\right)^2 + \lambda \sum_{j=1}^{J}|\beta_j|$$

- Case 1: If $\beta_j = 0 \rightsquigarrow$ not differentiable. But $\beta_j = 0$
- Case 2: If $\beta_j > (<) 0 \rightsquigarrow$ differentiable $\rightsquigarrow$ differentiate and solve for β_j

Define $\tilde{y}_i^j = \beta_0 + \sum_{l \neq j} x_{il}\beta_l$
$r^j \equiv \frac{1}{N}\sum_{i=1}^{N} x_{ij}(y_i - \tilde{y}_i^j)$

Lasso Regression Optimization: Coordinate Descent

$$\tilde{f}(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) \quad = \quad \frac{1}{2N}\sum_{i=1}^{N}\left(y_i - \beta_0 - \sum_{j=1}^{J}\beta_j x_{ij}\right)^2 + \lambda \sum_{j=1}^{J}|\beta_j|$$

- Case 1: If $\beta_j = 0 \rightsquigarrow$ not differentiable. But $\beta_j = 0$
- Case 2: If $\beta_j > (<)0 \rightsquigarrow$ differentiable $\rightsquigarrow$ differentiate and solve for β_j

Define $\tilde{y}_i^j = \beta_0 + \sum_{l \neq j} x_{il}\beta_l$
$r^j \equiv \frac{1}{N}\sum_{i=1}^{N} x_{ij}(y_i - \tilde{y}_i^j)$
Update step for β_j is

Lasso Regression Optimization: Coordinate Descent

$$\tilde{f}(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) = \frac{1}{2N}\sum_{i=1}^{N}\left(y_i - \beta_0 - \sum_{j=1}^{J}\beta_j x_{ij}\right)^2 + \lambda\sum_{j=1}^{J}|\beta_j|$$

- Case 1: If $\beta_j = 0 \rightsquigarrow$ not differentiable. But $\beta_j = 0$
- Case 2: If $\beta_j > (<) 0 \rightsquigarrow$ differentiable $\rightsquigarrow$ differentiate and solve for β_j

Define $\tilde{y}_i^j = \beta_0 + \sum_{l \neq j} x_{il}\beta_l$
$r^j \equiv \frac{1}{N}\sum_{i=1}^{N} x_{ij}(y_i - \tilde{y}_i^j)$
Update step for β_j is

$$\beta_j \leftarrow \text{sign}(r^j)\max(|r^j| - \lambda, 0)$$

Lasso Regression$\rightsquigarrow$ Intuition 1, Soft Thresholding

Suppose again $\boldsymbol{X}^{'}\boldsymbol{X} = \boldsymbol{I}_J$

Lasso Regression$\rightsquigarrow$ Intuition 1, Soft Thresholding

Suppose again $\boldsymbol{X}^{'}\boldsymbol{X} = \boldsymbol{I}_J$

$$f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) \;\; = \;\; (Y - \boldsymbol{X}\boldsymbol{\beta})^{'}(Y - \boldsymbol{X}\boldsymbol{\beta}) + \lambda \sum_{j=1}^{J} |\beta_j|$$

Lasso Regression$\rightsquigarrow$ Intuition 1, Soft Thresholding

Suppose again $\boldsymbol{X}^{'}\boldsymbol{X} = \boldsymbol{I}_J$

$$\begin{aligned} f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) &= (Y - \boldsymbol{X}\boldsymbol{\beta})^{'}(Y - \boldsymbol{X}\boldsymbol{\beta}) + \lambda \sum_{j=1}^{J} |\beta_j| \\ &= -2\boldsymbol{X}^{'}\boldsymbol{Y}\boldsymbol{\beta} + \boldsymbol{\beta}^{'}\boldsymbol{\beta} + \lambda \sum_{j=1}^{J} |\beta_j| \end{aligned}$$

Lasso Regression$\rightsquigarrow$ Intuition 1, Soft Thresholding

Suppose again $\boldsymbol{X}^{'}\boldsymbol{X} = \boldsymbol{I}_J$

$$\begin{aligned} f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) &= (Y - \boldsymbol{X}\boldsymbol{\beta})^{'}(Y - \boldsymbol{X}\boldsymbol{\beta}) + \lambda \sum_{j=1}^{J} |\beta_j| \\ &= -2\boldsymbol{X}^{'}\boldsymbol{Y}\boldsymbol{\beta} + \boldsymbol{\beta}^{'}\boldsymbol{\beta} + \lambda \sum_{j=1}^{J} |\beta_j| \end{aligned}$$

The coefficient is

Lasso Regression$\rightsquigarrow$ Intuition 1, Soft Thresholding

Suppose again $\boldsymbol{X}^{'}\boldsymbol{X} = \boldsymbol{I}_J$

$$\begin{aligned} f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) &= (Y - \boldsymbol{X}\boldsymbol{\beta})^{'}(Y - \boldsymbol{X}\boldsymbol{\beta}) + \lambda\sum_{j=1}^{J}|\beta_j| \\ &= -2\boldsymbol{X}^{'}\boldsymbol{Y}\boldsymbol{\beta} + \boldsymbol{\beta}^{'}\boldsymbol{\beta} + \lambda\sum_{j=1}^{J}|\beta_j| \end{aligned}$$

The coefficient is

$$\beta_j^{\text{LASSO}} = \text{sign}\left(\widehat{\beta}_j\right)\left(|\widehat{\beta}_j| - \lambda\right)_{+}$$

Lasso Regression$\rightsquigarrow$ Intuition 1, Soft Thresholding

Suppose again $\boldsymbol{X}^{'}\boldsymbol{X} = \boldsymbol{I}_J$

$$
\begin{aligned}
f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) &= (Y - \boldsymbol{X}\boldsymbol{\beta})^{'}(Y - \boldsymbol{X}\boldsymbol{\beta}) + \lambda \sum_{j=1}^{J} |\beta_j| \\
&= -2\boldsymbol{X}^{'}\boldsymbol{Y}\boldsymbol{\beta} + \boldsymbol{\beta}^{'}\boldsymbol{\beta} + \lambda \sum_{j=1}^{J} |\beta_j|
\end{aligned}
$$

The coefficient is

$$
\beta_j^{\text{LASSO}} = \text{sign}\left(\widehat{\beta}_j\right)\left(|\widehat{\beta}_j| - \lambda\right)_{+}
$$

- $\text{sign}(\cdot) \rightsquigarrow 1$ or -1

Lasso Regression$\rightsquigarrow$ Intuition 1, Soft Thresholding

Suppose again $\boldsymbol{X}^{'}\boldsymbol{X} = \boldsymbol{I}_J$

$$
\begin{aligned}
f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) &= (Y - \boldsymbol{X}\boldsymbol{\beta})^{'}(Y - \boldsymbol{X}\boldsymbol{\beta}) + \lambda \sum_{j=1}^{J} |\beta_j| \\
&= -2\boldsymbol{X}^{'}\boldsymbol{Y}\boldsymbol{\beta} + \boldsymbol{\beta}^{'}\boldsymbol{\beta} + \lambda \sum_{j=1}^{J} |\beta_j|
\end{aligned}
$$

The coefficient is

$$
\beta_j^{\text{LASSO}} = \text{sign}\left(\widehat{\beta}_j\right)\left(|\widehat{\beta}_j| - \lambda\right)_+
$$

- $\text{sign}(\cdot) \rightsquigarrow 1$ or -1
- $\left(|\widehat{\beta}_j| - \lambda\right)_+ = \max(|\widehat{\beta}_j| - \lambda, 0)$

Lasso Regression$\rightsquigarrow$ Intuition 1, Soft Thresholding

Compare soft assignment

Lasso Regression$\rightsquigarrow$ Intuition 1, Soft Thresholding

Compare soft assignment

$$\beta_j^{\text{LASSO}} \quad = \quad \text{sign}\left(\widehat{\beta}_j\right)\left(|\widehat{\beta}_j| - \lambda\right)_+$$

Lasso Regression⇝ Intuition 1, Soft Thresholding

Compare soft assignment

$$\beta_j^{\text{LASSO}} \quad = \quad \text{sign}\left(\widehat{\beta}_j\right)\left(|\widehat{\beta}_j| - \lambda\right)_+$$

With hard assignment, selecting M biggest components

Lasso Regression$\rightsquigarrow$ Intuition 1, Soft Thresholding

Compare soft assignment

$$\beta_j^{\text{LASSO}} = \text{sign}\left(\widehat{\beta}_j\right)\left(|\widehat{\beta}_j| - \lambda\right)_+$$

With hard assignment, selecting M biggest components

$$\beta_j^{\text{subset}} = \widehat{\beta}_j \cdot I\left(|\widehat{\beta}_j| \geq |\widehat{\beta}_{(M)}|\right)$$

Lasso Regression$\rightsquigarrow$ Intuition 1, Soft Thresholding

Compare soft assignment

$$\beta_j^{\text{LASSO}} \quad = \quad \text{sign}\left(\widehat{\beta}_j\right)\left(|\widehat{\beta}_j| - \lambda\right)_+$$

With hard assignment, selecting M biggest components

$$\beta_j^{\text{subset}} \quad = \quad \widehat{\beta}_j \cdot I\left(|\widehat{\beta}_j| \geq |\widehat{\beta}_{(M)}|\right)$$

Intuition 2: Prior on coefficients $\rightsquigarrow$ Laplace “The Bayesian LASSO”

Lasso Regression$\rightsquigarrow$ Intuition 1, Soft Thresholding

Compare soft assignment

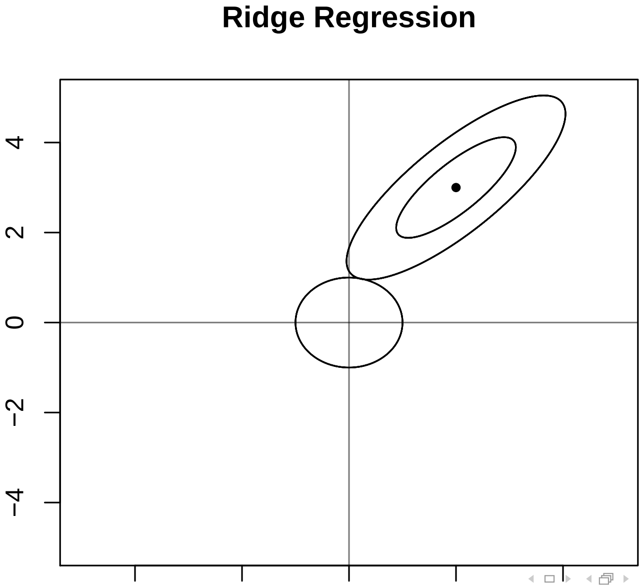
$$\beta_j^{\text{LASSO}} \quad = \quad \text{sign}\left(\widehat{\beta}_j\right)\left(|\widehat{\beta}_j| - \lambda\right)_+$$

With hard assignment, selecting M biggest components

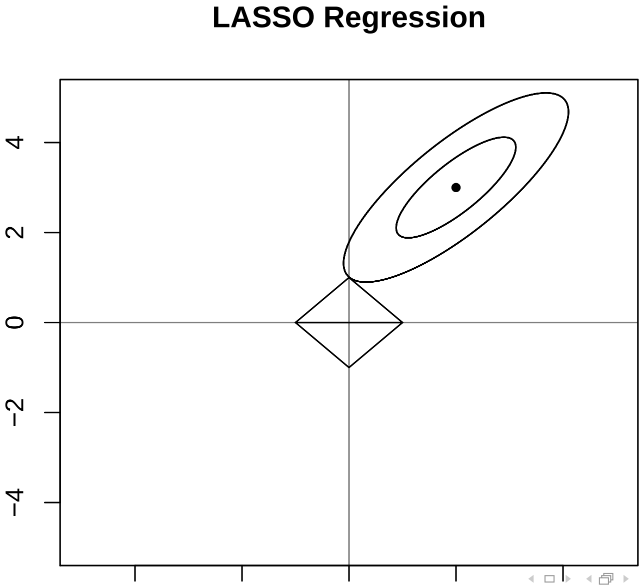
$$\beta_j^{\text{subset}} \quad = \quad \widehat{\beta}_j \cdot I\left(|\widehat{\beta}_j| \geq |\widehat{\beta}_{(M)}|\right)$$

Intuition 2: Prior on coefficients $\rightsquigarrow$ Laplace "The Bayesian LASSO"
Why does LASSO induce sparsity?

Comparing Ridge and LASSO



Comparing Ridge and LASSO



Comparing Ridge and LASSO

Contrast $\beta = (\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$ and $\tilde{\beta} = (1, 0)$

Comparing Ridge and LASSO

Contrast $\beta = (\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$ and $\tilde{\beta} = (1, 0)$

Under ridge:

Comparing Ridge and LASSO

Contrast $\beta = (\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$ and $\tilde{\beta} = (1, 0)$

Under ridge:

$$\sum_{j=1}^{2} \beta_j^2 \quad = \quad \frac{1}{2} + \frac{1}{2} = 1$$

Comparing Ridge and LASSO

Contrast $\beta = (\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$ and $\tilde{\beta} = (1, 0)$

Under ridge:

$$\sum_{j=1}^{2} \beta_j^2 = \frac{1}{2} + \frac{1}{2} = 1$$

$$\sum_{j=1}^{2} \tilde{\beta}_j^2 = 1 + 0 = 1$$

Comparing Ridge and LASSO

Contrast $\beta = (\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$ and $\tilde{\beta} = (1, 0)$

Under ridge:

$$\begin{aligned}
\sum_{j=1}^{2} \beta_j^2 &= \frac{1}{2} + \frac{1}{2} = 1 \\
\sum_{j=1}^{2} \tilde{\beta}_j^2 &= 1 + 0 = 1
\end{aligned}$$

Under LASSO

Comparing Ridge and LASSO

Contrast $\beta = (\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$ and $\tilde{\beta} = (1, 0)$

Under ridge:

$$\begin{aligned}
\sum_{j=1}^{2} \beta_j^2 &= \frac{1}{2} + \frac{1}{2} = 1 \\
\sum_{j=1}^{2} \tilde{\beta}_j^2 &= 1 + 0 = 1
\end{aligned}$$

Under LASSO

$$\sum_{j=1}^{2} |\beta_j| = \frac{1}{\sqrt{2}} + \frac{1}{\sqrt{2}} = \sqrt{2}$$

Comparing Ridge and LASSO

Contrast $\beta = (\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$ and $\tilde{\beta} = (1, 0)$

Under ridge:

$$\begin{aligned} \sum_{j=1}^{2} \beta_j^2 &= \frac{1}{2} + \frac{1}{2} = 1 \\ \sum_{j=1}^{2} \tilde{\beta}_j^2 &= 1 + 0 = 1 \end{aligned}$$

Under LASSO

$$\begin{aligned} \sum_{j=1}^{2} |\beta_j| &= \frac{1}{\sqrt{2}} + \frac{1}{\sqrt{2}} = \sqrt{2} \\ \sum_{j=1}^{2} |\tilde{\beta}_j| &= 1 + 0 = 1 \end{aligned}$$

Ridge and LASSO: The Elastic-Net

Combining the two criteria ⇝ Elastic-Net

$$f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) \quad = \quad \frac{1}{2N} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{J} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{J} \left(\frac{1}{2} (1 - \alpha) \beta_j^2 + \alpha |\beta_j| \right)$$

Ridge and LASSO: The Elastic-Net

Combining the two criteria $\rightsquigarrow$ Elastic-Net

$$f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) \quad = \quad \frac{1}{2N}\sum_{i=1}^{N}\left(y_i - \beta_0 - \sum_{j=1}^{J}\beta_j x_{ij}\right)^2 + \lambda \sum_{j=1}^{J}\left(\frac{1}{2}(1-\alpha)\beta_j^2 + \alpha|\beta_j|\right)$$

The new update step (for coordinate descent:)

Ridge and LASSO: The Elastic-Net

Combining the two criteria ⇝ Elastic-Net

$$f(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) \quad = \quad \frac{1}{2N} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{J} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{J} \left(\frac{1}{2}(1-\alpha)\beta_j^2 + \alpha|\beta_j| \right)$$

The new update step (for coordinate descent:)

$$\beta_j \quad \leftarrow \quad \frac{\text{sign}(r^j)\text{max}(|r^j| - \lambda\alpha, 0)}{1 + \lambda(1-\alpha)}$$

Selecting λ

How do we determine λ? $\rightsquigarrow$ Cross validation

Selecting λ

How do we determine λ? $\rightsquigarrow$ Cross validation
Applying models gives score (probability) of document belong to class$\rightsquigarrow$ threshold to classify

Selecting λ

How do we determine λ? $\rightsquigarrow$ Cross validation
Applying models gives score (probability) of document belong to class$\rightsquigarrow$ threshold to classify

Loss Functions and Model Complexity

Suppose observations i have dependent variables Y_i and covariates $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iP})$.

Loss Functions and Model Complexity

Suppose observations i have dependent variables Y_i and covariates $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iP})$.
Assume:

$$\begin{aligned} Y_i &\sim \text{Distribution}(\mu_i, \phi) \\ \mu_i &= f(\boldsymbol{\beta}, \boldsymbol{x}_i) \end{aligned}$$

Use MLE to obtain $\hat{\boldsymbol{\beta}}$.

Loss Functions and Model Complexity

Suppose observations i have dependent variables Y_i and covariates $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iP})$.
Assume:

$$\begin{aligned} Y_i &\sim \text{Distribution}(\mu_i, \phi) \\ \mu_i &= f(\boldsymbol{\beta}, \boldsymbol{x}_i) \end{aligned}$$

Use MLE to obtain $\hat{\boldsymbol{\beta}}$.
Potential loss functions:

Loss Functions and Model Complexity

Suppose observations i have dependent variables Y_i and covariates $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iP})$.
Assume:

$$\begin{aligned} Y_i &\sim \text{Distribution}(\mu_i, \phi) \\ \mu_i &= f(\boldsymbol{\beta}, \boldsymbol{x}_i) \end{aligned}$$

Use MLE to obtain $\hat{\boldsymbol{\beta}}$.
Potential loss functions:

$$L\left(Y_i, f(\hat{\boldsymbol{\beta}}, \boldsymbol{x}_i)\right)$$

Loss Functions and Model Complexity

Suppose observations i have dependent variables Y_i and covariates $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iP})$.
Assume:

$$\begin{aligned} Y_i & \sim \text{Distribution}(\mu_i, \phi) \\ \mu_i & = f(\boldsymbol{\beta}, \boldsymbol{x}_i) \end{aligned}$$

Use MLE to obtain $\hat{\boldsymbol{\beta}}$.
Potential loss functions:

$$L\left(Y_i, f(\hat{\boldsymbol{\beta}}, \boldsymbol{x}_i)\right) = \left(Y_i - f(\hat{\boldsymbol{\beta}}, \boldsymbol{x}_i)\right)^2$$

Loss Functions and Model Complexity

Suppose observations i have dependent variables Y_i and covariates $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iP})$.
Assume:

$$\begin{aligned} Y_i &\sim \text{Distribution}(\mu_i, \phi) \\ \mu_i &= f(\boldsymbol{\beta}, \boldsymbol{x}_i) \end{aligned}$$

Use MLE to obtain $\hat{\boldsymbol{\beta}}$.
Potential loss functions:

$$\begin{aligned} L\left(Y_i, f(\hat{\boldsymbol{\beta}}, \boldsymbol{x}_i)\right) &= \left(Y_i - f(\hat{\boldsymbol{\beta}}, \boldsymbol{x}_i)\right)^2 \\ &= \left|Y_i - f(\hat{\boldsymbol{\beta}}, \boldsymbol{x}_i)\right| \end{aligned}$$

Loss Functions and Model Complexity

Suppose observations i have dependent variables Y_i and covariates $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iP})$.
Assume:

$$
\begin{aligned}
Y_i &\sim \text{Distribution}(\mu_i, \phi) \\
\mu_i &= f(\boldsymbol{\beta}, \boldsymbol{x}_i)
\end{aligned}
$$

Use MLE to obtain $\hat{\boldsymbol{\beta}}$.
Potential loss functions:

$$
\begin{aligned}
L\left(Y_i, f(\hat{\boldsymbol{\beta}}, \boldsymbol{x}_i)\right) &= \left(Y_i - f(\hat{\boldsymbol{\beta}}, \boldsymbol{x}_i)\right)^2 \\
&= \left|Y_i - f(\hat{\boldsymbol{\beta}}, \boldsymbol{x}_i)\right| \\
&= I\left(Y_i = 1 - I(f(\hat{\boldsymbol{\beta}}, \boldsymbol{x}_i) > \tau)\right)
\end{aligned}
$$

Training and Test Sets

The useful “fiction” of training and test sets:

Training and Test Sets

The useful “fiction” of training and test sets:

- Training set: data set used to fit the model

Training and Test Sets

The useful “fiction” of training and test sets:

- Training set: data set used to fit the model
- Test set: data used to evaluate fit of the model

Training and Test Sets

The useful "fiction" of training and test sets:

- Training set: data set used to fit the model
- Test set: data used to evaluate fit of the model

Even if no division, useful to think about systematic components of data.

Loss Functions and Model Complexity

Suppose that we have:

$$=$$

Loss Functions and Model Complexity

Suppose that we have:

- Training sets, $\mathcal{T}$, with $|\mathcal{T}| = N_{\text{train}}$

$$=$$

Loss Functions and Model Complexity

Suppose that we have:

- Training sets, $\mathcal{T}$, with $|\mathcal{T}| = N_{\text{train}}$
- Test sets, $\mathcal{O}$ with $|\mathcal{O}| = N_{\text{test}}$

Loss Functions and Model Complexity

Suppose that we have:

- Training sets, $\mathcal{T}$, with $|\mathcal{T}| = N_{\text{train}}$
- Test sets, $\mathcal{O}$ with $|\mathcal{O}| = N_{\text{test}}$

Training (in-sample) error is:

$$=$$

Loss Functions and Model Complexity

Suppose that we have:

- Training sets, $\mathcal{T}$, with $|\mathcal{T}| = N_{\text{train}}$
- Test sets, $\mathcal{O}$ with $|\mathcal{O}| = N_{\text{test}}$

Training (in-sample) error is:

$$\text{Error}_{\text{in}} =$$

Loss Functions and Model Complexity

Suppose that we have:

- Training sets, $\mathcal{T}$, with $|\mathcal{T}| = N_{\text{train}}$
- Test sets, $\mathcal{O}$ with $|\mathcal{O}| = N_{\text{test}}$

Training (in-sample) error is:

$$\text{Error}_{\text{in}} = \sum_{i \in \mathcal{T}} \frac{1}{N_{\text{train}}} L(Y_i, f(\hat{\boldsymbol{\beta}}, \boldsymbol{x}_i))$$

Loss Functions and Model Complexity

Suppose that we have:

- Training sets, $\mathcal{T}$, with $|\mathcal{T}| = N_{\text{train}}$
- Test sets, $\mathcal{O}$ with $|\mathcal{O}| = N_{\text{test}}$

Training (in-sample) error is:

$$\text{Error}_{\text{in}} = \sum_{i \in \mathcal{T}} \frac{1}{N_{\text{train}}} L(Y_i, f(\hat{\boldsymbol{\beta}}, \boldsymbol{x}_i))$$

We'd like to estimate out of sample performance with

Loss Functions and Model Complexity

Suppose that we have:

- Training sets, $\mathcal{T}$, with $|\mathcal{T}| = N_{\text{train}}$
- Test sets, $\mathcal{O}$ with $|\mathcal{O}| = N_{\text{test}}$

Training (in-sample) error is:

$$\text{Error}_{\text{in}} = \sum_{i \in \mathcal{T}} \frac{1}{N_{\text{train}}} L(Y_i, f(\hat{\boldsymbol{\beta}}, \boldsymbol{x}_i))$$

We'd like to estimate out of sample performance with

$$\text{Error}_{\text{out}} = \text{E}[L(\boldsymbol{Y}_{i \in \mathcal{O}}, f(\hat{\boldsymbol{\beta}}, \boldsymbol{x}_{i \in \mathcal{O}}))|\mathcal{T}]$$

Loss Functions and Model Complexity

Suppose that we have:

- Training sets, $\mathcal{T}$, with $|\mathcal{T}| = N_{\text{train}}$
- Test sets, $\mathcal{O}$ with $|\mathcal{O}| = N_{\text{test}}$

Training (in-sample) error is:

$$\text{Error}_{\text{in}} = \sum_{i \in \mathcal{T}} \frac{1}{N_{\text{train}}} L(Y_i, f(\hat{\beta}, \boldsymbol{x}_i))$$

We'd like to estimate out of sample performance with

$$\text{Error}_{\text{out}} = \text{E}[L(\boldsymbol{Y}_{i \in \mathcal{O}}, f(\hat{\beta}, \boldsymbol{x}_{i \in \mathcal{O}}))|\mathcal{T}]$$

where the expectation is taken over samples for test sets and supposes we have a training set.

Loss Functions and Model Complexity

Suppose that we have:

- Training sets, $\mathcal{T}$, with $|\mathcal{T}| = N_{\text{train}}$
- Test sets, $\mathcal{O}$ with $|\mathcal{O}| = N_{\text{test}}$

Training (in-sample) error is:

$$\text{Error}_{\text{in}} = \sum_{i \in \mathcal{T}} \frac{1}{N_{\text{train}}} L(Y_i, f(\hat{\beta}, \boldsymbol{x}_i))$$

We'd like to estimate out of sample performance with

$$\text{Error}_{\text{out}} = \text{E}[L(\boldsymbol{Y}_{i \in \mathcal{O}}, f(\hat{\beta}, \boldsymbol{x}_{i \in \mathcal{O}}))|\mathcal{T}]$$

where the expectation is taken over samples for test sets and supposes we have a training set.

$$\text{Error} = \text{E}\left[\text{E}[L(\boldsymbol{Y}, f(\hat{\beta}, \boldsymbol{X}))|\mathcal{T}]\right]$$

Loss Functions and Model Complexity

Suppose $Y_i = f(\boldsymbol{x}_i) + \epsilon_i$

Loss Functions and Model Complexity

Suppose $Y_i = f(\boldsymbol{x}_i) + \epsilon_i$
Where $E[\epsilon_i] = 0$

Loss Functions and Model Complexity

Suppose $Y_i = f(\boldsymbol{x}_i) + \epsilon_i$
Where $E[\epsilon_i] = 0$
$\text{var}(\epsilon_i) = \sigma^2_{\epsilon}$

Loss Functions and Model Complexity

Suppose $Y_i = f(\boldsymbol{x}_i) + \epsilon_i$
Where $E[\epsilon_i] = 0$
var$(\epsilon_i) = \sigma^2_{\epsilon}$
Define $f(\hat{\boldsymbol{\beta}}, \boldsymbol{x}) = \hat{f}(\boldsymbol{x})$

Loss Functions and Model Complexity

Suppose $Y_i = f(\boldsymbol{x}_i) + \epsilon_i$
Where $E[\epsilon_i] = 0$
$\text{var}(\epsilon_i) = \sigma^2_{\epsilon}$
Define $f(\hat{\boldsymbol{\beta}}, \boldsymbol{x}) = \hat{f}(\boldsymbol{x})$
With squared error loss:

Loss Functions and Model Complexity

Suppose $Y_i = f(\boldsymbol{x}_i) + \epsilon_i$
Where $E[\epsilon_i] = 0$
$\text{var}(\epsilon_i) = \sigma^2_{\epsilon}$
Define $f(\hat{\boldsymbol{\beta}}, \boldsymbol{x}) = \hat{f}(\boldsymbol{x})$
With squared error loss:

$$\text{Error}(\boldsymbol{x}_0) \quad = \quad \text{E}[(Y_i - \hat{f}(\boldsymbol{x}_i))^2 | \boldsymbol{x}_i = \boldsymbol{x}_0]$$

Loss Functions and Model Complexity

Suppose $Y_i = f(\boldsymbol{x}_i) + \epsilon_i$
Where $E[\epsilon_i] = 0$
$\text{var}(\epsilon_i) = \sigma^2_{\epsilon}$
Define $f(\hat{\boldsymbol{\beta}}, \boldsymbol{x}) = \hat{f}(\boldsymbol{x})$
With squared error loss:

$$\begin{aligned}
\text{Error}(\boldsymbol{x}_0) &= \text{E}[(Y_i - \hat{f}(\boldsymbol{x}_i))^2 | \boldsymbol{x}_i = \boldsymbol{x}_0] \\
&= \text{E}[(f(\boldsymbol{x}_i) + \epsilon_i - \hat{f}(\boldsymbol{x}_i))^2 | \boldsymbol{x}_i = \boldsymbol{x}_0]
\end{aligned}$$

Loss Functions and Model Complexity

Suppose $Y_i = f(\boldsymbol{x}_i) + \epsilon_i$
Where $E[\epsilon_i] = 0$
$\text{var}(\epsilon_i) = \sigma^2_\epsilon$
Define $f(\hat{\boldsymbol{\beta}}, \boldsymbol{x}) = \hat{f}(\boldsymbol{x})$
With squared error loss:

$$\begin{aligned}
\text{Error}(\boldsymbol{x}_0) &= \text{E}[(Y_i - \hat{f}(\boldsymbol{x}_i))^2 | \boldsymbol{x}_i = \boldsymbol{x}_0] \\
&= \text{E}[(f(\boldsymbol{x}_i) + \epsilon_i - \hat{f}(\boldsymbol{x}_i))^2 | \boldsymbol{x}_i = \boldsymbol{x}_0] \\
&= \sigma^2_\epsilon + \left[f(\boldsymbol{x}_0) - \text{E}[\hat{f}(\boldsymbol{x}_0)]\right]^2 + E[\left(\hat{f}(\boldsymbol{x}_0) - E[\hat{f}(\boldsymbol{x}_0)]\right)^2]
\end{aligned}$$

Loss Functions and Model Complexity

Suppose $Y_i = f(\boldsymbol{x}_i) + \epsilon_i$
Where $E[\epsilon_i] = 0$
$\text{var}(\epsilon_i) = \sigma^2_{\epsilon}$
Define $f(\hat{\boldsymbol{\beta}}, \boldsymbol{x}) = \hat{f}(\boldsymbol{x})$
With squared error loss:

$$\begin{aligned}
\text{Error}(\boldsymbol{x}_0) &= \text{E}[(Y_i - \hat{f}(\boldsymbol{x}_i))^2 | \boldsymbol{x}_i = \boldsymbol{x}_0] \\
&= \text{E}[(f(\boldsymbol{x}_i) + \epsilon_i - \hat{f}(\boldsymbol{x}_i))^2 | \boldsymbol{x}_i = \boldsymbol{x}_0] \\
&= \sigma^2_{\epsilon} + \left[f(\boldsymbol{x}_0) - \text{E}[\hat{f}(\boldsymbol{x}_0)]\right]^2 + E[\left(\hat{f}(\boldsymbol{x}_0) - E[\hat{f}(\boldsymbol{x}_0)]\right)^2] \\
&= \text{Irreducible error} + \text{Bias}^2 + \text{Variance}
\end{aligned}$$

Probit Regression (for motivational purposes)

Suppose:

$$\begin{aligned} Y_i &\sim \text{Bernoulli}(\pi_i) \\ \pi_i &= \Phi(\boldsymbol{\beta}^{'}\boldsymbol{x}_i) \end{aligned}$$

where $\Phi(\cdot)$ is the cumulative normal distribution.
Implies log-likelihood

$$\log \text{L}(\boldsymbol{\beta}|\boldsymbol{X}, \boldsymbol{Y}) = \sum_{i=1}^{N} \left[Y_i \log \Phi(\boldsymbol{\beta}^{'}\boldsymbol{x}_i) + (1 - Y_i) \log(1 - \Phi(\boldsymbol{\beta}^{'}\boldsymbol{x}_i)) \right]$$

Log-likelihood is a loss function $\rightsquigarrow$ overly optimistic: improves with more parameters

How Do We Build A Model?

There are many ways to fit models

And many choices made when performing model fit

How do we choose?

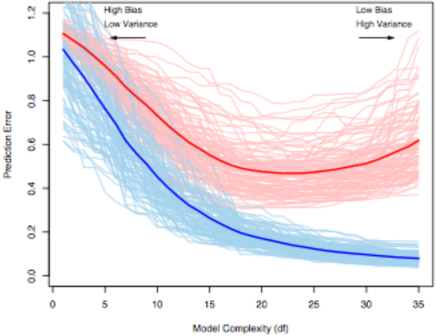


FIGURE 7.1. *Behavior of test sample and training sample error as the model complexity is varied. The light blue curves show the training error* $\overline{\text{err}}$*, while the light red curves show the conditional test error* $\text{Err}_{\mathcal{T}}$ *for 100 training sets of size 50 each, as the model complexity is increased. The solid curves show the expected test error* Err *and the expected training error* $\text{E}[\overline{\text{err}}]$.

How Do We Build A Model?

There are many ways to fit models

And many choices made when performing model fit

How do we choose?

Bad way to choose:

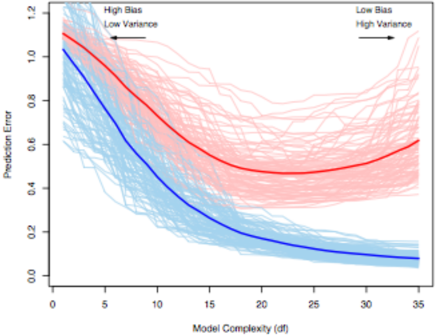


FIGURE 7.1. *Behavior of test sample and training sample error as the model complexity is varied. The light blue curves show the training error* $\overline{\mathrm{err}}$*, while the light red curves show the conditional test error* $\mathrm{Err}_{\mathcal{T}}$ *for 100 training sets of size 50 each, as the model complexity is increased. The solid curves show the expected test error* Err *and the expected training error* $\mathrm{E}[\overline{\mathrm{err}}]$.

How Do We Build A Model?

There are many ways to fit models

And many choices made when performing model fit

How do we choose?

Bad way to choose: within sample model fit (HTF Figure 7.1)

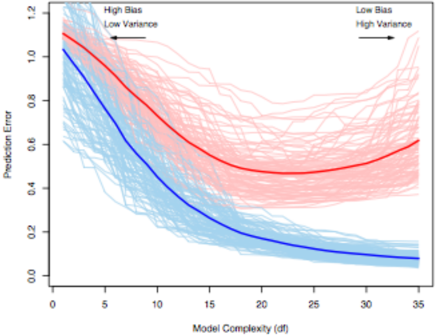


FIGURE 7.1. *Behavior of test sample and training sample error as the model complexity is varied. The light blue curves show the training error* $\overline{\text{err}}$*, while the light red curves show the conditional test error* $\text{Err}_{\mathcal{T}}$ *for 100 training sets of size 50 each, as the model complexity is increased. The solid curves show the expected test error* Err *and the expected training error* $\text{E}[\overline{\text{err}}]$.

How Do We Build A Model?

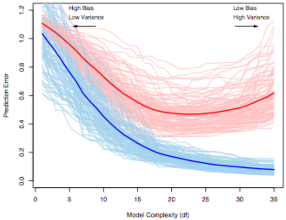


FIGURE 7.1. *Behavior of test sample and training sample error as the model complexity is varied. The light blue curves show the training error* $\overline{\text{err}}$*, while the light red curves show the conditional test error* $\text{Err}_{\mathcal{T}}$ *for 100 training sets of size 50 each, as the model complexity is increased. The solid curves show the expected test error* Err *and the expected training error* $\text{E}[\overline{\text{err}}]$.

Model overfit⇝ in sample error is optimistic:

How Do We Build A Model?

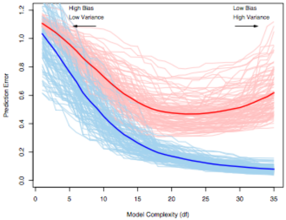


FIGURE 7.1. *Behavior of test sample and training sample error as the model complexity is varied. The light blue curves show the training error* $\overline{\text{err}}$*, while the light red curves show the conditional test error* $\text{Err}_{\mathcal{T}}$ *for 100 training sets of size 50 each, as the model complexity is increased. The solid curves show the expected test error* Err *and the expected training error* $\text{E}[\overline{\text{err}}]$.

Model overfit⇝ in sample error is optimistic:

- Some model complexity captures systematic features of the data

How Do We Build A Model?

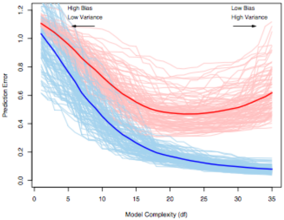


FIGURE 7.1. *Behavior of test sample and training sample error as the model complexity is varied. The light blue curves show the training error* $\overline{\mathrm{err}}$, *while the light red curves show the conditional test error* $\mathrm{Err}_{\mathcal{T}}$ *for 100 training sets of size 50 each, as the model complexity is increased. The solid curves show the expected test error* Err *and the expected training error* $\mathrm{E}[\overline{\mathrm{err}}]$.

Model overfit⇝ in sample error is optimistic:

- Some model complexity captures systematic features of the data
- Characteristics found in both training and test set

How Do We Build A Model?

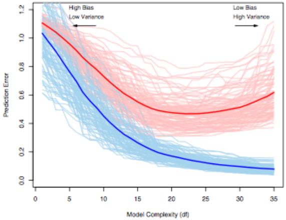


FIGURE 7.1. *Behavior of test sample and training sample error as the model complexity is varied. The light blue curves show the training error* $\overline{\text{err}}$*, while the light red curves show the conditional test error* $\text{Err}_{\mathcal{T}}$ *for 100 training sets of size 50 each, as the model complexity is increased. The solid curves show the expected test error* Err *and the expected training error* $\text{E}[\overline{\text{err}}]$.

Model overfit⇝ in sample error is optimistic:

- Some model complexity captures systematic features of the data
- Characteristics found in both training and test set
- Reduces error in both training and test set

How Do We Build A Model?

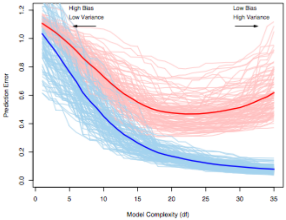


FIGURE 7.1. *Behavior of test sample and training sample error as the model complexity is varied. The light blue curves show the training error* $\overline{\mathrm{err}}$, *while the light red curves show the conditional test error* $\mathrm{Err}_{\mathcal{T}}$ *for 100 training sets of size 50 each, as the model complexity is increased. The solid curves show the expected test error* Err *and the expected training error* $\mathrm{E}[\overline{\mathrm{err}}]$.

Model overfit⇝ in sample error is optimistic:

- Some model complexity captures systematic features of the data
- Characteristics found in both training and test set
- Reduces error in both training and test set
- Additional model complexity: idiosyncratic features of the training set

How Do We Build A Model?

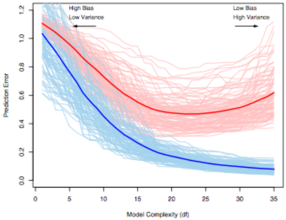


FIGURE 7.1. *Behavior of test sample and training sample error as the model complexity is varied. The light blue curves show the training error* $\overline{\mathrm{err}}$*, while the light red curves show the conditional test error* $\mathrm{Err}_{\mathcal{T}}$ *for* 100 *training sets of size* 50 *each, as the model complexity is increased. The solid curves show the expected test error* Err *and the expected training error* $\mathrm{E}[\overline{\mathrm{err}}]$.

Model overfit⇝ in sample error is optimistic:

- Some model complexity captures systematic features of the data
- Characteristics found in both training and test set
- Reduces error in both training and test set
- Additional model complexity: idiosyncratic features of the training set
- Reduces error in training set, increases error in test set

How Do We Choose Covariates?

Best model depends on task

- Causal inference observational study: make treatment assignment ignorable
- Prediction: improve predictive performance

Stepwise Regression

Suppose we have P covariates.
2^P potential models

Stepwise Regression

Suppose we have P covariates.
2^P potential models
Stepwise procedures

Stepwise Regression

Suppose we have P covariates.

2^P potential models

Stepwise procedures

1) Forward selection
    a) No variables in model.
    b) Check all variables p-value if include, include lowest p-value
    c) Repeat until included p-value is above some threshold

Stepwise Regression

Suppose we have P covariates.

2^P potential models

Stepwise procedures

1) Forward selection
   a) No variables in model.
   b) Check all variables p-value if include, include lowest p-value
   c) Repeat until included p-value is above some threshold
2) Backward elimination
   a) Fit model with all variables (if possible)
   b) Remove variable with largest p-value
   c) Repeat until potentially excluded p-value is below some threshold

Stepwise Regression

Suppose we have P covariates.

2^P potential models

Stepwise procedures

1) Forward selection
    a) No variables in model.
    b) Check all variables p-value if include, include lowest p-value
    c) Repeat until included p-value is above some threshold
2) Backward elimination
    a) Fit model with all variables (if possible)
    b) Remove variable with largest p-value
    c) Repeat until potentially excluded p-value is below some threshold

Problematic:

1) Not optimal model selection (path dependent)
2) P-value $\neq$ objective of model

Analytic Solutions

Approximate optimism and compensate in loss function.

Analytic Solutions

Approximate optimism and compensate in loss function.
Akaike Information Criterion (AIC) $\rightsquigarrow$ Minimize

Analytic Solutions

Approximate optimism and compensate in loss function.

Akaike Information Criterion (AIC) $\rightsquigarrow$ Minimize

As $N \rightarrow \infty$

Analytic Solutions

Approximate optimism and compensate in loss function.

Akaike Information Criterion (AIC) $\rightsquigarrow$ Minimize

As $N \to \infty$

$$-2\text{E}[\log P_{\hat{\boldsymbol{\beta}}}(Y)] = -2\left[\text{E}[\log \text{L}(\hat{\beta}|\boldsymbol{X}, \boldsymbol{Y})] - d\right]$$

Analytic Solutions

Approximate optimism and compensate in loss function.
Akaike Information Criterion (AIC) $\rightsquigarrow$ Minimize
As $N \rightarrow \infty$

$$
\begin{aligned}
-2\mathrm{E}[\log P_{\hat{\boldsymbol{\beta}}}(Y)] &= -2\left[\mathrm{E}[\log \mathrm{L}(\hat{\beta}|\boldsymbol{X}, \boldsymbol{Y})] - d\right] \\
\mathrm{AIC} &= -2\left[\log \mathrm{L}(\hat{\beta}|\boldsymbol{X}, \boldsymbol{Y}) - d\right]
\end{aligned}
$$

Analytic Solutions

Approximate optimism and compensate in loss function.
Akaike Information Criterion (AIC) $\rightsquigarrow$ Minimize
As $N \to \infty$

$$
\begin{aligned}
-2\mathrm{E}[\log P_{\hat{\boldsymbol{\beta}}}(Y)] &= -2\left[\mathrm{E}[\log \mathrm{L}(\hat{\boldsymbol{\beta}}|\boldsymbol{X}, \boldsymbol{Y})] - d\right] \\
\mathrm{AIC} &= -2\left[\log \mathrm{L}(\hat{\boldsymbol{\beta}}|\boldsymbol{X}, \boldsymbol{Y}) - d\right]
\end{aligned}
$$

where d is the number of parameters in the model

Analytic Solutions

Approximate optimism and compensate in loss function.
Akaike Information Criterion (AIC) $\rightsquigarrow$ Minimize
As $N \to \infty$

$$
\begin{aligned}
-2\text{E}[\log P_{\hat{\boldsymbol{\beta}}}(Y)] &= -2\left[\text{E}[\log \text{L}(\hat{\beta}|\boldsymbol{X}, \boldsymbol{Y})] - d\right] \\
\text{AIC} &= -2\left[\log \text{L}(\hat{\beta}|\boldsymbol{X}, \boldsymbol{Y}) - d\right]
\end{aligned}
$$

where d is the number of parameters in the model

- Intuition: balances model fit with penalty for complexity

Analytic Solutions

Approximate optimism and compensate in loss function.
Akaike Information Criterion (AIC) $\rightsquigarrow$ Minimize
As $N \to \infty$

$$\begin{aligned} -2\text{E}[\log P_{\hat{\boldsymbol{\beta}}}(Y)] &= -2\left[\text{E}[\log \text{L}(\hat{\beta}|\boldsymbol{X}, \boldsymbol{Y})] - d\right] \\ \text{AIC} &= -2\left[\log \text{L}(\hat{\beta}|\boldsymbol{X}, \boldsymbol{Y}) - d\right] \end{aligned}$$

where d is the number of parameters in the model

- Intuition: balances model fit with penalty for complexity
- Derived from method to estimate optimism in likelihood based models

Analytic Solutions

Approximate optimism and compensate in loss function.
Akaike Information Criterion (AIC) $\rightsquigarrow$ Minimize
As $N \to \infty$

$$
\begin{aligned}
-2\mathrm{E}[\log P_{\hat{\boldsymbol{\beta}}}(Y)] &= -2\left[\mathrm{E}[\log \mathrm{L}(\hat{\boldsymbol{\beta}}|\boldsymbol{X}, \boldsymbol{Y})] - d\right] \\
\mathrm{AIC} &= -2\left[\log \mathrm{L}(\hat{\boldsymbol{\beta}}|\boldsymbol{X}, \boldsymbol{Y}) - d\right]
\end{aligned}
$$

where d is the number of parameters in the model

- Intuition: balances model fit with penalty for complexity
- Derived from method to estimate optimism in likelihood based models
- Derived from a method to compute similarity between estimated model and true model (under assumptions of course)

Analytic Solutions

Approximate optimism and compensate in loss function.
Akaike Information Criterion (AIC) $\rightsquigarrow$ Minimize
As $N \rightarrow \infty$

$$\begin{aligned} -2\text{E}[\log P_{\hat{\boldsymbol{\beta}}}(Y)] &= -2\left[\text{E}[\log \text{L}(\hat{\boldsymbol{\beta}}|\boldsymbol{X}, \boldsymbol{Y})] - d\right] \\ \text{AIC} &= -2\left[\log \text{L}(\hat{\boldsymbol{\beta}}|\boldsymbol{X}, \boldsymbol{Y}) - d\right] \end{aligned}$$

where d is the number of parameters in the model

- Intuition: balances model fit with penalty for complexity
- Derived from method to estimate optimism in likelihood based models
- Derived from a method to compute similarity between estimated model and true model (under assumptions of course)
- Can be extended to general models, though requires estimate of irresolvable error

Analytic Solutions

Bayesian Information Criterion (BIC) [Schwarz Criterion]

Analytic Solutions

Bayesian Information Criterion (BIC) [Schwarz Criterion]

$$\text{BIC} = -2 \log \text{L}(\widehat{\beta}|\boldsymbol{X}, \boldsymbol{Y}) + (\log N)d$$

Analytic Solutions

Bayesian Information Criterion (BIC) [Schwarz Criterion]

$$\text{BIC} = -2 \log \text{L}(\widehat{\beta}|\boldsymbol{X}, \boldsymbol{Y}) + (\log N)d$$

where d is again the effective number of parameters

Analytic Solutions

Bayesian Information Criterion (BIC) [Schwarz Criterion]

$$\text{BIC} \quad = \quad -2 \log \mathrm{L}(\widehat{\beta} | \boldsymbol{X}, \boldsymbol{Y}) + (\log N) d$$

where d is again the effective number of parameters

- Intuition: balances model fit with penalty for complexity

Analytic Solutions

Bayesian Information Criterion (BIC) [Schwarz Criterion]

$$\text{BIC} \quad = \quad -2 \log \text{L}(\widehat{\beta}|\boldsymbol{X}, \boldsymbol{Y}) + (\log N)d$$

where d is again the effective number of parameters

- Intuition: balances model fit with penalty for complexity
- Derived from Bayesian approach to model selection

Analytic Solutions

Bayesian Information Criterion (BIC) [Schwarz Criterion]

$$\text{BIC} \quad = \quad -2 \log \text{L}(\widehat{\beta} | \boldsymbol{X}, \boldsymbol{Y}) + (\log N) d$$

where d is again the effective number of parameters

- Intuition: balances model fit with penalty for complexity
- Derived from Bayesian approach to model selection
- Approximation to Bayes' factor

Analytic Solutions

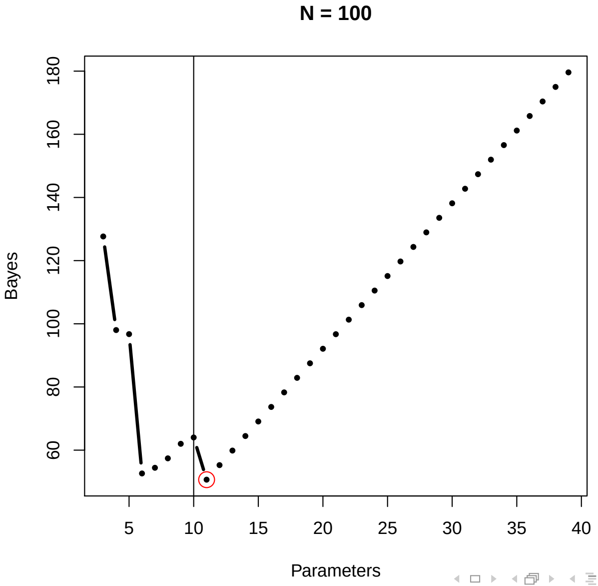
Bayesian Information Criterion (BIC) [Schwarz Criterion]

$$\text{BIC} = -2\log \text{L}(\widehat{\beta}|\boldsymbol{X}, \boldsymbol{Y}) + (\log N)d$$

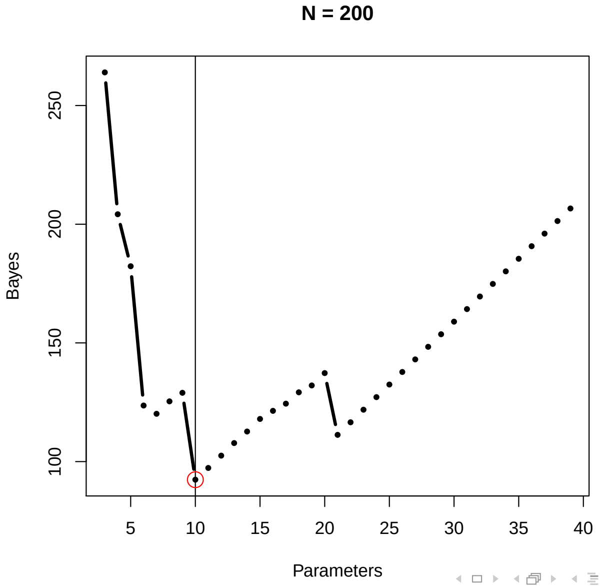
where d is again the effective number of parameters

- Intuition: balances model fit with penalty for complexity
- Derived from Bayesian approach to model selection
- Approximation to Bayes' factor
- Penalizes more heavily than AIC

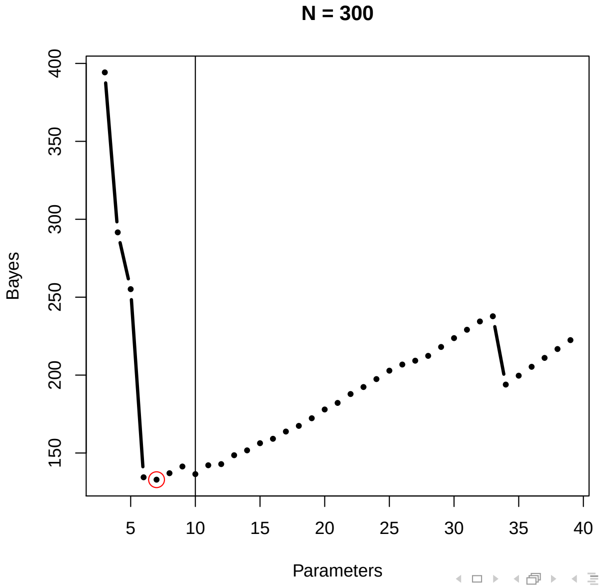
BIC or AIC?


N = 100
Bayes
60
80
100
120
140
160
180
5
10
15
20
25
30
35
40
Parameters

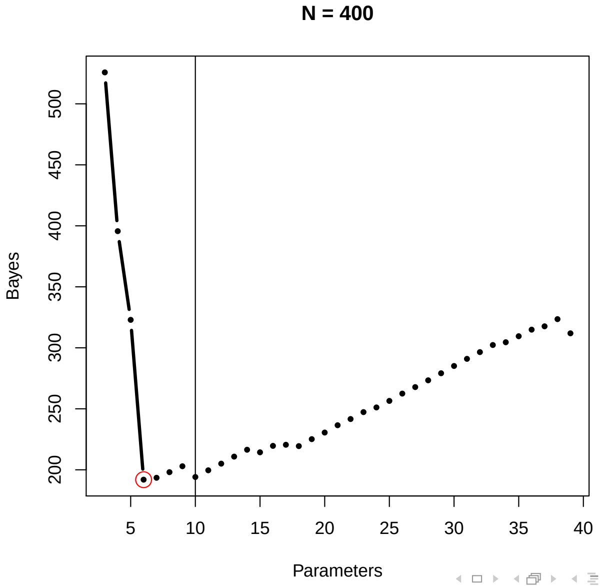
BIC or AIC?



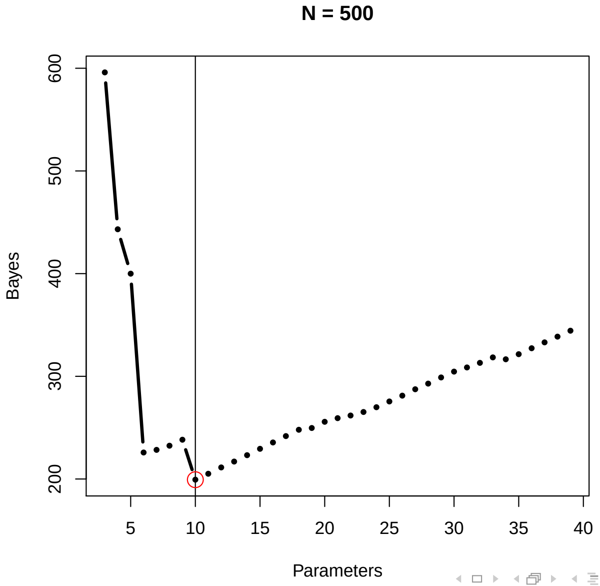
BIC or AIC?


N = 300
Bayes
400
350
300
250
200
150
Parameters
5
10
15
20
25
30
35
40

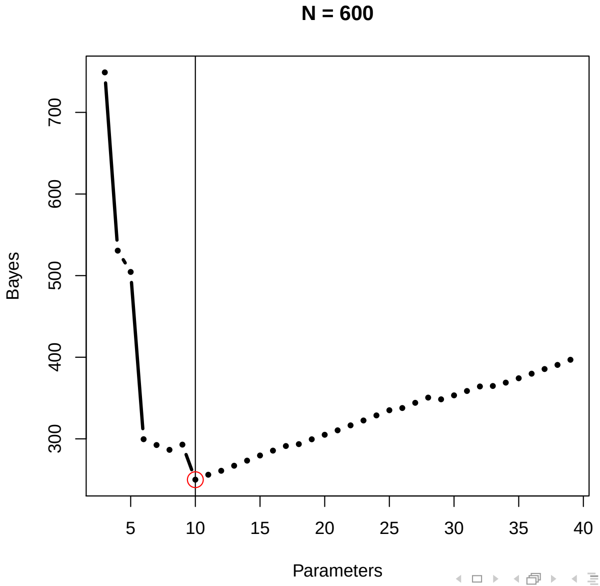
BIC or AIC?


N = 400
Bayes
Parameters
200
250
300
350
400
450
500
5
10
15
20
25
30
35
40

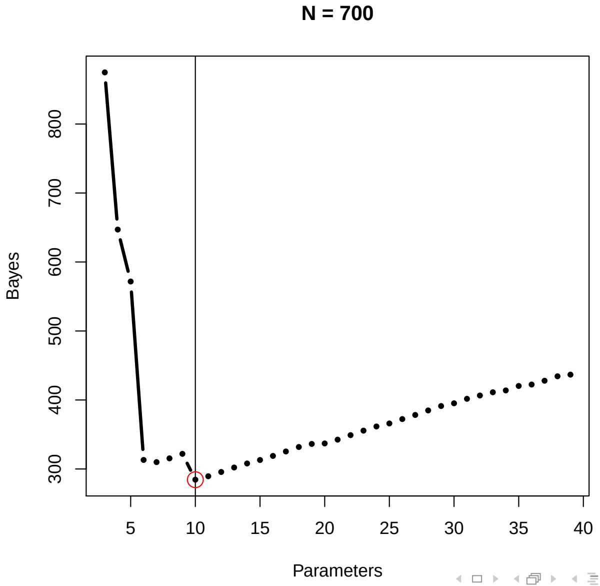
BIC or AIC?



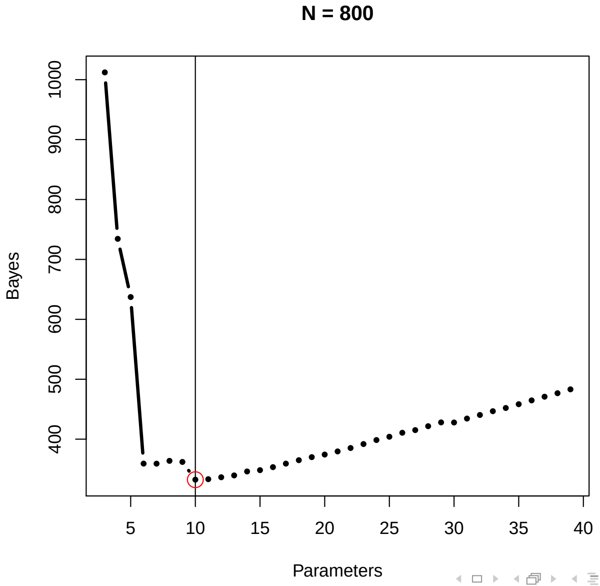
BIC or AIC?



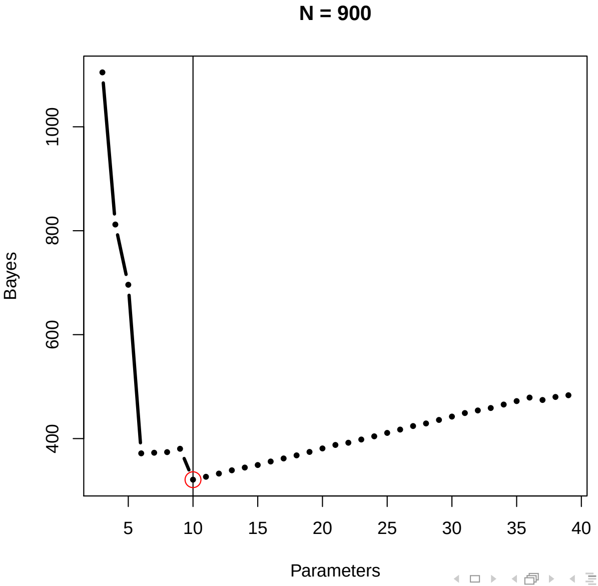
BIC or AIC?


N = 700
Bayes
800
700
600
500
400
300
5
10
15
20
25
30
35
40
Parameters

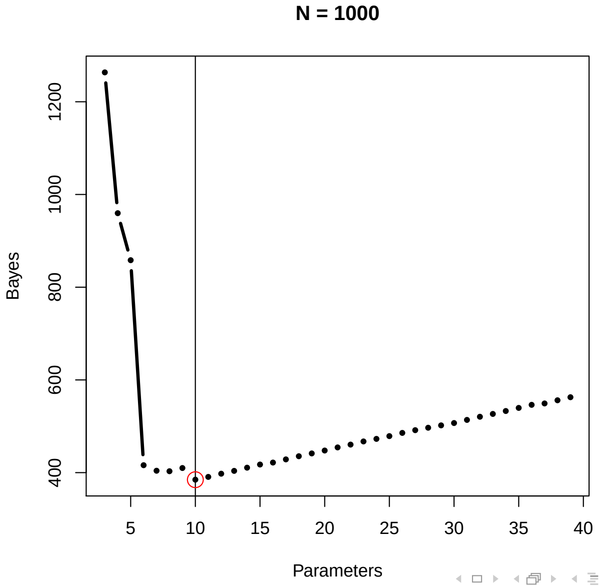
BIC or AIC?


N = 800
Bayes
Parameters
1000
900
800
700
600
500
400
5
10
15
20
25
30
35
40

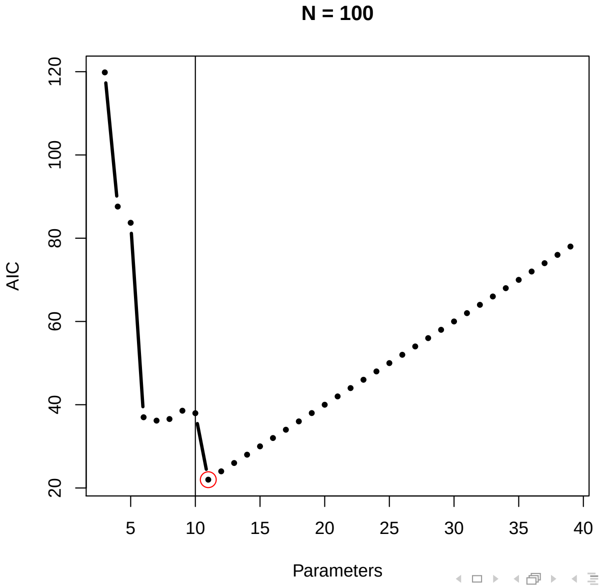
BIC or AIC?



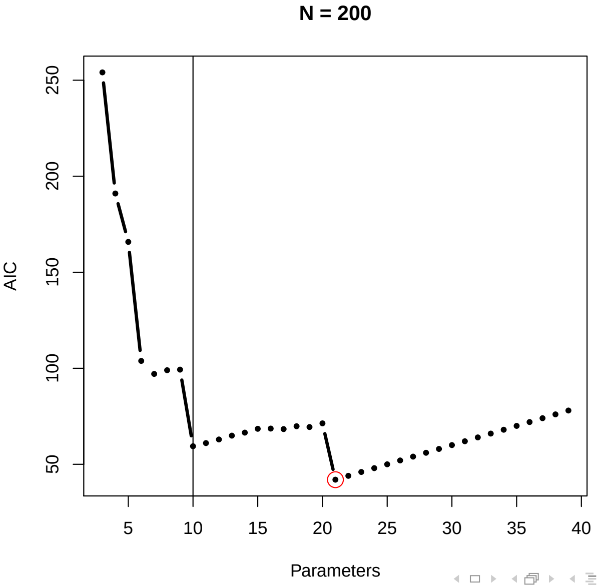
BIC or AIC?


N = 1000
Bayes
Parameters
400
600
800
1000
1200
5
10
15
20
25
30
35
40

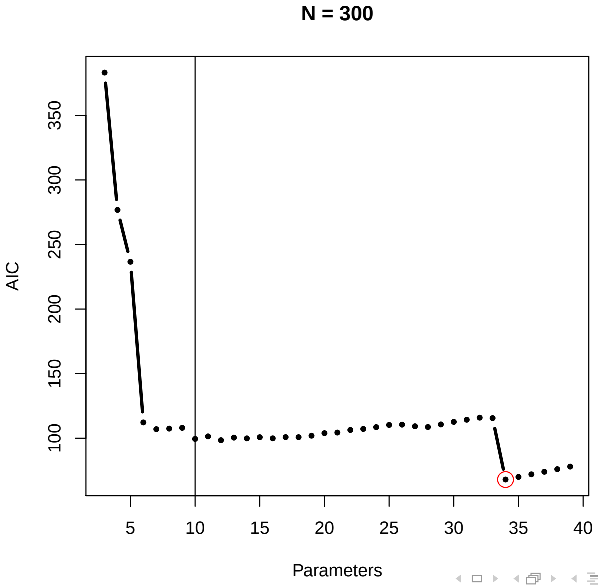
BIC or AIC?



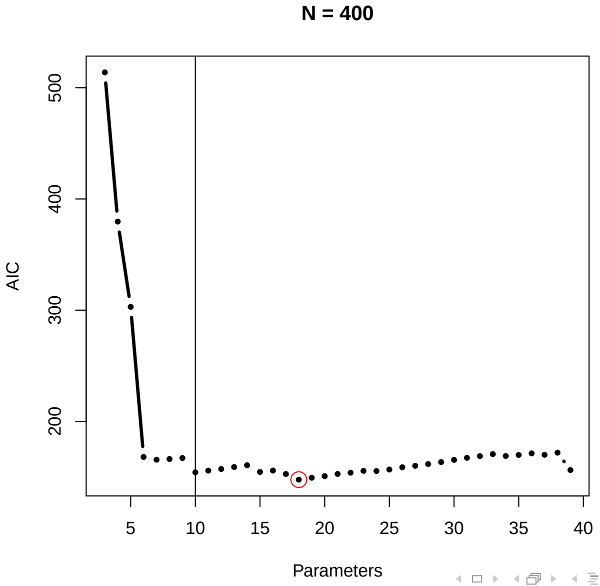
BIC or AIC?



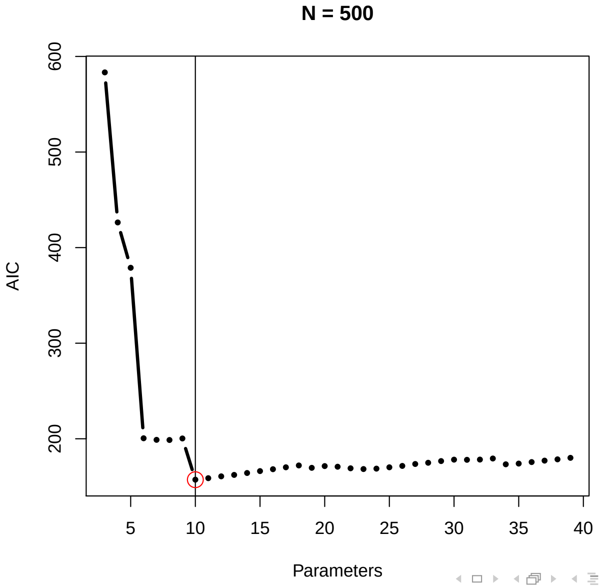
BIC or AIC?


N = 300
AIC
350
300
250
200
150
100
5
10
15
20
25
30
35
40
Parameters

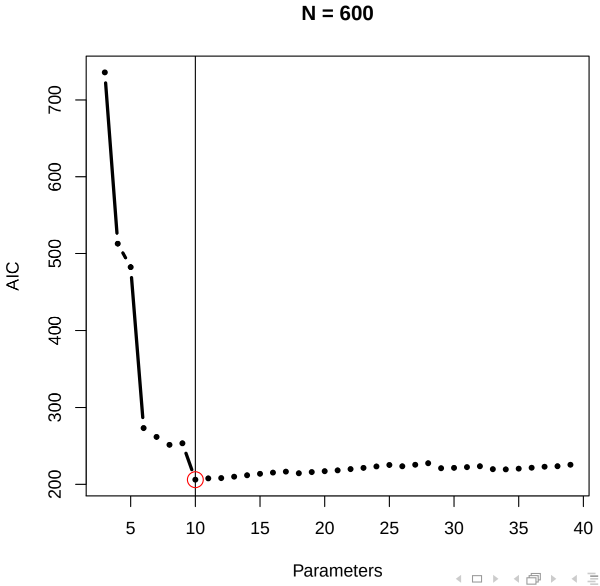
BIC or AIC?



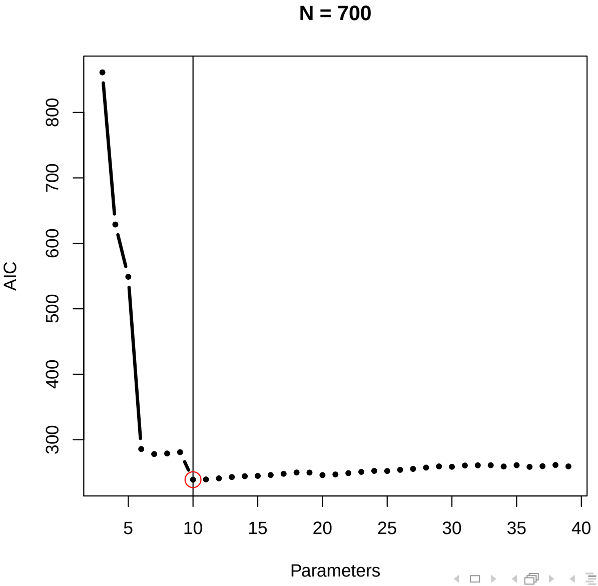
BIC or AIC?



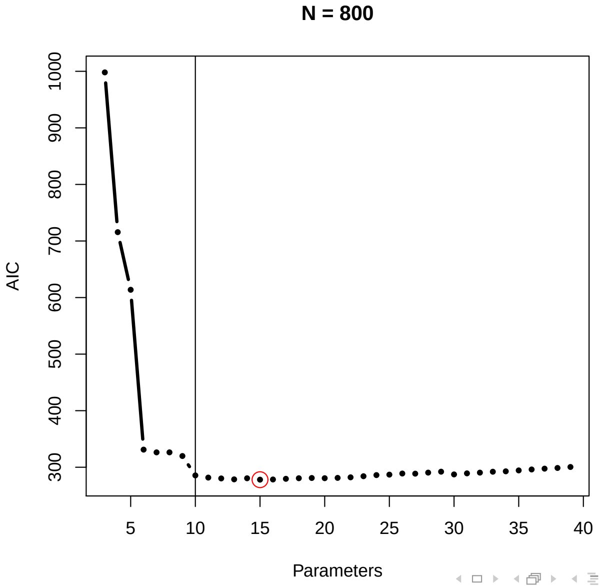
BIC or AIC?


N = 600
AIC
200
300
400
500
600
700
5
10
15
20
25
30
35
40
Parameters

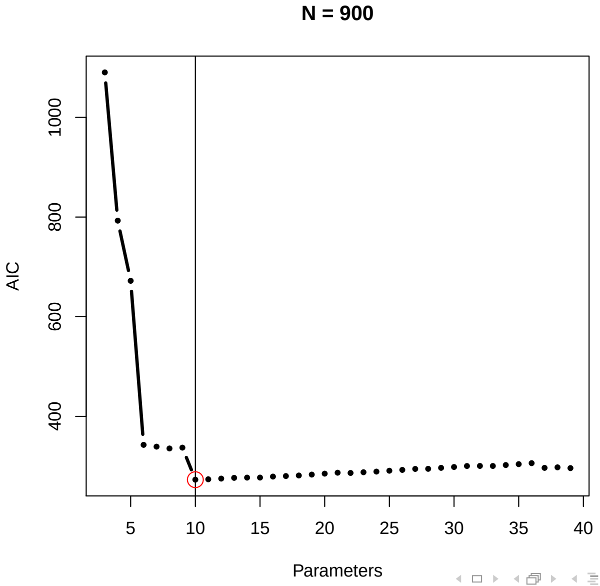
BIC or AIC?


N = 700
AIC
800
700
600
500
400
300
Parameters
5
10
15
20
25
30
35
40

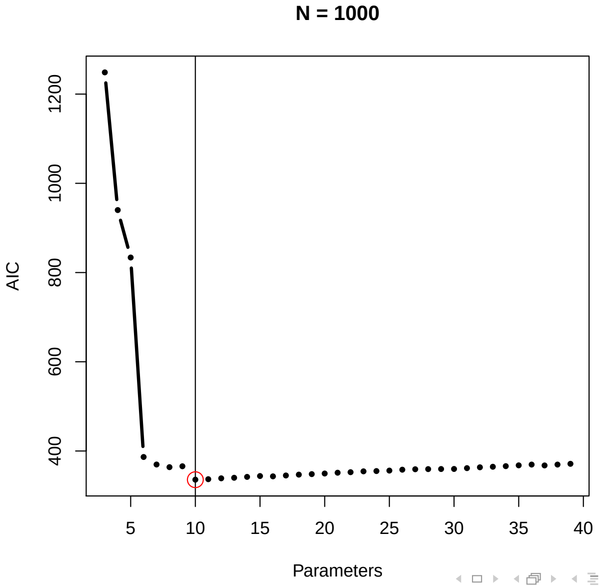
BIC or AIC?


N = 800
AIC
1000
900
800
700
600
500
400
300
5
10
15
20
25
30
35
40
Parameters

BIC or AIC?



BIC or AIC?


N = 1000
AIC
1200
1000
800
600
400
Parameters
5
10
15
20
25
30
35
40

BIC or AIC?

- BIC
  - Asymptotically consistent if true model is in choice set
  - As $N \rightarrow \infty$ will choose correct model with probability 1 (if available)
  - Small samples$\rightsquigarrow$ overpenalize
- AIC
  - No asymptotic guarantees $\rightsquigarrow$ derivation doesn't require truth in set. (KL-criteria)
  - In large samples$\rightsquigarrow$ favors complexity
  - Small samples$\rightsquigarrow$ avoids over penalization

How Do We Select A Model?

Analytic statistics for selection, include penalty for complexity

How Do We Select A Model?

Analytic statistics for selection, include penalty for complexity

- AIC : Akaka Information Criterion

How Do We Select A Model?

Analytic statistics for selection, include penalty for complexity

- AIC : Akaka Information Criterion
- BIC: Bayesian Information Criterion

How Do We Select A Model?

Analytic statistics for selection, include penalty for complexity

- AIC : Akaka Information Criterion
- BIC: Bayesian Information Criterion
- DIC: Deviance Information Criterion

How Do We Select A Model?

Analytic statistics for selection, include penalty for complexity

- AIC : Akaka Information Criterion
- BIC: Bayesian Information Criterion
- DIC: Deviance Information Criterion

Can work well, but...

How Do We Select A Model?

Analytic statistics for selection, include penalty for complexity

- AIC : Akaka Information Criterion
- BIC: Bayesian Information Criterion
- DIC: Deviance Information Criterion

Can work well, but...

- Rely on specific loss function

How Do We Select A Model?

Analytic statistics for selection, include penalty for complexity

- AIC : Akaka Information Criterion
- BIC: Bayesian Information Criterion
- DIC: Deviance Information Criterion

Can work well, but...

- Rely on specific loss function
- Rely on asymptotic argument

How Do We Select A Model?

Analytic statistics for selection, include penalty for complexity

- AIC : Akaka Information Criterion
- BIC: Bayesian Information Criterion
- DIC: Deviance Information Criterion

Can work well, but...

- Rely on specific loss function
- Rely on asymptotic argument
- Rely on estimate of number of parameters

How Do We Select A Model?

Analytic statistics for selection, include penalty for complexity

- AIC : Akaka Information Criterion
- BIC: Bayesian Information Criterion
- DIC: Deviance Information Criterion

Can work well, but...

- Rely on specific loss function
- Rely on asymptotic argument
- Rely on estimate of number of parameters
- Extremely model dependent

How Do We Select A Model?

Analytic statistics for selection, include penalty for complexity

- AIC : Akaka Information Criterion
- BIC: Bayesian Information Criterion
- DIC: Deviance Information Criterion

Can work well, but...

- Rely on specific loss function
- Rely on asymptotic argument
- Rely on estimate of number of parameters
- Extremely model dependent

Need: general tool for evaluating models, replicates decision problem

Cross-Validation: Some Intuition

Optimal division of data for prediction:

Cross-Validation: Some Intuition

Optimal division of data for prediction:

- Train: build model

Cross-Validation: Some Intuition

Optimal division of data for prediction:

- Train: build model
- Validation: assess model

Cross-Validation: Some Intuition

Optimal division of data for prediction:

- Train: build model
- Validation: assess model
- Test: predict remaining data

Cross-Validation: Some Intuition

Optimal division of data for prediction:

- Train: build model
- Validation: assess model
- Test: predict remaining data

K-fold Cross-validation idea: create many training and test sets.

Cross-Validation: Some Intuition

Optimal division of data for prediction:

- Train: build model
- Validation: assess model
- Test: predict remaining data

K-fold Cross-validation idea: create many training and test sets.

- Idea: use observations both in training and test sets

Cross-Validation: Some Intuition

Optimal division of data for prediction:

- Train: build model
- Validation: assess model
- Test: predict remaining data

K-fold Cross-validation idea: create many training and test sets.

- Idea: use observations both in training and test sets
- Each step: use held out data to evaluate performance

Cross-Validation: Some Intuition

Optimal division of data for prediction:

- Train: build model
- Validation: assess model
- Test: predict remaining data

K-fold Cross-validation idea: create many training and test sets.

- Idea: use observations both in training and test sets
- Each step: use held out data to evaluate performance
- Avoid overfitting and have context specific penalty

Cross-Validation: Some Intuition

Optimal division of data for prediction:

- Train: build model
- Validation: assess model
- Test: predict remaining data

K-fold Cross-validation idea: create many training and test sets.

- Idea: use observations both in training and test sets
- Each step: use held out data to evaluate performance
- Avoid overfitting and have context specific penalty

Estimates:

$$\text{Error} \quad = \quad \text{E}\left[\text{E}[L(\boldsymbol{Y}, f(\hat{\beta}, \boldsymbol{X}))|\mathcal{T}]\right]$$

Cross-Validation: A How To Guide

Process:

Cross-Validation: A How To Guide

Process:

- Randomly partition data into K groups.

Cross-Validation: A How To Guide

Process:

- Randomly partition data into K groups.
  (Group 1, Group 2, Group3, . . ., Group K)

Cross-Validation: A How To Guide

Process:

- Randomly partition data into K groups.
  (Group 1, Group 2, Group3, . . ., Group K)
- Rotate through groups as follows

Cross-Validation: A How To Guide

Process:

- Randomly partition data into K groups.
  (Group 1, Group 2, Group3, . . ., Group K)
- Rotate through groups as follows

Step	Training	Validation (“Test”)

Cross-Validation: A How To Guide

Process:

- Randomly partition data into K groups.
  (Group 1, Group 2, Group3, . . ., Group K)
- Rotate through groups as follows

Step	Training	Validation ("Test")
1	Group2, Group3, Group 4, . . ., Group K	Group 1

Cross-Validation: A How To Guide

Process:

- Randomly partition data into K groups.
  (Group 1, Group 2, Group3, . . ., Group K)
- Rotate through groups as follows

Step	Training	Validation ("Test")
1	Group2, Group3, Group 4, . . ., Group K	Group 1
2	Group 1, Group3, Group 4, . . ., Group K	Group 2

Cross-Validation: A How To Guide

Process:

- Randomly partition data into K groups.
  (Group 1, Group 2, Group3, . . ., Group K)
- Rotate through groups as follows

Step	Training	Validation ("Test")
1	Group2, Group3, Group 4, . . ., Group K	Group 1
2	Group 1, Group3, Group 4, . . ., Group K	Group 2
3	Group 1, Group 2, Group 4, . . ., Group K	Group 3

Cross-Validation: A How To Guide

Process:

- Randomly partition data into K groups.
  (Group 1, Group 2, Group3, . . ., Group K)
- Rotate through groups as follows

Step	Training	Validation ("Test")
1	Group2, Group3, Group 4, . . ., Group K	Group 1
2	Group 1, Group3, Group 4, . . ., Group K	Group 2
3	Group 1, Group 2, Group 4, . . ., Group K	Group 3
⋮	⋮	⋮

Cross-Validation: A How To Guide

Process:

- Randomly partition data into K groups.
  (Group 1, Group 2, Group3, . . ., Group K)
- Rotate through groups as follows

Step	Training	Validation ("Test")
1	Group2, Group3, Group 4, . . ., Group K	Group 1
2	Group 1, Group3, Group 4, . . ., Group K	Group 2
3	Group 1, Group 2, Group 4, . . ., Group K	Group 3
⋮	⋮	⋮
K	Group 1, Group 2, Group 3, . . ., Group K - 1	Group K

Cross-Validation: A How To Guide

Step	Training	Validation ("Test")
1	Group2, Group3, Group 4, . . ., Group K	Group 1
2	Group 1, Group3, Group 4, . . ., Group K	Group 2
3	Group 1, Group 2, Group 4, . . ., Group K	Group 3
⋮	⋮	⋮
K	Group 1, Group 2, Group 3, . . ., Group K - 1	Group K

Cross-Validation: A How To Guide

Step	Training	Validation ("Test")
1	Group2, Group3, Group 4, ..., Group K	Group 1
2	Group 1, Group3, Group 4, ..., Group K	Group 2
3	Group 1, Group 2, Group 4, ..., Group K	Group 3
⋮	⋮	⋮
K	Group 1, Group 2, Group 3, ..., Group K - 1	Group K

Strategy:

Cross-Validation: A How To Guide

Step	Training	Validation ("Test")
1	Group2, Group3, Group 4, ..., Group K	Group 1
2	Group 1, Group3, Group 4, ..., Group K	Group 2
3	Group 1, Group 2, Group 4, ..., Group K	Group 3
⋮	⋮	⋮
K	Group 1, Group 2, Group 3, ..., Group K - 1	Group K

Strategy:

- Divide data into K groups

Cross-Validation: A How To Guide

Step	Training	Validation ("Test")
1	Group2, Group3, Group 4, ..., Group K	Group 1
2	Group 1, Group3, Group 4, ..., Group K	Group 2
3	Group 1, Group 2, Group 4, ..., Group K	Group 3
⋮	⋮	⋮
K	Group 1, Group 2, Group 3, ..., Group K - 1	Group K

Strategy:

- Divide data into K groups
- Train data on $K - 1$ groups. Estimate $\hat{f}^{-K}(\boldsymbol{\beta}, \boldsymbol{X})$

Cross-Validation: A How To Guide

Step	Training	Validation ("Test")
1	Group2, Group3, Group 4, ..., Group K	Group 1
2	Group 1, Group3, Group 4, ..., Group K	Group 2
3	Group 1, Group 2, Group 4, ..., Group K	Group 3
⋮	⋮	⋮
K	Group 1, Group 2, Group 3, ..., Group K - 1	Group K

Strategy:

- Divide data into K groups
- Train data on $K-1$ groups. Estimate $\hat{f}^{-K}(\boldsymbol{\beta}, \boldsymbol{X})$
- Predict values for K^{th}

Cross-Validation: A How To Guide

Step	Training	Validation ("Test")
1	Group2, Group3, Group 4, ..., Group K	Group 1
2	Group 1, Group3, Group 4, ..., Group K	Group 2
3	Group 1, Group 2, Group 4, ..., Group K	Group 3
⋮	⋮	⋮
K	Group 1, Group 2, Group 3, ..., Group K - 1	Group K

Strategy:

- Divide data into K groups
- Train data on $K - 1$ groups. Estimate $\hat{f}^{-K}(\boldsymbol{\beta}, \boldsymbol{X})$
- Predict values for K^{th}
- Summarize performance with loss function: $L(\boldsymbol{Y}_i, \hat{f}^{-k}(\boldsymbol{\beta}, \boldsymbol{X}))$

Cross-Validation: A How To Guide

Step	Training	Validation ("Test")
1	Group2, Group3, Group 4, . . ., Group K	Group 1
2	Group 1, Group3, Group 4, . . ., Group K	Group 2
3	Group 1, Group 2, Group 4, . . ., Group K	Group 3
⋮	⋮	⋮
K	Group 1, Group 2, Group 3, . . ., Group K - 1	Group K

Strategy:

- Divide data into K groups
- Train data on $K - 1$ groups. Estimate $\hat{f}^{-K}(\boldsymbol{\beta}, \boldsymbol{X})$
- Predict values for K^{th}
- Summarize performance with loss function: $L(\boldsymbol{Y}_i, \hat{f}^{-k}(\boldsymbol{\beta}, \boldsymbol{X}))$
    - Mean square error, Absolute error, Prediction error, ...

Cross-Validation: A How To Guide

Step	Training	Validation ("Test")
1	Group2, Group3, Group 4, . . ., Group K	Group 1
2	Group 1, Group3, Group 4, . . ., Group K	Group 2
3	Group 1, Group 2, Group 4, . . ., Group K	Group 3
⋮	⋮	⋮
K	Group 1, Group 2, Group 3, . . ., Group K - 1	Group K

Strategy:

- Divide data into K groups
- Train data on $K - 1$ groups. Estimate $\hat{f}^{-K}(\boldsymbol{\beta}, \boldsymbol{X})$
- Predict values for K^{th}
- Summarize performance with loss function: $L(\boldsymbol{Y}_i, \hat{f}^{-k}(\boldsymbol{\beta}, \boldsymbol{X}))$
  - Mean square error, Absolute error, Prediction error, ...

CV(ind. classification) $= \frac{1}{N}\sum_{i=1}^{N} L(\boldsymbol{Y}_i, f^{-k}(\boldsymbol{\beta}, \boldsymbol{X}_i))$

Cross-Validation: A How To Guide

Step	Training	Validation ("Test")
1	Group2, Group3, Group 4, . . ., Group K	Group 1
2	Group 1, Group3, Group 4, . . ., Group K	Group 2
3	Group 1, Group 2, Group 4, . . ., Group K	Group 3
⋮	⋮	⋮
K	Group 1, Group 2, Group 3, . . ., Group K - 1	Group K

Strategy:

- Divide data into K groups
- Train data on $K-1$ groups. Estimate $\hat{f}^{-K}(\boldsymbol{\beta}, \boldsymbol{X})$
- Predict values for K^{th}
- Summarize performance with loss function: $L(\boldsymbol{Y}_i, \hat{f}^{-k}(\boldsymbol{\beta}, \boldsymbol{X}))$
    - Mean square error, Absolute error, Prediction error, ...

CV(ind. classification) $= \frac{1}{N}\sum_{i=1}^{N} L(\boldsymbol{Y}_i, f^{-k}(\boldsymbol{\beta}, \boldsymbol{X}_i))$

CV(proportions) $=$
$\frac{1}{K}\sum_{j=1}^{K}$ Mean Square Error Proportions from Group j

Cross-Validation: A How To Guide

Step	Training	Validation ("Test")
1	Group2, Group3, Group 4, ..., Group K	Group 1
2	Group 1, Group3, Group 4, ..., Group K	Group 2
3	Group 1, Group 2, Group 4, ..., Group K	Group 3
⋮	⋮	⋮
K	Group 1, Group 2, Group 3, ..., Group K - 1	Group K

Strategy:

- Divide data into K groups
- Train data on $K-1$ groups. Estimate $\hat{f}^{-K}(\boldsymbol{\beta}, \boldsymbol{X})$
- Predict values for K^{th}
- Summarize performance with loss function: $L(\boldsymbol{Y}_i, \hat{f}^{-k}(\boldsymbol{\beta}, \boldsymbol{X}))$
    - Mean square error, Absolute error, Prediction error, ...

  CV(ind. classification) = $\frac{1}{N}\sum_{i=1}^{N} L(\boldsymbol{Y}_i, f^{-k}(\boldsymbol{\beta}, \boldsymbol{X}_i))$

  CV(proportions) =
  $\frac{1}{K}\sum_{j=1}^{K}$ Mean Square Error Proportions from Group j
- Final choice: model with highest CV score

How Do We Select K? (HTF, Section 7.10)

Common values of K

- $K = 5$: Five fold cross validation
- $K = 10$: Ten fold cross validation
- $K = N$: Leave one out cross validation

Considerations:

- How sensitive are inferences to number of coded documents? (HTF, pg 243-244)
- 200 labeled documents
  - $K = N \rightarrow 199$ documents to train,
  - $K = 10 \rightarrow 180$ documents to train
  - $K = 5 \rightarrow 160$ documents to train
- 50 labeled documents
  - $K = N \rightarrow 49$ documents to train,
  - $K = 10 \rightarrow 45$ documents to train
  - $K = 5 \rightarrow 40$ documents to train
- How long will it take to run models?
  - $K-$fold cross validation requires $K \times$ One model run
- What is the correct loss function?

If you cross validate, you really need to cross validate (Section 7.10.2, ESL)

- Use CV to estimate prediction error
- All supervised steps performed in cross-validation
- Underestimate prediction error
- Could lead to selecting lower performing model

Example from Facebook Data

What do people say to legislators? (Franco, Grimmer, and Lee 2017)

1) Example: estimating classification error
    a) Accuracy in legislator posts: 75%
    b) Accuracy in public posts: 66.25%

Credit Claiming (Back to Ridge/Lasso, Grimmer, Westwood, and Messing 2014)

```
library(glmnet)
set.seed(8675309) ##setting seed
folds<- sample(1:10, nrow(dtm), replace=T) ##assigning to fold
out_of_samp<- c() ##collecting the predictions
```

Credit Claiming (Back to Ridge/Lasso, Grimmer, Westwood, and Messing 2014)

```
for(z in 1:10){
train<- which(folds!=z) ##the observations we will use to train the model

test<- which(folds==z) ##the observations we will use to test the model
part1<- cv.glmnet(x = dtm[train,], y = credit[train], alpha = 1, family =
b́inomial)́ ##fitting the LASSO model on the data.
## alpha = 1 -> LASSO
## alpha = 0 -> RIDGE
## 0<alpha<1 -> Elastic-Net
out_of_samp[test]<- predict(part1, newx= dtm[test,], s = part1$lambda.min,
type =ćlass)́ ##predicting the labels
print(z) ##printing the labels
}
conf_table<- table(out_of_samp, credit) ##calculating the confusion table
> round(sum(diag(conf_table))/len(credit), 3)
[1] 0.844
```

Generalized Cross Validation and Ridge Regression

In some special cases there are analytic solutions:

Generalized Cross Validation and Ridge Regression

In some special cases there are analytic solutions:

$$\beta^{\text{Ridge}} = \left(\boldsymbol{X}^{'}\boldsymbol{X} + \lambda \boldsymbol{I}_J\right)^{-1}\boldsymbol{X}^{'}\boldsymbol{Y}$$

Generalized Cross Validation and Ridge Regression

In some special cases there are analytic solutions:

$$\begin{aligned} \boldsymbol{\beta}^{\text{Ridge}} &= \left(\boldsymbol{X}^{'}\boldsymbol{X} + \lambda \boldsymbol{I}_J\right)^{-1} \boldsymbol{X}^{'}\boldsymbol{Y} \\ \widehat{\boldsymbol{Y}} &= \boldsymbol{X}(\beta)^{\text{Ridge}} \end{aligned}$$

Generalized Cross Validation and Ridge Regression

In some special cases there are analytic solutions:

$$\begin{aligned}
\boldsymbol{\beta}^{\text{Ridge}} &= \left(\boldsymbol{X}^{'}\boldsymbol{X} + \lambda \boldsymbol{I}_J\right)^{-1}\boldsymbol{X}^{'}\boldsymbol{Y} \\
\widehat{\boldsymbol{Y}} &= \boldsymbol{X}(\boldsymbol{\beta})^{\text{Ridge}} \\
&= \underbrace{\boldsymbol{X}\left(\boldsymbol{X}^{'}\boldsymbol{X} + \lambda \boldsymbol{I}_J\right)^{-1}\boldsymbol{X}^{'}}_{\text{Hat Matrix}}\boldsymbol{Y}
\end{aligned}$$

Generalized Cross Validation and Ridge Regression

In some special cases there are analytic solutions:

$$
\begin{aligned}
\boldsymbol{\beta}^{\text{Ridge}} &= \left(\boldsymbol{X}^{'}\boldsymbol{X} + \lambda \boldsymbol{I}_J\right)^{-1} \boldsymbol{X}^{'}\boldsymbol{Y} \\
\widehat{\boldsymbol{Y}} &= \boldsymbol{X}(\boldsymbol{\beta})^{\text{Ridge}} \\
&= \underbrace{\boldsymbol{X}\left(\boldsymbol{X}^{'}\boldsymbol{X} + \lambda \boldsymbol{I}_J\right)^{-1} \boldsymbol{X}^{'}}_{\text{Hat Matrix}}\boldsymbol{Y} \\
\widehat{\boldsymbol{Y}} &= \underbrace{\boldsymbol{H}}_{\text{Smoother Matrix}} \quad \boldsymbol{Y}
\end{aligned}
$$

Generalized Cross Validation and Ridge Regression

In some special cases there are analytic solutions:

$$\begin{aligned}
\boldsymbol{\beta}^{\text{Ridge}} &= \left(\boldsymbol{X}^{'}\boldsymbol{X} + \lambda \boldsymbol{I}_J\right)^{-1}\boldsymbol{X}^{'}\boldsymbol{Y} \\
\widehat{\boldsymbol{Y}} &= \boldsymbol{X}(\boldsymbol{\beta})^{\text{Ridge}} \\
&= \underbrace{\boldsymbol{X}\left(\boldsymbol{X}^{'}\boldsymbol{X} + \lambda \boldsymbol{I}_J\right)^{-1}\boldsymbol{X}^{'}}_{\text{Hat Matrix}}\boldsymbol{Y} \\
\widehat{\boldsymbol{Y}} &= \underbrace{\boldsymbol{H}}_{\text{Smoother Matrix}} \quad \boldsymbol{Y}
\end{aligned}$$

Generalized Cross Validation and Ridge Regression

Why do we care?

Generalized Cross Validation and Ridge Regression

Why do we care?
Leave one out cross validation

Generalized Cross Validation and Ridge Regression

Why do we care?
Leave one out cross validation

$$\text{Cross Validation}(1) \quad = \quad \frac{1}{N}\sum_{i=1}^{N}(Y_i - f(\boldsymbol{X}_{-i}, \boldsymbol{Y}_{-i}, \lambda, \hat{\beta}))^2$$

Generalized Cross Validation and Ridge Regression

Why do we care?
Leave one out cross validation

$$\begin{aligned}
\text{Cross Validation}(1) &= \frac{1}{N}\sum_{i=1}^{N}(Y_i - f(\boldsymbol{X}_{-i}, \boldsymbol{Y}_{-i}, \lambda, \hat{\boldsymbol{\beta}}))^2 \\
&= \frac{1}{N}\sum_{i=1}^{N}\left(\frac{Y_i - f(\boldsymbol{X}, \boldsymbol{Y}, \lambda, \hat{\boldsymbol{\beta}})}{1 - H_{ii}}\right)^2
\end{aligned}$$

Generalized Cross Validation and Ridge Regression

Calculating $\boldsymbol{H}$ can be computationally expensive

Generalized Cross Validation and Ridge Regression

Calculating $\boldsymbol{H}$ can be computationally expensive

- Trace($\boldsymbol{H}$) $\equiv$ Tr($\boldsymbol{H}$) $= \sum_{i=1}^{N} H_{ii}$

Generalized Cross Validation and Ridge Regression

Calculating $\boldsymbol{H}$ can be computationally expensive

- Trace($\boldsymbol{H}$) $\equiv$ Tr($\boldsymbol{H}$) $= \sum_{i=1}^{N} H_{ii}$
- Tr($\boldsymbol{H}$) $=$ Effective number of parameters (class regression $=$ number of independent variables $+$ 1)

Generalized Cross Validation and Ridge Regression

Calculating $\boldsymbol{H}$ can be computationally expensive

- Trace($\boldsymbol{H}$) $\equiv$ Tr($\boldsymbol{H}$) $= \sum_{i=1}^{N} H_{ii}$
- Tr($\boldsymbol{H}$) = Effective number of parameters (class regression = number of independent variables + 1)
- For Ridge regression:

Generalized Cross Validation and Ridge Regression

Calculating $\boldsymbol{H}$ can be computationally expensive

- Trace($\boldsymbol{H}$) $\equiv$ Tr($\boldsymbol{H}$) $= \sum_{i=1}^{N} H_{ii}$
- Tr($\boldsymbol{H}$) = Effective number of parameters (class regression = number of independent variables + 1)
- For Ridge regression:

$$\text{Tr}(\boldsymbol{H}) \quad = \quad \sum_{j=1}^{J} \frac{\lambda_j}{\lambda_j + \underbrace{\lambda}_{\text{Penalty}}}$$

Generalized Cross Validation and Ridge Regression

Calculating $\boldsymbol{H}$ can be computationally expensive

- Trace($\boldsymbol{H}$) $\equiv$ Tr($\boldsymbol{H}$) $= \sum_{i=1}^{N} H_{ii}$
- Tr($\boldsymbol{H}$) = Effective number of parameters (class regression = number of independent variables + 1)
- For Ridge regression:

$$\text{Tr}(\boldsymbol{H}) \;=\; \sum_{j=1}^{J} \frac{\lambda_j}{\lambda_j + \underbrace{\lambda}_{\text{Penalty}}}$$

where λ_j is the j^{th} Eigenvalue from $\boldsymbol{\Sigma} = \boldsymbol{X}^{'}\boldsymbol{X}$

Generalized Cross Validation and Ridge Regression

Calculating $\boldsymbol{H}$ can be computationally expensive

- $\text{Trace}(\boldsymbol{H}) \equiv \text{Tr}(\boldsymbol{H}) = \sum_{i=1}^{N} H_{ii}$
- $\text{Tr}(\boldsymbol{H})$ = Effective number of parameters (class regression = number of independent variables + 1)
- For Ridge regression:

$$\text{Tr}(\boldsymbol{H}) \; = \; \sum_{j=1}^{J} \frac{\lambda_j}{\lambda_j + \underbrace{\lambda}_{\text{Penalty}}}$$

where λ_j is the j^{th} Eigenvalue from $\boldsymbol{\Sigma} = \boldsymbol{X}^{'}\boldsymbol{X}$ (!!!!!)

Generalized Cross Validation and Ridge Regression

Calculating $\boldsymbol{H}$ can be computationally expensive

- Trace($\boldsymbol{H}$) $\equiv$ Tr($\boldsymbol{H}$) $= \sum_{i=1}^{N} H_{ii}$
- Tr($\boldsymbol{H}$) = Effective number of parameters (class regression = number of independent variables + 1)
- For Ridge regression:

$$\text{Tr}(\boldsymbol{H}) \;=\; \sum_{j=1}^{J} \frac{\lambda_j}{\lambda_j + \underbrace{\lambda}_{\text{Penalty}}}$$

where λ_j is the j^{th} Eigenvalue from $\boldsymbol{\Sigma} = \boldsymbol{X}^{'}\boldsymbol{X}$ (!!!!!)

Define generalized cross validation:

Generalized Cross Validation and Ridge Regression

Calculating $\boldsymbol{H}$ can be computationally expensive

- Trace($\boldsymbol{H}$) $\equiv$ Tr($\boldsymbol{H}$) $= \sum_{i=1}^{N} H_{ii}$
- Tr($\boldsymbol{H}$) = Effective number of parameters (class regression = number of independent variables + 1)
- For Ridge regression:

$$\text{Tr}(\boldsymbol{H}) \quad = \quad \sum_{j=1}^{J} \frac{\lambda_j}{\lambda_j + \underbrace{\lambda}_{\text{Penalty}}}$$

where λ_j is the j^{th} Eigenvalue from $\boldsymbol{\Sigma} = \boldsymbol{X}^{'}\boldsymbol{X}$ (!!!!!)

Define generalized cross validation:

$$\text{GCV} \quad = \quad \frac{1}{N}\sum_{i=1}^{N}\left(\frac{Y_i - \hat{Y}_i}{1 - \frac{\text{Tr}(\boldsymbol{H})}{N}}\right)^2$$

Generalized Cross Validation and Ridge Regression

Calculating $\boldsymbol{H}$ can be computationally expensive

- Trace($\boldsymbol{H}$) $\equiv$ Tr($\boldsymbol{H}$) $= \sum_{i=1}^{N} H_{ii}$
- Tr($\boldsymbol{H}$) = Effective number of parameters (class regression = number of independent variables + 1)
- For Ridge regression:

$$\text{Tr}(\boldsymbol{H}) = \sum_{j=1}^{J} \frac{\lambda_j}{\lambda_j + \underbrace{\lambda}_{\text{Penalty}}}$$

  where λ_j is the j^{th} Eigenvalue from $\boldsymbol{\Sigma} = \boldsymbol{X}^{'}\boldsymbol{X}$ (!!!!!)

Define generalized cross validation:

$$\text{GCV} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{Y_i - \hat{Y}_i}{1 - \frac{\text{Tr}(\boldsymbol{H})}{N}}\right)^2$$

Applicable in any setting where we can write Smoother matrix

Generalized Cross Validation and Ridge Regression

Calculating $\boldsymbol{H}$ can be computationally expensive

- Trace($\boldsymbol{H}$) $\equiv$ Tr($\boldsymbol{H}$) $= \sum_{i=1}^{N} H_{ii}$
- Tr($\boldsymbol{H}$) = Effective number of parameters (class regression = number of independent variables + 1)
- For Ridge regression:

$$\text{Tr}(\boldsymbol{H}) \quad = \quad \sum_{j=1}^{J} \frac{\lambda_j}{\lambda_j + \underbrace{\lambda}_{\text{Penalty}}}$$

where λ_j is the j^{th} Eigenvalue from $\boldsymbol{\Sigma} = \boldsymbol{X}^{'}\boldsymbol{X}$ (!!!!!)

Define generalized cross validation:

$$\text{GCV} \quad = \quad \frac{1}{N}\sum_{i=1}^{N}\left(\frac{Y_i - \hat{Y}_i}{1 - \frac{\text{Tr}(\boldsymbol{H})}{N}}\right)^2$$

Applicable in any setting where we can write Smoother matrix