Text as Data

Justin Grimmer

Associate Professor
Department of Political Science
University of Chicago

January 10th, 2018

Regular Expressions (from Jurafsky Slides)

Systematic Searches

A language for searching texts:

- Count mentions of a person
- Calculate amount of money discussed
- Prepare texts for analysis: Identify where to "split" a document
- ...

Provide a quick introduction here, with some examples

Regular Expressions, Some Basics (from Jurafsky Slides)

- Disjunctions

RE	**Match**	**Example Patterns Matched**
`[mM]oney`	Money or money	"Money"
`[abc]`	'a', 'b', *or* 'c'	"Investing in Iran"
		"is dangerous business"
`[1234567890]`	any digit	"sitting on $7.5 billion dollars"
		"2005 and 2006, more than "
		"$150 million dollars"
`[\.]`	A period	" 'Run!', he screamed."

Regular Expressions, Some Basics (from Jurafsky Slides)

- Ranges

RE	Match	Example Patterns Matched
`[A-Z]`	an upper case letter	"Rep. Anthony Weiner (D-Brooklyn & Queens)"
`[a-z]`	a lower case letter	"ACORN's"
`[0-9]`	a single digit	"(9th CD) "

Regular Expressions, Some Basics (from Jurafsky Slides)

- Negations

RE	Match	Example Patterns Matched
[^A-Z]	not an upper case letter	"ACORN's"
[^Ss]	neither 'S' nor 's'	"ACORN's"
[^\.]	not a period	" 'Run!', he screamed."

Regular Expressions, Some Basics (from Jurafsky Slides)

- Optional Characters: `?`, `*`, `+`

RE	Match	Example Patterns Matched
`colou?r`	Words with `u` 0 or 1 times	"color" or "colour "
`oo*h!`	Words with `o` 0 or more times	"oh!" or "ooh!" or "oooh!"
`o+h!`	Words with `o` 1 or more times	"oh!" or "ooh!" or "oooooh!" or

Regular Expressions, Some Basics (from Jurafsky Slides)

- Wild Cards .

RE	**Match**	**Example Patterns Matched**
`beg.n`	Any word with "beg" then "n"	"begin" or
		"began" or
		"begun" or
		"beggn" (Poor grammar!)

Regular Expressions, Some Basics (from Jurafsky Slides)

- Start of the line anchor ^, end of the line anchor $

RE	Match	Example Patterns Match
`^[A-Z]`	Upper case start of line	"Palo Alto"
		"the town of Palo Alto"
`^[^A-Z]`	Not upper case start of line	"the town of Palo Alto"
		"Palo Alto"
`^.`	Start of line	"Palo Alto"
		"the town of Palo Alto"
`.$`	Identify character that ends a line	"Wait!"
		"This is the end."

Regular Expressions, Some Basics (from Jurafsky Slides)

- "Or" | statements, Useful short hand

RE	**Match**	**Example Patterns Matched**
yours|mine	Matches "yours" or "mine"	"it's either <u>yours</u> or <u>mine</u>"
\ d	Any digit	"<u>1</u>-Mississippi"
\ D	Any non-digit	"1<u>-Mississippi</u>"
\ s	Any whitespace character	"1,<u> </u>2"
\ S	Any non-whitespace character	"<u>1,</u> <u>2</u>"
\ w	Any alpha-numeric	"<u>1</u>-<u>Mississippi</u> "
\ W	Any non-alpha numeric	"1<u>-</u>Mississippi"

Regular Expressions, Some Basics (from Jurafsky Slides)

Quick Example to Illuminate Differences:
A “simple” example: identify all instances of `the`.

Regular Expressions, Some Basics (from Jurafsky Slides)

Quick Example to Illuminate Differences:
A "simple" example: identify all instances of `the`.

- `the`

Regular Expressions, Some Basics (from Jurafsky Slides)

Quick Example to Illuminate Differences:
A “simple” example: identify all instances of `the`.

- `the`

  Misses capitalized examples

Regular Expressions, Some Basics (from Jurafsky Slides)

Quick Example to Illuminate Differences:
A “simple” example: identify all instances of `the`.

- `the`

  Misses capitalized examples
- `[tT]he`

Regular Expressions, Some Basics (from Jurafsky Slides)

Quick Example to Illuminate Differences:
A “simple” example: identify all instances of `the`.

- `the`

  Misses capitalized examples

- `[tT]he`

  Returns words that are too long (`theocrat`, `theme`)

Regular Expressions, Some Basics (from Jurafsky Slides)

Quick Example to Illuminate Differences:
A "simple" example: identify all instances of `the`.

- `the`

  Misses capitalized examples
- `[tT]he`

  Returns words that are too long (`theocrat`, `theme`)
- [^a-zA-Z][tT]he[^a-zA-Z]

Regular Expressions, Some Basics (from Jurafsky Slides)

Quick Example to Illuminate Differences:
A "simple" example: identify all instances of `the`.

- `the`

  Misses capitalized examples

- `[tT]he`

  Returns words that are too long (`theocrat`, `theme`)

- [^a-zA-Z][tT]he[^a-zA-Z]

  Misses the first "the" in a sentence

Regular Expressions, Some Basics (from Jurafsky Slides)

Quick Example to Illuminate Differences:
A "simple" example: identify all instances of `the`.

- `the`

  Misses capitalized examples

- `[tT]he`

  Returns words that are too long (`theocrat`, `theme`)

- [^a-zA-Z][tT]he[^a-zA-Z]

  Misses the first "the" in a sentence

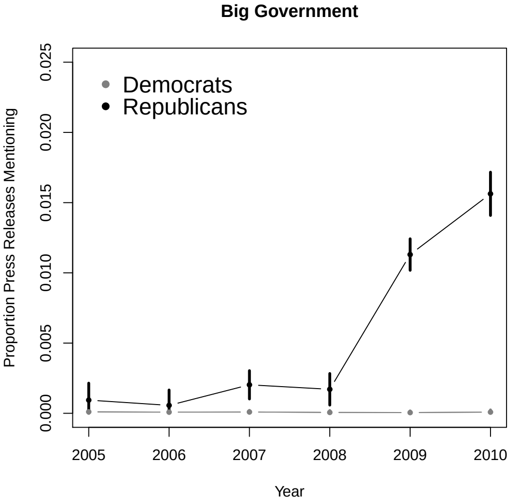
- (^ | [^ a-zA-Z])[tT]he[^ a-zA-Z]

An Example: Searching for Tea Party Language

Grimmer, Westwood, and Messing (2014): Criticism and credit

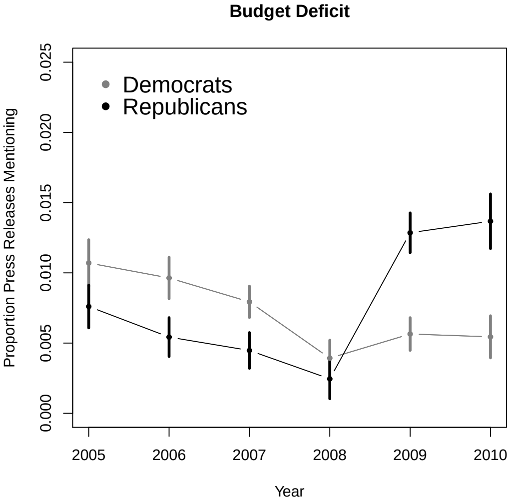
An Example: Searching for Tea Party Language

Grimmer, Westwood, and Messing (2014): Criticism and credit



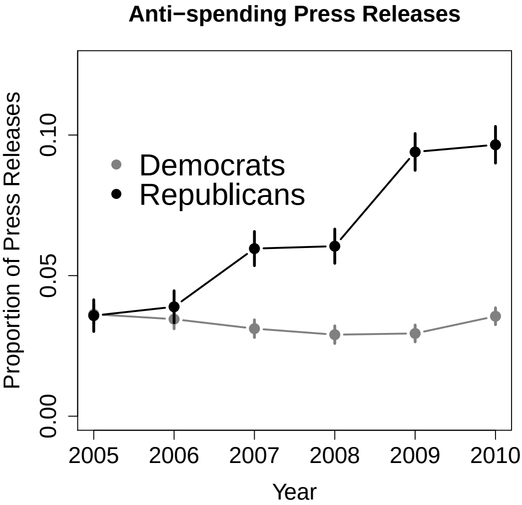
An Example: Searching for Tea Party Language

Grimmer, Westwood, and Messing (2014): Criticism and credit



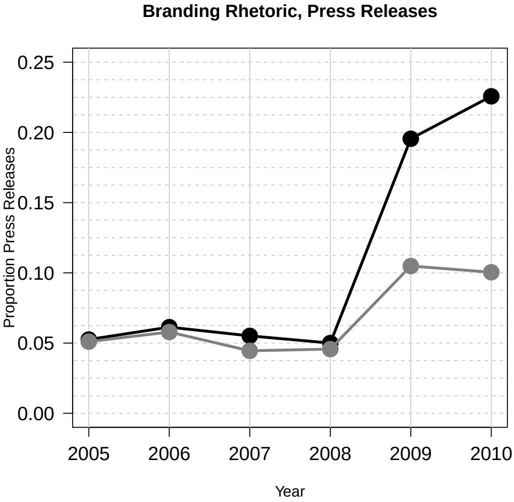
An Example: Searching for Tea Party Language

Grimmer, Westwood, and Messing (2014): Criticism and credit



An Example: Searching for Tea Party Language

Goodman, Grimmer, Parker, Zlotnik (2015): Criticism



Regular Expressions on Steroids: Cheating Detection Software

- WCopyFind:
  `http://plagiarism.bloomfieldmedia.com/z-wordpress/software/wcopyfind/`

Regular Expressions on Steroids: Cheating Detection Software

- WCopyFind:
  http://plagiarism.bloomfieldmedia.com/z-wordpress/software/wcopyfind/

- What constitutes plagiarism?

Regular Expressions on Steroids: Cheating Detection Software

- WCopyFind:
  `http://plagiarism.bloomfieldmedia.com/z-wordpress/software/wcopyfind/`

- What constitutes plagiarism?
- Edit distance:

Regular Expressions on Steroids: Cheating Detection Software

- WCopyFind:
  `http://plagiarism.bloomfieldmedia.com/z-wordpress/software/wcopyfind/`
- What constitutes plagiarism?
- Edit distance:
  - Heuristically: how many letters to change from *a* to *b*

Regular Expressions on Steroids: Cheating Detection Software

- WCopyFind:
  `http://plagiarism.bloomfieldmedia.com/z-wordpress/software/wcopyfind/`
- What constitutes plagiarism?
- Edit distance:
  - Heuristically: how many letters to change from *a* to *b*
- Sets many parameters:

Regular Expressions on Steroids: Cheating Detection Software

- WCopyFind:
  `http://plagiarism.bloomfieldmedia.com/z-wordpress/software/wcopyfind/`
- What constitutes plagiarism?
- Edit distance:
  - Heuristically: how many letters to change from *a* to *b*
- Sets many parameters:
  - Number of differences between pair of "strings"

Regular Expressions on Steroids: Cheating Detection Software

- WCopyFind:
  `http://plagiarism.bloomfieldmedia.com/z-wordpress/software/wcopyfind/`
- What constitutes plagiarism?
- Edit distance:
  - Heuristically: how many letters to change from *a* to *b*
- Sets many parameters:
  - Number of differences between pair of "strings"
  - Length of character strings to consider

Regular Expressions on Steroids: Cheating Detection Software

- WCopyFind:
  `http://plagiarism.bloomfieldmedia.com/z-wordpress/software/wcopyfind/`
- What constitutes plagiarism?
- Edit distance:
  - Heuristically: how many letters to change from *a* to *b*
- Sets many parameters:
  - Number of differences between pair of "strings"
  - Length of character strings to consider
  - Number of matching strings to constitute match

Regular Expressions on Steroids: Cheating Detection Software

- WCopyFind:
  `http://plagiarism.bloomfieldmedia.com/z-wordpress/software/wcopyfind/`
- What constitutes plagiarism?
- Edit distance:
  - Heuristically: how many letters to change from *a* to *b*
- Sets many parameters:
  - Number of differences between pair of "strings"
  - Length of character strings to consider
  - Number of matching strings to constitute match
- Useful:

Regular Expressions on Steroids: Cheating Detection Software

- WCopyFind:
  `http://plagiarism.bloomfieldmedia.com/z-wordpress/software/wcopyfind/`
- What constitutes plagiarism?
- Edit distance:
  - Heuristically: how many letters to change from *a* to *b*
- Sets many parameters:
  - Number of differences between pair of "strings"
  - Length of character strings to consider
  - Number of matching strings to constitute match
- Useful:
  - Media uptake

Regular Expressions on Steroids: Cheating Detection Software

- WCopyFind:
  `http://plagiarism.bloomfieldmedia.com/z-wordpress/software/wcopyfind/`
- What constitutes plagiarism?
- Edit distance:
  - Heuristically: how many letters to change from *a* to *b*
- Sets many parameters:
  - Number of differences between pair of "strings"
  - Length of character strings to consider
  - Number of matching strings to constitute match
- Useful:
  - Media uptake
  - Joint Press Releases

Research process:

- Discovery
- Measurement
- Causal Inference (Prediction)

Gary King, Jen Pan, and Molly Roberts (2015)$\rightsquigarrow$ discovery

Texts and Geometry

Consider a document-term matrix

$$\boldsymbol{X} = \begin{pmatrix} 1 & 2 & 0 & \ldots & 0 \\ 0 & 0 & 3 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & \ldots & 3 \end{pmatrix}$$

Texts and Geometry

Consider a document-term matrix

$$\boldsymbol{X} = \begin{pmatrix} 1 & 2 & 0 & \ldots & 0 \\ 0 & 0 & 3 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & \ldots & 3 \end{pmatrix}$$

Suppose documents live in a space

Texts and Geometry

Consider a document-term matrix

$$\boldsymbol{X} = \begin{pmatrix} 1 & 2 & 0 & \ldots & 0 \\ 0 & 0 & 3 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & \ldots & 3 \end{pmatrix}$$

Suppose documents live in a space $\rightsquigarrow$ rich set of results from linear algebra

Texts and Geometry

Consider a document-term matrix

$$\boldsymbol{X} = \begin{pmatrix} 1 & 2 & 0 & \ldots & 0 \\ 0 & 0 & 3 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & \ldots & 3 \end{pmatrix}$$

Suppose documents live in a space $\leadsto$ rich set of results from linear algebra

- Provides a geometry

Texts and Geometry

Consider a document-term matrix

$$\boldsymbol{X} = \begin{pmatrix} 1 & 2 & 0 & \ldots & 0 \\ 0 & 0 & 3 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & \ldots & 3 \end{pmatrix}$$

Suppose documents live in a space $\rightsquigarrow$ rich set of results from linear algebra

- Provides a geometry $\rightsquigarrow$ modify with word weighting

Texts and Geometry

Consider a document-term matrix

$$\boldsymbol{X} = \begin{pmatrix} 1 & 2 & 0 & \ldots & 0 \\ 0 & 0 & 3 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & \ldots & 3 \end{pmatrix}$$

Suppose documents live in a space $\rightsquigarrow$ rich set of results from linear algebra

- Provides a geometry $\rightsquigarrow$ modify with word weighting
- Natural notions of distance

Texts and Geometry

Consider a document-term matrix

$$\boldsymbol{X} = \begin{pmatrix} 1 & 2 & 0 & \ldots & 0 \\ 0 & 0 & 3 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & \ldots & 3 \end{pmatrix}$$

Suppose documents live in a space $\rightsquigarrow$ rich set of results from linear algebra

- Provides a geometry $\rightsquigarrow$ modify with word weighting
- Natural notions of distance
- Kernel Trick: richer comparisons of large feature spaces

Texts and Geometry

Consider a document-term matrix

$$\boldsymbol{X} = \begin{pmatrix} 1 & 2 & 0 & \ldots & 0 \\ 0 & 0 & 3 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & \ldots & 3 \end{pmatrix}$$

Suppose documents live in a space $\rightsquigarrow$ rich set of results from linear algebra

- Provides a geometry $\rightsquigarrow$ modify with word weighting
- Natural notions of distance
- Kernel Trick: richer comparisons of large feature spaces
- Building block for clustering, supervised learning, and scaling

Texts in Space

Texts in Space

$$\text{Doc1} \quad = \quad (1, 1, 3, \ldots, 5)$$

Texts in Space

$$\begin{aligned} \text{Doc1} &= (1, 1, 3, \ldots, 5) \\ \text{Doc2} &= (2, 0, 0, \ldots, 1) \end{aligned}$$

Texts in Space

$$
\begin{aligned}
\text{Doc1} &= (1, 1, 3, \ldots, 5) \\
\text{Doc2} &= (2, 0, 0, \ldots, 1) \\
\textbf{Doc1}, \textbf{Doc2} &\in \Re^{J}
\end{aligned}
$$

Texts in Space

$$\begin{aligned} \text{Doc1} &= (1, 1, 3, \ldots, 5) \\ \text{Doc2} &= (2, 0, 0, \ldots, 1) \\ \textbf{Doc1}, \textbf{Doc2} &\in \Re^J \end{aligned}$$

Inner Product between documents:

Texts in Space

$$\begin{aligned}
\text{Doc1} &= (1, 1, 3, \ldots, 5) \\
\text{Doc2} &= (2, 0, 0, \ldots, 1) \\
\textbf{Doc1}, \textbf{Doc2} &\in \Re^J
\end{aligned}$$

Inner Product between documents:

$$\textbf{Doc1} \cdot \textbf{Doc2} = (1, 1, 3, \ldots, 5)^{'}(2, 0, 0, \ldots, 1)$$

Texts in Space

$$\begin{aligned}
\text{Doc1} &= (1, 1, 3, \ldots, 5) \\
\text{Doc2} &= (2, 0, 0, \ldots, 1) \\
\textbf{Doc1}, \textbf{Doc2} &\in \Re^{J}
\end{aligned}$$

Inner Product between documents:

$$\begin{aligned}
\textbf{Doc1} \cdot \textbf{Doc2} &= (1, 1, 3, \ldots, 5)^{'}(2, 0, 0, \ldots, 1) \\
&= 1 \times 2 + 1 \times 0 + 3 \times 0 + \ldots + 5 \times 1
\end{aligned}$$

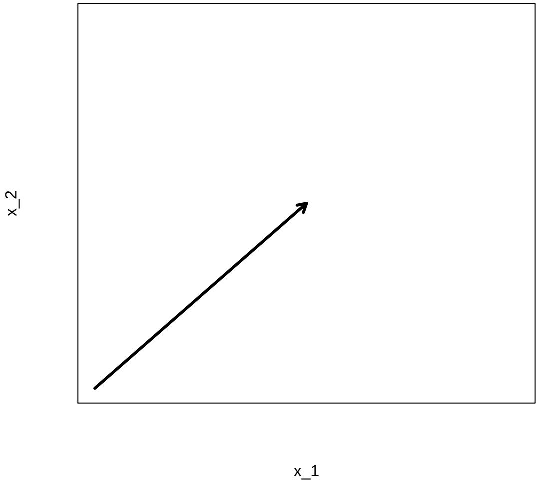
Texts in Space

$$\begin{aligned}
\text{Doc1} &= (1, 1, 3, \ldots, 5) \\
\text{Doc2} &= (2, 0, 0, \ldots, 1) \\
\textbf{Doc1}, \textbf{Doc2} &\in \Re^J
\end{aligned}$$

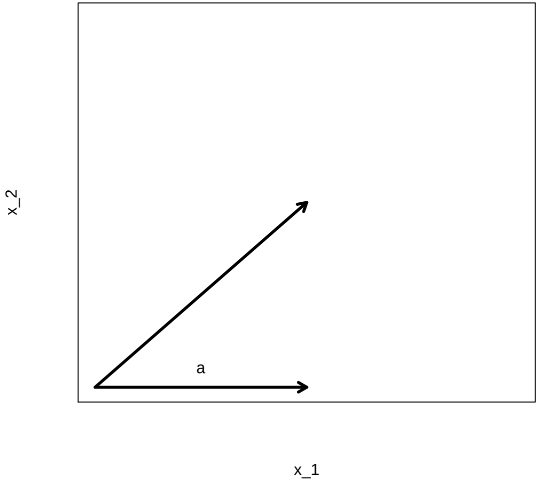
Inner Product between documents:

$$\begin{aligned}
\textbf{Doc1} \cdot \textbf{Doc2} &= (1, 1, 3, \ldots, 5)^{'}(2, 0, 0, \ldots, 1) \\
&= 1 \times 2 + 1 \times 0 + 3 \times 0 + \ldots + 5 \times 1 \\
&= 7
\end{aligned}$$

Vector Length

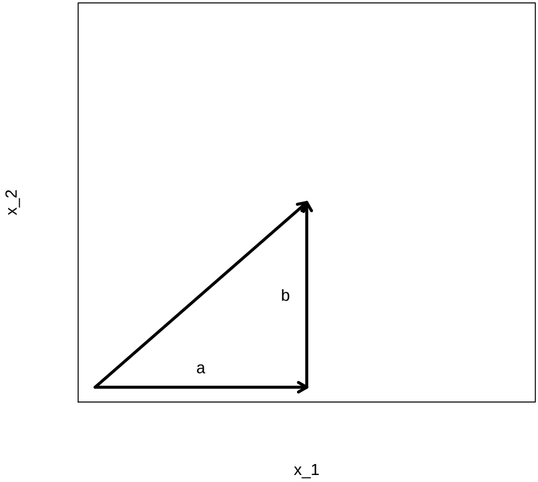


Vector Length



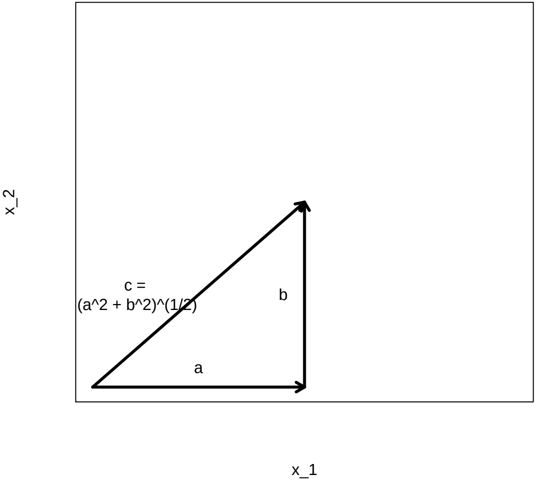
- Pythogorean Theorem: Side with length *a*

Vector Length



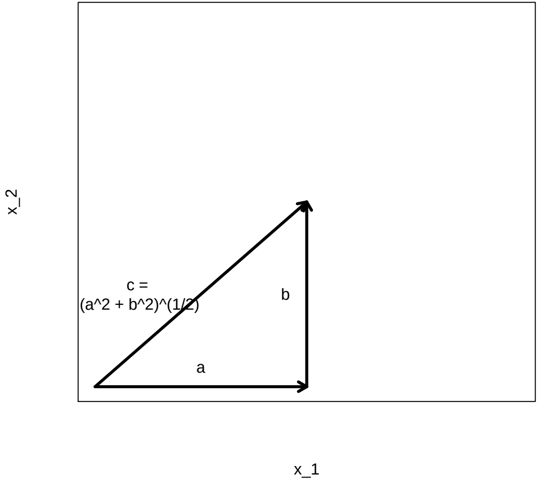
- Pythogorean Theorem: Side with length a
- Side with length b and right triangle

Vector Length



- Pythogorean Theorem: Side with length a
- Side with length b and right triangle
- $c = \sqrt{a^2 + b^2}$

Vector Length



- Pythogorean Theorem: Side with length a
- Side with length b and right triangle
- $c = \sqrt{a^2 + b^2}$
- This is generally true

Vector (Euclidean) Length

Definition

Suppose $\boldsymbol{v} \in \Re^J$. *Then, we will define its length as*

$$\begin{aligned} ||\boldsymbol{v}|| &= (\boldsymbol{v} \cdot \boldsymbol{v})^{1/2} \\ &= (v_1^2 + v_2^2 + v_3^2 + \ldots + v_J^2)^{1/2} \end{aligned}$$

Measures of Dissimilarity

Initial guess$\rightsquigarrow$ Distance metrics
Properties of a metric: (distance function) $d(\cdot, \cdot)$. Consider arbitrary documents $\boldsymbol{X}_i$, $\boldsymbol{X}_j$, $\boldsymbol{X}_k$

$\rightsquigarrow$

Measures of Dissimilarity

Initial guess$\rightsquigarrow$ Distance metrics

Properties of a metric: (distance function) $d(\cdot, \cdot)$. Consider arbitrary documents $\boldsymbol{X}_i$, $\boldsymbol{X}_j$, $\boldsymbol{X}_k$

1) $d(\boldsymbol{X}_i, \boldsymbol{X}_j) \geq 0$

$\rightsquigarrow$

Measures of Dissimilarity

Initial guess$\leadsto$ Distance metrics

Properties of a metric: (distance function) $d(\cdot, \cdot)$. Consider arbitrary documents $\boldsymbol{X}_i$, $\boldsymbol{X}_j$, $\boldsymbol{X}_k$

1) $d(\boldsymbol{X}_i, \boldsymbol{X}_j) \geq 0$
2) $d(\boldsymbol{X}_i, \boldsymbol{X}_j) = 0$ if and only if $\boldsymbol{X}_i = \boldsymbol{X}_j$

$\leadsto$

Measures of Dissimilarity

Initial guess$\leadsto$ Distance metrics

Properties of a metric: (distance function) $d(\cdot, \cdot)$. Consider arbitrary documents $\boldsymbol{X}_i$, $\boldsymbol{X}_j$, $\boldsymbol{X}_k$

1) $d(\boldsymbol{X}_i, \boldsymbol{X}_j) \geq 0$
2) $d(\boldsymbol{X}_i, \boldsymbol{X}_j) = 0$ if and only if $\boldsymbol{X}_i = \boldsymbol{X}_j$
3) $d(\boldsymbol{X}_i, \boldsymbol{X}_j) = d(\boldsymbol{X}_j, \boldsymbol{X}_i)$

$\leadsto$

Measures of Dissimilarity

Initial guess $\rightsquigarrow$ Distance metrics

Properties of a metric: (distance function) $d(\cdot, \cdot)$. Consider arbitrary documents $\boldsymbol{X}_i$, $\boldsymbol{X}_j$, $\boldsymbol{X}_k$

1) $d(\boldsymbol{X}_i, \boldsymbol{X}_j) \geq 0$
2) $d(\boldsymbol{X}_i, \boldsymbol{X}_j) = 0$ if and only if $\boldsymbol{X}_i = \boldsymbol{X}_j$
3) $d(\boldsymbol{X}_i, \boldsymbol{X}_j) = d(\boldsymbol{X}_j, \boldsymbol{X}_i)$
4) $d(\boldsymbol{X}_i, \boldsymbol{X}_k) \leq d(\boldsymbol{X}_i, \boldsymbol{X}_j) + d(\boldsymbol{X}_j, \boldsymbol{X}_k)$

$\rightsquigarrow$

Measures of Dissimilarity

Initial guess$\rightsquigarrow$ Distance metrics

Properties of a metric: (distance function) $d(\cdot, \cdot)$. Consider arbitrary documents $\boldsymbol{X}_i$, $\boldsymbol{X}_j$, $\boldsymbol{X}_k$

1) $d(\boldsymbol{X}_i, \boldsymbol{X}_j) \geq 0$
2) $d(\boldsymbol{X}_i, \boldsymbol{X}_j) = 0$ if and only if $\boldsymbol{X}_i = \boldsymbol{X}_j$
3) $d(\boldsymbol{X}_i, \boldsymbol{X}_j) = d(\boldsymbol{X}_j, \boldsymbol{X}_i)$
4) $d(\boldsymbol{X}_i, \boldsymbol{X}_k) \leq d(\boldsymbol{X}_i, \boldsymbol{X}_j) + d(\boldsymbol{X}_j, \boldsymbol{X}_k)$

Explore distance functions to compare documents$\rightsquigarrow$

Measures of Dissimilarity

Initial guess$\rightsquigarrow$ Distance metrics

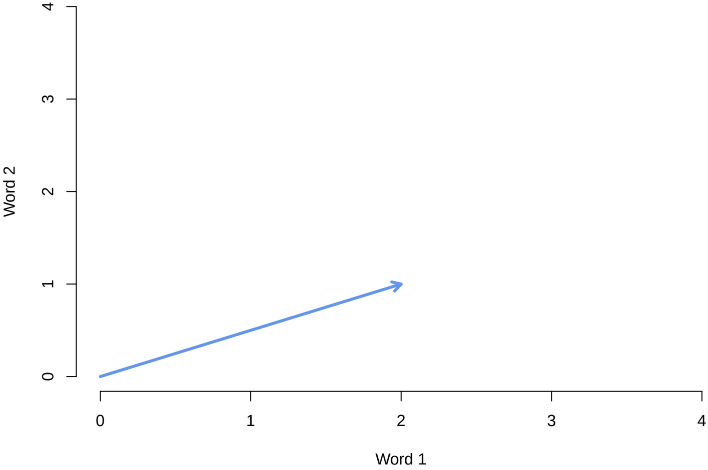
Properties of a metric: (distance function) $d(\cdot, \cdot)$. Consider arbitrary documents $\boldsymbol{X}_i$, $\boldsymbol{X}_j$, $\boldsymbol{X}_k$

1) $d(\boldsymbol{X}_i, \boldsymbol{X}_j) \geq 0$
2) $d(\boldsymbol{X}_i, \boldsymbol{X}_j) = 0$ if and only if $\boldsymbol{X}_i = \boldsymbol{X}_j$
3) $d(\boldsymbol{X}_i, \boldsymbol{X}_j) = d(\boldsymbol{X}_j, \boldsymbol{X}_i)$
4) $d(\boldsymbol{X}_i, \boldsymbol{X}_k) \leq d(\boldsymbol{X}_i, \boldsymbol{X}_j) + d(\boldsymbol{X}_j, \boldsymbol{X}_k)$

Explore distance functions to compare documents$\rightsquigarrow$Do we want additional assumptions/properties?

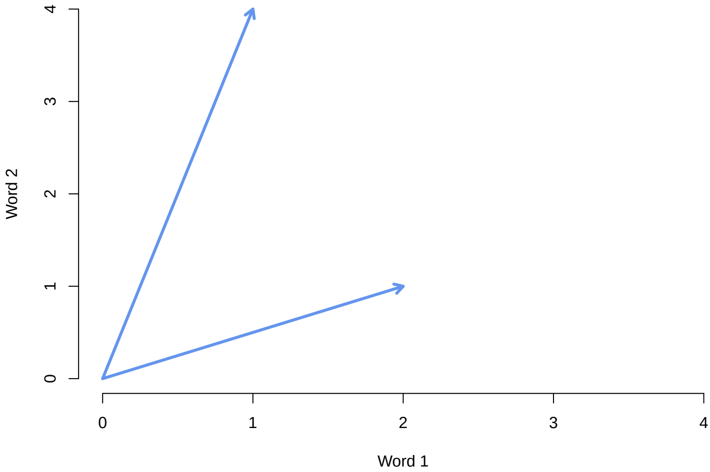
Measuring the Distance Between Documents

Euclidean Distance



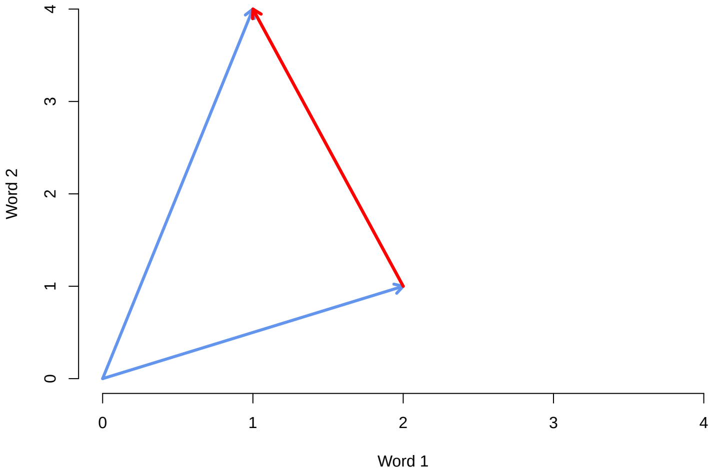
Measuring the Distance Between Documents

Euclidean Distance



Measuring the Distance Between Documents

Euclidean Distance



Measuring the Distance Between Documents

Definition

The Euclidean distance between documents $\boldsymbol{X}_i$ *and* $\boldsymbol{X}_j$ *as*

$$||\boldsymbol{X}_i - \boldsymbol{X}_j|| \quad = \quad \sqrt{\sum_{m=1}^{J} (x_{im} - x_{jm})^2}$$

Measuring the Distance Between Documents

Definition

The Euclidean distance between documents $\boldsymbol{X}_i$ *and* $\boldsymbol{X}_j$ *as*

$$||\boldsymbol{X}_i - \boldsymbol{X}_j|| = \sqrt{\sum_{m=1}^{J} (x_{im} - x_{jm})^2}$$

Suppose $\boldsymbol{X}_i = (1, 4)$ and $\boldsymbol{X}_j = (2, 1)$. The distance between the documents is:

$$\begin{aligned} ||(1,4) - (2,1)|| &= \sqrt{(1-2)^2 + (4-1)^2} \\ &= \sqrt{10} \end{aligned}$$

Measuring the Distance Between Documents

Many distance metrics

Measuring the Distance Between Documents

Many distance metrics Consider the Minkowski family

Measuring the Distance Between Documents

Many distance metrics Consider the Minkowski family

Definition

The Minkowski Distance between documents $\boldsymbol{X}_i$ *and* $\boldsymbol{X}_j$ *for value p is*

$$d_p(\mathbf{X}_i, \mathbf{X}_j) \quad = \quad \left(\sum_{m=1}^{J} |x_{im} - x_{jm}|^p \right)^{1/p}$$

Members of the Minkowski Family

Members of the Minkowski Family

Manhattan metric

Members of the Minkowski Family

Manhattan metric

$$d_1(\mathbf{X}_i, \mathbf{X}_j) = \sum_{m=1}^{J} |x_{im} - x_{jm}|$$

Members of the Minkowski Family

Manhattan metric

$$
\begin{aligned}
d_1(\mathbf{X}_i, \mathbf{X}_j) &= \sum_{m=1}^{J} |x_{im} - x_{jm}| \\
d_1((1,4),(2,1)) &= |1| + |3| = 4
\end{aligned}
$$

Members of the Minkowski Family

Manhattan metric

$$
\begin{aligned}
d_1(\mathbf{X}_i, \mathbf{X}_j) &= \sum_{m=1}^{J} |x_{im} - x_{jm}| \\
d_1((1,4),(2,1)) &= |1| + |3| = 4
\end{aligned}
$$

Minkowski (p) metric

Members of the Minkowski Family

Manhattan metric

$$
\begin{aligned}
d_1(\mathbf{X}_i, \mathbf{X}_j) &= \sum_{m=1}^{J} |x_{im} - x_{jm}| \\
d_1((1,4),(2,1)) &= |1| + |3| = 4
\end{aligned}
$$

Minkowski (p) metric

$$
\begin{aligned}
d_p(\mathbf{X}_i, \mathbf{X}_j) &= \left(\sum_{m=1}^{J} |x_{im} - x_{jm}|^p \right)^{1/p} \\
d_p((1,4),(2,1)) &= (|1-2|^p + |4-1|^p)^{1/p}
\end{aligned}
$$

What Does p Do?

What Does p Do?

Increasing $p \rightsquigarrow$ greater importance of coordinates with largest differences

What Does p Do?

Increasing $p \rightsquigarrow$ greater importance of coordinates with largest differences
If we let $p \to \infty$ Obtain maximum-metric (Chebyshev's Metric)

What Does p Do?

Increasing $p \rightsquigarrow$ greater importance of coordinates with largest differences
If we let $p \rightarrow \infty$ Obtain maximum-metric (Chebyshev's Metric)

$$\lim_{p \rightarrow \infty} d_p(\mathbf{X}_i, \mathbf{X}_j) \quad = \quad \max_{m=1}^{J} |x_{im} - x_{jm}|$$

What Does p Do?

Increasing $p \rightsquigarrow$ greater importance of coordinates with largest differences
If we let $p \rightarrow \infty$ Obtain maximum-metric (Chebyshev's Metric)

$$\lim_{p \rightarrow \infty} d_p(\mathbf{X}_i, \mathbf{X}_j) \quad = \quad \max_{m=1}^{J} |x_{im} - x_{jm}|$$

In words: distance between documents only the biggest difference

What Does p Do?

Increasing $p \rightsquigarrow$ greater importance of coordinates with largest differences
If we let $p \rightarrow \infty$ Obtain maximum-metric (Chebyshev's Metric)

$$\lim_{p \rightarrow \infty} d_p(\mathbf{X}_i, \mathbf{X}_j) = \max_{m=1}^{J} |x_{im} - x_{jm}|$$

In words: distance between documents only the biggest difference
All other differences do not contribute to distance measure

What Does p Do?

Increasing $p \rightsquigarrow$ greater importance of coordinates with largest differences
If we let $p \rightarrow \infty$ Obtain maximum-metric (Chebyshev's Metric)

$$\lim_{p \rightarrow \infty} d_p(\mathbf{X}_i, \mathbf{X}_j) \quad = \quad \max_{m=1}^{J} |x_{im} - x_{jm}|$$

In words: distance between documents only the biggest difference
All other differences do not contribute to distance measure
Decreasing $p \rightsquigarrow$ greater importance of coordinates with smallest differences

What Does p Do?

Increasing $p \rightsquigarrow$ greater importance of coordinates with largest differences
If we let $p \rightarrow \infty$ Obtain maximum-metric (Chebyshev's Metric)

$$\lim_{p \rightarrow \infty} d_p(\mathbf{X}_i, \mathbf{X}_j) = \max_{m=1}^{J} |x_{im} - x_{jm}|$$

In words: distance between documents only the biggest difference
All other differences do not contribute to distance measure
Decreasing $p \rightsquigarrow$ greater importance of coordinates with smallest differences

$$\lim_{p \rightarrow -\infty} d_p(\mathbf{X}_i, \mathbf{X}_j) = \min_{m=1}^{J} |x_{im} - x_{jm}|$$

Comparing the Metrics

Suppose $\boldsymbol{X}_i = (10, 4, 3)$, $\boldsymbol{X}_j = (0, 4, 3)$, and $\boldsymbol{X}_k = (0, 0, 0)$

Comparing the Metrics

Suppose $\boldsymbol{X}_i = (10, 4, 3)$, $\boldsymbol{X}_j = (0, 4, 3)$, and $\boldsymbol{X}_k = (0, 0, 0)$
Then:

Comparing the Metrics

Suppose $\boldsymbol{X}_i = (10, 4, 3)$, $\boldsymbol{X}_j = (0, 4, 3)$, and $\boldsymbol{X}_k = (0, 0, 0)$
Then:

$$d_1(\boldsymbol{X}_i, \boldsymbol{X}_j) = 10$$

Comparing the Metrics

Suppose $\boldsymbol{X}_i = (10, 4, 3)$, $\boldsymbol{X}_j = (0, 4, 3)$, and $\boldsymbol{X}_k = (0, 0, 0)$
Then:

$$
\begin{aligned}
d_1(\boldsymbol{X}_i, \boldsymbol{X}_j) &= 10 \\
d_1(\boldsymbol{X}_i, \boldsymbol{X}_k) &= 10 + 4 + 3 = 17
\end{aligned}
$$

Comparing the Metrics

Suppose $\boldsymbol{X}_i = (10, 4, 3)$, $\boldsymbol{X}_j = (0, 4, 3)$, and $\boldsymbol{X}_k = (0, 0, 0)$
Then:

$$\begin{aligned} d_1(\boldsymbol{X}_i, \boldsymbol{X}_j) &= 10 \\ d_1(\boldsymbol{X}_i, \boldsymbol{X}_k) &= 10 + 4 + 3 = 17 \\ d_2(\boldsymbol{X}_i, \boldsymbol{X}_j) &= 10 \end{aligned}$$

Comparing the Metrics

Suppose $\boldsymbol{X}_i = (10, 4, 3)$, $\boldsymbol{X}_j = (0, 4, 3)$, and $\boldsymbol{X}_k = (0, 0, 0)$
Then:

$$
\begin{aligned}
d_1(\boldsymbol{X}_i, \boldsymbol{X}_j) &= 10 \\
d_1(\boldsymbol{X}_i, \boldsymbol{X}_k) &= 10 + 4 + 3 = 17 \\
d_2(\boldsymbol{X}_i, \boldsymbol{X}_j) &= 10 \\
d_2(\boldsymbol{X}_i, \boldsymbol{X}_k) &= \sqrt{10^2 + 4^2 + 3^2} = \sqrt{125} = 11.18
\end{aligned}
$$

Comparing the Metrics

Suppose $\boldsymbol{X}_i = (10, 4, 3)$, $\boldsymbol{X}_j = (0, 4, 3)$, and $\boldsymbol{X}_k = (0, 0, 0)$
Then:

$$
\begin{aligned}
d_1(\boldsymbol{X}_i, \boldsymbol{X}_j) &= 10 \\
d_1(\boldsymbol{X}_i, \boldsymbol{X}_k) &= 10 + 4 + 3 = 17 \\
d_2(\boldsymbol{X}_i, \boldsymbol{X}_j) &= 10 \\
d_2(\boldsymbol{X}_i, \boldsymbol{X}_k) &= \sqrt{10^2 + 4^2 + 3^2} = \sqrt{125} = 11.18 \\
d_4(\boldsymbol{X}_i, \boldsymbol{X}_j) &= 10
\end{aligned}
$$

Comparing the Metrics

Suppose $\boldsymbol{X}_i = (10, 4, 3)$, $\boldsymbol{X}_j = (0, 4, 3)$, and $\boldsymbol{X}_k = (0, 0, 0)$
Then:

$$
\begin{aligned}
d_1(\boldsymbol{X}_i, \boldsymbol{X}_j) &= 10 \\
d_1(\boldsymbol{X}_i, \boldsymbol{X}_k) &= 10 + 4 + 3 = 17 \\
d_2(\boldsymbol{X}_i, \boldsymbol{X}_j) &= 10 \\
d_2(\boldsymbol{X}_i, \boldsymbol{X}_k) &= \sqrt{10^2 + 4^2 + 3^2} = \sqrt{125} = 11.18 \\
d_4(\boldsymbol{X}_i, \boldsymbol{X}_j) &= 10 \\
d_4(\boldsymbol{X}_i, \boldsymbol{X}_k) &= \sqrt{10^4 + 4^4 + 3^4} = (10337)^{1/4} = 10.08
\end{aligned}
$$

Comparing the Metrics

Suppose $\boldsymbol{X}_i = (10, 4, 3)$, $\boldsymbol{X}_j = (0, 4, 3)$, and $\boldsymbol{X}_k = (0, 0, 0)$
Then:

$$\begin{aligned}
d_1(\boldsymbol{X}_i, \boldsymbol{X}_j) &= 10 \\
d_1(\boldsymbol{X}_i, \boldsymbol{X}_k) &= 10 + 4 + 3 = 17 \\
d_2(\boldsymbol{X}_i, \boldsymbol{X}_j) &= 10 \\
d_2(\boldsymbol{X}_i, \boldsymbol{X}_k) &= \sqrt{10^2 + 4^2 + 3^2} = \sqrt{125} = 11.18 \\
d_4(\boldsymbol{X}_i, \boldsymbol{X}_j) &= 10 \\
d_4(\boldsymbol{X}_i, \boldsymbol{X}_k) &= \sqrt{10^4 + 4^4 + 3^4} = (10337)^{1/4} = 10.08 \\
d_\infty(\boldsymbol{X}_i, \boldsymbol{X}_j) &= 10
\end{aligned}$$

Comparing the Metrics

Suppose $\boldsymbol{X}_i = (10, 4, 3)$, $\boldsymbol{X}_j = (0, 4, 3)$, and $\boldsymbol{X}_k = (0, 0, 0)$
Then:

$$\begin{aligned}
d_1(\boldsymbol{X}_i, \boldsymbol{X}_j) &= 10 \\
d_1(\boldsymbol{X}_i, \boldsymbol{X}_k) &= 10 + 4 + 3 = 17 \\
d_2(\boldsymbol{X}_i, \boldsymbol{X}_j) &= 10 \\
d_2(\boldsymbol{X}_i, \boldsymbol{X}_k) &= \sqrt{10^2 + 4^2 + 3^2} = \sqrt{125} = 11.18 \\
d_4(\boldsymbol{X}_i, \boldsymbol{X}_j) &= 10 \\
d_4(\boldsymbol{X}_i, \boldsymbol{X}_k) &= \sqrt{10^4 + 4^4 + 3^4} = (10337)^{1/4} = 10.08 \\
d_\infty(\boldsymbol{X}_i, \boldsymbol{X}_j) &= 10 \\
d_\infty(\boldsymbol{X}_i, \boldsymbol{X}_k) &= 10
\end{aligned}$$

Are all differences equal?

Previous metrics treat all dimensions as equal

Are all differences equal?

Previous metrics treat all dimensions as equal
We may want to engage in some scaling/reweighting

Are all differences equal?

Previous metrics treat all dimensions as equal
We may want to engage in some scaling/reweighting
Mahalanobis Distance

Are all differences equal?

Previous metrics treat all dimensions as equal
We may want to engage in some scaling/reweighting
Mahalanobis Distance

Definition

Suppose that we have a covariance matrix $\boldsymbol{\Sigma}$

Are all differences equal?

Previous metrics treat all dimensions as equal
We may want to engage in some scaling/reweighting
Mahalanobis Distance

Definition

Suppose that we have a covariance matrix $\boldsymbol{\Sigma}$. Then we can define the Mahalanobis Distance between documents $\boldsymbol{X}_i$ and $\boldsymbol{X}_j$ as ,

Are all differences equal?

Previous metrics treat all dimensions as equal
We may want to engage in some scaling/reweighting
Mahalanobis Distance

Definition

Suppose that we have a covariance matrix $\boldsymbol{\Sigma}$. Then we can define the Mahalanobis Distance between documents $\boldsymbol{X}_i$ and $\boldsymbol{X}_j$ as ,

$$d_{Mah}(\boldsymbol{X}_i, \boldsymbol{X}_j) = \sqrt{(\boldsymbol{X}_i - \boldsymbol{X}_j)'\boldsymbol{\Sigma}^{-1}(\boldsymbol{X}_i - \boldsymbol{X}_j)}$$

Are all differences equal?

Previous metrics treat all dimensions as equal
We may want to engage in some scaling/reweighting
Mahalanobis Distance

Definition

Suppose that we have a covariance matrix $\boldsymbol{\Sigma}$*. Then we can define the Mahalanobis Distance between documents* $\boldsymbol{X}_i$ *and* $\boldsymbol{X}_j$ *as ,*

$$d_{Mah}(\boldsymbol{X}_i, \boldsymbol{X}_j) = \sqrt{(\boldsymbol{X}_i - \boldsymbol{X}_j)' \boldsymbol{\Sigma}^{-1} (\boldsymbol{X}_i - \boldsymbol{X}_j)}$$

More generally: $\boldsymbol{\Sigma}$ could be symmetric and positive-definite

Are all differences equal?

Previous metrics treat all dimensions as equal
We may want to engage in some scaling/reweighting
Mahalanobis Distance

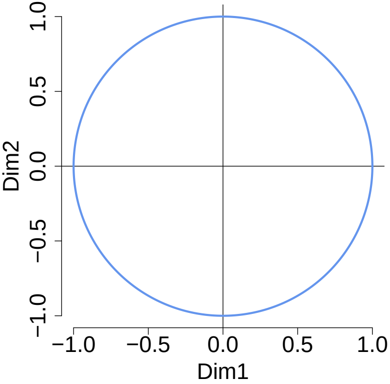
Definition

Suppose that we have a covariance matrix $\boldsymbol{\Sigma}$*. Then we can define the Mahalanobis Distance between documents* $\boldsymbol{X}_i$ *and* $\boldsymbol{X}_j$ *as ,*

$$d_{Mah}(\boldsymbol{X}_i, \boldsymbol{X}_j) = \sqrt{(\boldsymbol{X}_i - \boldsymbol{X}_j)' \boldsymbol{\Sigma}^{-1} (\boldsymbol{X}_i - \boldsymbol{X}_j)}$$

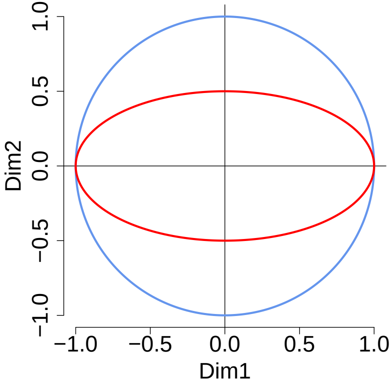
More generally: $\boldsymbol{\Sigma}$ could be symmetric and positive-definite
What does $\boldsymbol{\Sigma}$ do?

Some Intuition: The Unit Circle



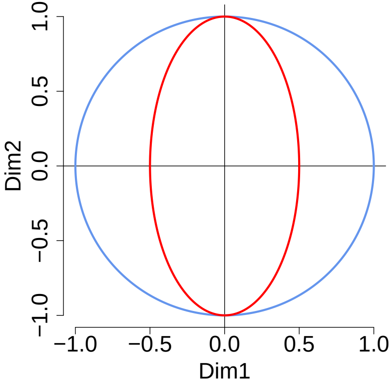
$$\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Some Intuition: The Unit Circle



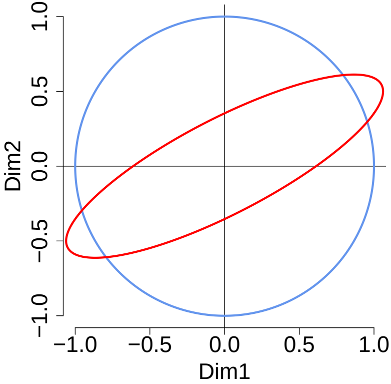
$$\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0 \\ 0 & 0.5 \end{pmatrix}$$

Some Intuition: The Unit Circle



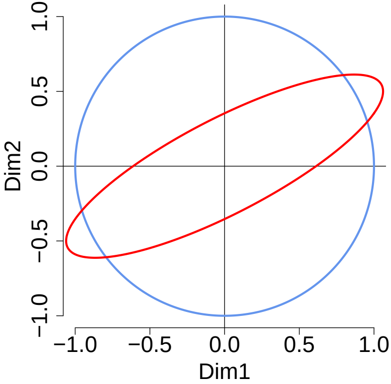
$$\boldsymbol{\Sigma} = \begin{pmatrix} 0.5 & 0 \\ 0 & 1 \end{pmatrix}$$

Some Intuition: The Unit Circle



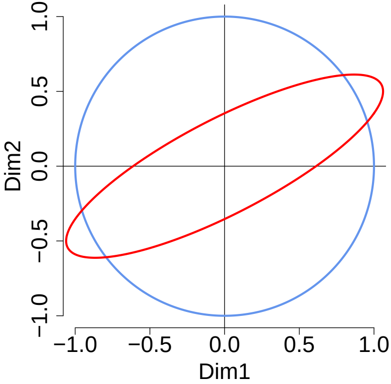
$$\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0.3 \\ 0.3 & 0.5 \end{pmatrix}$$

Some Intuition: The Unit Circle



$$\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0.3 \\ 0.3 & 0.5 \end{pmatrix}$$

Some Intuition: The Unit Circle



$$\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0.3 \\ 0.3 & 0.5 \end{pmatrix}$$

Measuring Distance with Mahalanobis

Special Case 1: Identity Matrix

Measuring Distance with Mahalanobis

Special Case 1: Identity Matrix

$$\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Measuring Distance with Mahalanobis

Special Case 1: Identity Matrix

$$\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Then distance is Euclidean

Measuring Distance with Mahalanobis

Special Case 1: Identity Matrix

$$\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Then distance is Euclidean

Special Case 2: Diagonal Matrix

$$\boldsymbol{\Sigma} = \begin{pmatrix} \sigma_1^2 & 0 & \ldots & 0 \\ 0 & \sigma_2^2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \sigma_J^2 \end{pmatrix}$$

Measuring Distance with Mahalanobis

Special Case 1: Identity Matrix

$$\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Then distance is Euclidean

Special Case 2: Diagonal Matrix

$$\boldsymbol{\Sigma} = \begin{pmatrix} \sigma_1^2 & 0 & \ldots & 0 \\ 0 & \sigma_2^2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \sigma_J^2 \end{pmatrix}$$

Then

$$d_{\text{Mah}}(\boldsymbol{X}_i, \boldsymbol{X}_j) = \sqrt{\sum_{m=1}^{J} \frac{(x_{im} - x_{jm})^2}{\sigma_m^2}}$$

Measuring Similarity

Measuring Similarity

What properties should similarity measure have?

Measuring Similarity

What properties should similarity measure have?

- Maximum: document with itself

Measuring Similarity

What properties should similarity measure have?

- Maximum: document with itself
- Minimum: documents have no words in common (orthogonal)

Measuring Similarity

What properties should similarity measure have?

- Maximum: document with itself
- Minimum: documents have no words in common (orthogonal)
- Increasing when more of same words used

Measuring Similarity

What properties should similarity measure have?

- Maximum: document with itself
- Minimum: documents have no words in common (orthogonal)
- Increasing when more of same words used
- ? $s(a, b) = s(b, a)$.

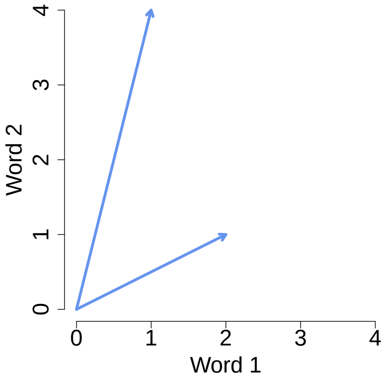
Measuring Similarity

What properties should similarity measure have?

- Maximum: document with itself
- Minimum: documents have no words in common (orthogonal)
- Increasing when more of same words used
- ? $s(a, b) = s(b, a)$.

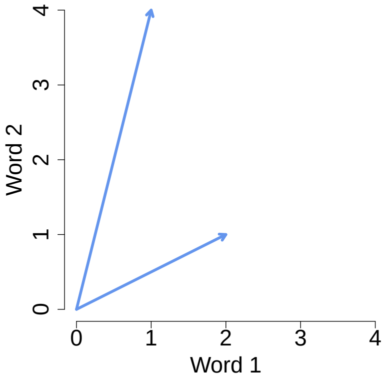
How should additional words be treated?

Measuring Similarity



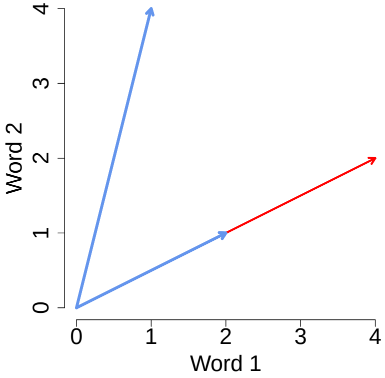
Measure 1: Inner product

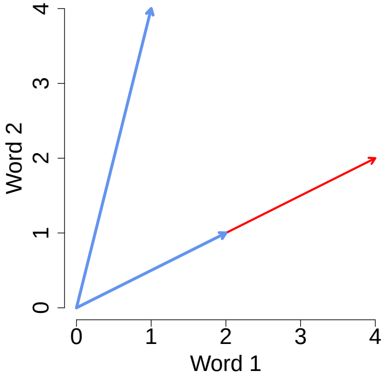
Measuring Similarity



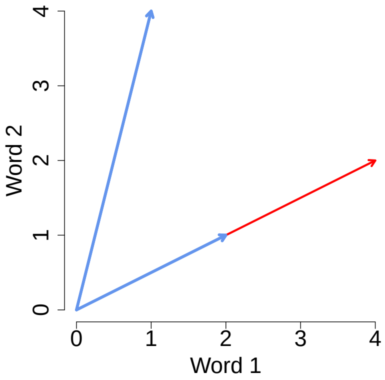
Measure 1: Inner product

$$(2,1)^{'} \cdot (1,4) \quad = \quad 6$$


Word 2
Word 1
0
1
2
3
4

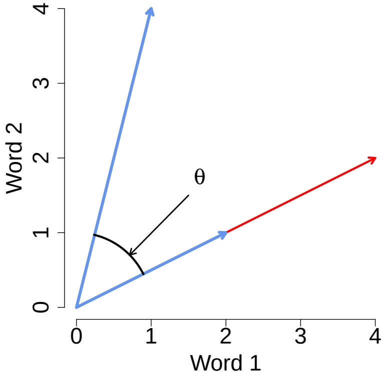


Problem(?): length dependent



Problem(?): length dependent

$$(4,2)^{'}(1,4) \quad = \quad 12$$



Problem(?): length dependent

$$\begin{aligned}(4,2)^{'}(1,4) &= 12 \\ a \cdot b &= ||a|| \times ||b|| \times \cos\theta\end{aligned}$$

Cosine Similarity

Cosine Similarity

$$\cos\theta \quad = \quad \left(\frac{a}{||a||}\right) \cdot \left(\frac{b}{||b||}\right)$$

Cosine Similarity

$$\cos\theta = \left(\frac{a}{||a||}\right)\cdot\left(\frac{b}{||b||}\right)$$

$$\frac{(4,2)}{||(4,2)||} = (0.89, 0.45)$$

Cosine Similarity

$$
\begin{aligned}
\cos\theta &= \left(\frac{a}{||a||}\right) \cdot \left(\frac{b}{||b||}\right) \\
\frac{(4,2)}{||(4,2)||} &= (0.89, 0.45) \\
\frac{(2,1)}{||(2,1)||} &= (0.89, 0.45)
\end{aligned}
$$

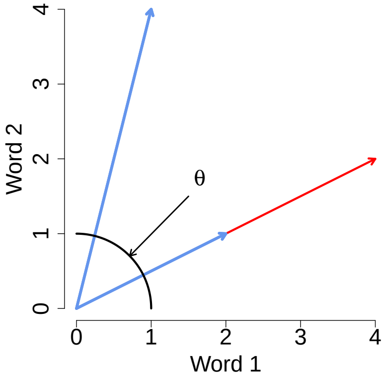
Cosine Similarity

$$
\begin{aligned}
\cos\theta &= \left(\frac{a}{||a||}\right) \cdot \left(\frac{b}{||b||}\right) \\
\frac{(4,2)}{||(4,2)||} &= (0.89, 0.45) \\
\frac{(2,1)}{||(2,1)||} &= (0.89, 0.45) \\
\frac{(1,4)}{||(1,4)||} &= (0.24, 0.97)
\end{aligned}
$$

Cosine Similarity

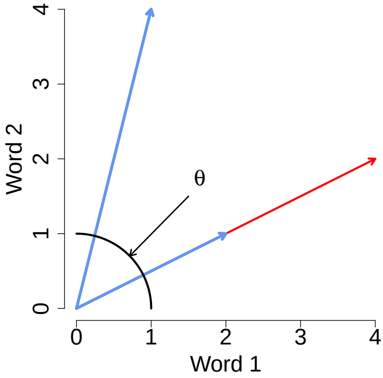
$$
\begin{aligned}
\cos\theta &= \left(\frac{a}{||a||}\right) \cdot \left(\frac{b}{||b||}\right) \\
\frac{(4,2)}{||(4,2)||} &= (0.89, 0.45) \\
\frac{(2,1)}{||(2,1)||} &= (0.89, 0.45) \\
\frac{(1,4)}{||(1,4)||} &= (0.24, 0.97) \\
(0.89, 0.45)^{'}(0.24, 0.97) &= 0.65
\end{aligned}
$$

Cosine Similarity



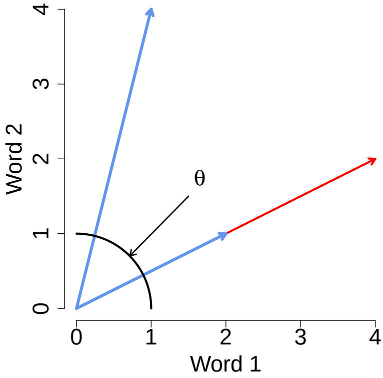
$\cos\theta$: removes document length from similarity measure

Cosine Similarity



$\cos\theta$: removes document length from similarity measure
Projects texts to unit length representation$\rightsquigarrow$ onto sphere

Cosine Similarity



$\cos\theta$: removes document length from similarity measure

Projects texts to unit length representation $\rightsquigarrow$ onto sphere

Von Mises-Fisher Distribution

Consider document $\boldsymbol{X}_i$.

Von Mises-Fisher Distribution

Consider document $\boldsymbol{X}_i$.

$$\boldsymbol{X}_i^{*} = \frac{\boldsymbol{X}_i}{||\boldsymbol{X}_i||}$$

Von Mises-Fisher Distribution

Consider document $\boldsymbol{X}_i$.

$$\boldsymbol{X}_i^* = \frac{\boldsymbol{X}_i}{||\boldsymbol{X}_i||}$$

Then we might suppose:

Von Mises-Fisher Distribution

Consider document $\boldsymbol{X}_i$.

$$\boldsymbol{X}_i^* = \frac{\boldsymbol{X}_i}{||\boldsymbol{X}_i||}$$

Then we might suppose:

$$\boldsymbol{X}_i^* \sim \text{von Mises-Fisher}(\kappa, \boldsymbol{\mu})$$

Von Mises-Fisher Distribution

Consider document $\boldsymbol{X}_i$.

$$\boldsymbol{X}_i^* = \frac{\boldsymbol{X}_i}{||\boldsymbol{X}_i||}$$

Then we might suppose:

$$\begin{aligned}\boldsymbol{X}_i^* &\sim \text{von Mises-Fisher}(\kappa, \boldsymbol{\mu}) \\ p(\boldsymbol{x}_i|\kappa, \boldsymbol{\mu}) &= c(\kappa)\exp\left(\kappa \boldsymbol{x}_i^* \boldsymbol{\mu}\right)\end{aligned}$$

Von Mises-Fisher Distribution

Consider document $\boldsymbol{X}_i$.

$$\boldsymbol{X}_i^* = \frac{\boldsymbol{X}_i}{||\boldsymbol{X}_i||}$$

Then we might suppose:

$$\begin{aligned} \boldsymbol{X}_i^* &\sim \text{von Mises-Fisher}(\kappa, \boldsymbol{\mu}) \\ p(\boldsymbol{x}_i|\kappa, \boldsymbol{\mu}) &= c(\kappa) \exp\left(\kappa \boldsymbol{x}_i^* \boldsymbol{\mu}\right) \end{aligned}$$

Normal distribution, on a sphere

Von Mises-Fisher Distribution

Consider document $\boldsymbol{X}_i$.

$$\boldsymbol{X}_i^* = \frac{\boldsymbol{X}_i}{||\boldsymbol{X}_i||}$$

Then we might suppose:

$$\begin{aligned} \boldsymbol{X}_i^* &\sim \text{von Mises-Fisher}(\kappa, \boldsymbol{\mu}) \\ p(\boldsymbol{x}_i|\kappa, \boldsymbol{\mu}) &= c(\kappa) \exp\left(\kappa \boldsymbol{x}_i^* \boldsymbol{\mu}\right) \end{aligned}$$

Normal distribution, on a sphere

- Straightforward to Maximize

Von Mises-Fisher Distribution

Consider document $\boldsymbol{X}_i$.

$$\boldsymbol{X}_i^* = \frac{\boldsymbol{X}_i}{||\boldsymbol{X}_i||}$$

Then we might suppose:

$$\begin{aligned}
\boldsymbol{X}_i^* &\sim \text{von Mises-Fisher}(\kappa, \boldsymbol{\mu}) \\
p(\boldsymbol{x}_i|\kappa, \boldsymbol{\mu}) &= c(\kappa)\exp\left(\kappa \boldsymbol{x}_i^* \boldsymbol{\mu}\right)
\end{aligned}$$

Normal distribution, on a sphere

- Straightforward to Maximize
- Conjugate to itself

Von Mises-Fisher Distribution

Consider document $\boldsymbol{X}_i$.

$$\boldsymbol{X}_i^* = \frac{\boldsymbol{X}_i}{||\boldsymbol{X}_i||}$$

Then we might suppose:

$$\begin{aligned}
\boldsymbol{X}_i^* &\sim \text{von Mises-Fisher}(\kappa, \boldsymbol{\mu}) \\
p(\boldsymbol{x}_i|\kappa, \boldsymbol{\mu}) &= c(\kappa) \exp\left(\kappa \boldsymbol{x}_i^* \boldsymbol{\mu}\right)
\end{aligned}$$

Normal distribution, on a sphere

- Straightforward to Maximize
- Conjugate to itself
- Useful for clustering, hierarchies of topics

Kernel Similarity

Definition

Suppose we have documents $\boldsymbol{X}_i$ and $\boldsymbol{X}_j$. Define the Gaussian kernel as

$$k(\boldsymbol{X}_i, \boldsymbol{X}_j) = \exp\left(-\frac{||\boldsymbol{X}_i - \boldsymbol{X}_j||^2}{\sigma^2}\right)$$

Kernel Similarity

Definition

Suppose we have documents $\boldsymbol{X}_i$ and $\boldsymbol{X}_j$. Define the Gaussian kernel as

$$k(\boldsymbol{X}_i, \boldsymbol{X}_j) = \exp\left(-\frac{||\boldsymbol{X}_i - \boldsymbol{X}_j||^2}{\sigma^2}\right)$$

Kernel of the Gaussian distribution

Kernel Similarity

Definition

Suppose we have documents $\boldsymbol{X}_i$ and $\boldsymbol{X}_j$. Define the Gaussian kernel as

$$k(\boldsymbol{X}_i, \boldsymbol{X}_j) = \exp\left(-\frac{||\boldsymbol{X}_i - \boldsymbol{X}_j||^2}{\sigma^2}\right)$$

Kernel of the Gaussian distribution

$\sigma^2 =$ determines sensitivity of the kernel

Kernel Similarity

Definition

Suppose we have documents $\boldsymbol{X}_i$ and $\boldsymbol{X}_j$. Define the Gaussian kernel as

$$k(\boldsymbol{X}_i, \boldsymbol{X}_j) \quad = \quad \exp\left(-\frac{||\boldsymbol{X}_i - \boldsymbol{X}_j||^2}{\sigma^2}\right)$$

Kernel of the Gaussian distribution

$\sigma^2 =$ determines sensitivity of the kernel

If $\boldsymbol{X}_i = \boldsymbol{X}_j$ then $k(\boldsymbol{X}_i, \boldsymbol{X}_j) = 1$

Kernel Similarity

Definition

Suppose we have documents $\boldsymbol{X}_i$ and $\boldsymbol{X}_j$. Define the Gaussian kernel as

$$k(\boldsymbol{X}_i, \boldsymbol{X}_j) \quad = \quad \exp\left(-\frac{||\boldsymbol{X}_i - \boldsymbol{X}_j||^2}{\sigma^2}\right)$$

Kernel of the Gaussian distribution

$\sigma^2 =$ determines sensitivity of the kernel

If $\boldsymbol{X}_i = \boldsymbol{X}_j$ then $k(\boldsymbol{X}_i, \boldsymbol{X}_j) = 1$

As $\boldsymbol{X}_i$ and $\boldsymbol{X}_j$ become more dissimilar, then $k(\boldsymbol{X}_i, \boldsymbol{X}_j) = 0$

Kernel Similarity

Definition

Suppose we have documents $\boldsymbol{X}_i$ *and* $\boldsymbol{X}_j$. *Define the Gaussian kernel as*

$$k(\boldsymbol{X}_i, \boldsymbol{X}_j) \quad = \quad \exp\left(-\frac{||\boldsymbol{X}_i - \boldsymbol{X}_j||^2}{\sigma^2}\right)$$

- Kernel of the Gaussian distribution
- $\sigma^2 =$ determines sensitivity of the kernel
- If $\boldsymbol{X}_i = \boldsymbol{X}_j$ then $k(\boldsymbol{X}_i, \boldsymbol{X}_j) = 1$
- As $\boldsymbol{X}_i$ and $\boldsymbol{X}_j$ become more dissimilar, then $k(\boldsymbol{X}_i, \boldsymbol{X}_j) = 0$
- Result$\leadsto$ often justify setting some kernel weights to zero

The Kernel Trick

Suppose all of our documents $\boldsymbol{X}_i \in \Re^J$

The Kernel Trick

Suppose all of our documents $\boldsymbol{X}_i \in \Re^J$

There may be some mapping $\phi : \Re^J \rightarrow \Re^M$ where $M > J$ that improves our performance "lift" to higher dimension

The Kernel Trick

Suppose all of our documents $\boldsymbol{X}_i \in \Re^J$
There may be some mapping $\phi : \Re^J \rightarrow \Re^M$ where $M > J$ that improves our performance "lift" to higher dimension
We might want, then,

The Kernel Trick

Suppose all of our documents $\boldsymbol{X}_i \in \Re^J$
There may be some mapping $\phi : \Re^J \rightarrow \Re^M$ where $M > J$ that improves our performance "lift" to higher dimension
We might want, then,

$$s(\phi(\boldsymbol{X}_i), \phi(\boldsymbol{X}_j)) \quad = \quad < \phi(\boldsymbol{X}_i), \phi(\boldsymbol{X}_j) >$$

The Kernel Trick

Suppose all of our documents $\boldsymbol{X}_i \in \Re^J$
There may be some mapping $\phi : \Re^J \rightarrow \Re^M$ where $M > J$ that improves our performance "lift" to higher dimension
We might want, then,

$$s(\phi(\boldsymbol{X}_i), \phi(\boldsymbol{X}_j)) \quad = \quad < \phi(\boldsymbol{X}_i), \phi(\boldsymbol{X}_j) >$$

- The only thing we care about, though is inner product of transformed variables

The Kernel Trick

Suppose all of our documents $\boldsymbol{X}_i \in \Re^J$
There may be some mapping $\phi : \Re^J \rightarrow \Re^M$ where $M > J$ that improves our performance "lift" to higher dimension
We might want, then,

$$s(\phi(\boldsymbol{X}_i), \phi(\boldsymbol{X}_j)) \quad = \quad < \phi(\boldsymbol{X}_i), \phi(\boldsymbol{X}_j) >$$

- The only thing we care about, though is inner product of transformed variables
- $\rightsquigarrow$ So long as we can calculate inner product, we need not make explicit transformation

The Kernel Trick

Suppose all of our documents $\boldsymbol{X}_i \in \Re^J$
There may be some mapping $\phi : \Re^J \rightarrow \Re^M$ where $M > J$ that improves our performance "lift" to higher dimension
We might want, then,

$$s(\phi(\boldsymbol{X}_i), \phi(\boldsymbol{X}_j)) \quad = \quad < \phi(\boldsymbol{X}_i), \phi(\boldsymbol{X}_j) >$$

- The only thing we care about, though is inner product of transformed variables
- $\leadsto$ So long as we can calculate inner product, we need not make explicit transformation
- $\leadsto$ Kernels provide methods for capture wide array of transformations.

The Kernel Trick

Suppose all of our documents $\boldsymbol{X}_i \in \Re^J$
There may be some mapping $\phi : \Re^J \rightarrow \Re^M$ where $M > J$ that improves our performance "lift" to higher dimension
We might want, then,

$$s(\phi(\boldsymbol{X}_i), \phi(\boldsymbol{X}_j)) \quad = \quad <\phi(\boldsymbol{X}_i), \phi(\boldsymbol{X}_j)>$$

- The only thing we care about, though is inner product of transformed variables
- $\rightsquigarrow$ So long as we can calculate inner product, we need not make explicit transformation
- $\rightsquigarrow$ Kernels provide methods for capture wide array of transformations.
- Kernel Trick$\rightsquigarrow$ calculate inner products on untransformed data (Gaussian Kernel), implicitly use wide array of ϕ's.

Weighting Words

Are all words created equal?

Weighting Words

Are all words created equal?

- Treat all words equally

Weighting Words

Are all words created equal?

- Treat all words equally
- Lots of noise

Weighting Words

Are all words created equal?

- Treat all words equally
- Lots of noise
- Reweight words

Weighting Words

Are all words created equal?

- Treat all words equally
- Lots of noise
- Reweight words
  - Accentuate words that are likely to be informative

Weighting Words

Are all words created equal?

- Treat all words equally
- Lots of noise
- Reweight words
  - Accentuate words that are likely to be informative
  - Make specific assumptions about characteristics of informative words

Weighting Words

Are all words created equal?

- Treat all words equally
- Lots of noise
- Reweight words
  - Accentuate words that are likely to be informative
  - Make specific assumptions about characteristics of informative words

How to generate weights?

Weighting Words

Are all words created equal?

- Treat all words equally
- Lots of noise
- Reweight words
  - Accentuate words that are likely to be informative
  - Make specific assumptions about characteristics of informative words

How to generate weights?

- Assumptions about separating words

Weighting Words

Are all words created equal?

- Treat all words equally
- Lots of noise
- Reweight words
  - Accentuate words that are likely to be informative
  - Make specific assumptions about characteristics of informative words

How to generate weights?

- Assumptions about separating words
- Use training set to identify separating words (Monroe, Ideology measurement)

Weighting Words: TF-IDF Weighting

What properties do words need to separate concepts?

Weighting Words: TF-IDF Weighting

What properties do words need to separate concepts?

- Used frequently

Weighting Words: TF-IDF Weighting

What properties do words need to separate concepts?

- Used frequently
- But not too frequently

Weighting Words: TF-IDF Weighting

What properties do words need to separate concepts?

- Used frequently
- But not too frequently

Ex. If all statements about OBL contain `Bin Laden` than this contributes nothing to similarity/dissimilarity measures

Weighting Words: TF-IDF Weighting

What properties do words need to separate concepts?

- Used frequently
- But not too frequently

Ex. If all statements about OBL contain `Bin Laden` than this contributes nothing to similarity/dissimilarity measures

Inverse document frequency:

Weighting Words: TF-IDF Weighting

What properties do words need to separate concepts?

- Used frequently
- But not too frequently

Ex. If all statements about OBL contain `Bin Laden` than this contributes nothing to similarity/dissimilarity measures

Inverse document frequency:

$$\text{n}_j = \text{No. documents in which word } j \text{ occurs}$$

Weighting Words: TF-IDF Weighting

What properties do words need to separate concepts?

- Used frequently
- But not too frequently

Ex. If all statements about OBL contain `Bin Laden` than this contributes nothing to similarity/dissimilarity measures

Inverse document frequency:

$$\begin{aligned} \mathrm{n}_j &= \text{No. documents in which word } j \text{ occurs} \\ \mathrm{idf}_j &= \log \frac{N}{n_j} \end{aligned}$$

Weighting Words: TF-IDF Weighting

What properties do words need to separate concepts?

- Used frequently
- But not too frequently

Ex. If all statements about OBL contain `Bin Laden` than this contributes nothing to similarity/dissimilarity measures

Inverse document frequency:

$$
\begin{aligned}
\mathrm{n}_j &= \text{No. documents in which word } j \text{ occurs} \\
\mathrm{idf}_j &= \log \frac{N}{n_j} \\
\mathbf{idf} &= (\mathrm{idf}_1, \mathrm{idf}_2, \ldots, \mathrm{idf}_J)
\end{aligned}
$$

Weighting Words: TF-IDF Weighting

Why log ?

Weighting Words: TF-IDF Weighting

Why log ?

- Maximum at $n_j = 1$

Weighting Words: TF-IDF Weighting

Why log ?

- Maximum at $n_j = 1$
- Decreases at rate $\frac{1}{n_j} \Rightarrow$ diminishing "penalty" for more common use

Weighting Words: TF-IDF Weighting

Why log ?

- Maximum at $n_j = 1$
- Decreases at rate $\frac{1}{n_j} \Rightarrow$ diminishing "penalty" for more common use
- Other functional forms are fine, embed assumptions about penalization of common use

Weighting Words: TF-IDF

Weighting Words: TF-IDF

$$\mathbf{X}_{i,\text{idf}} \equiv \underbrace{\mathbf{X}_i}_{\text{tf}} \times \mathbf{idf} \quad = \quad (X_{i1} \times \text{idf}_1, X_{i2} \times \text{idf}_2, \ldots, X_{iJ} \times \text{idf}_J)$$

Weighting Words: TF-IDF

$$\mathbf{X}_{i,\text{idf}} \equiv \underbrace{\mathbf{X}_i}_{\text{tf}} \times \textbf{idf} = (X_{i1} \times \text{idf}_1, X_{i2} \times \text{idf}_2, \ldots, X_{iJ} \times \text{idf}_J)$$

$$\mathbf{X}_{j,\text{idf}} \equiv \mathbf{X}_j \times \textbf{idf} = (X_{j1} \times \text{idf}_1, X_{j2} \times \text{idf}_2, \ldots, X_{jJ} \times \text{idf}_J)$$

Weighting Words: TF-IDF

$$\mathbf{X}_{i,\text{idf}} \equiv \underbrace{\mathbf{X}_i}_{\text{tf}} \times \mathbf{idf} = (X_{i1} \times \text{idf}_1, X_{i2} \times \text{idf}_2, \ldots, X_{iJ} \times \text{idf}_J)$$

$$\mathbf{X}_{j,\text{idf}} \equiv \mathbf{X}_j \times \mathbf{idf} = (X_{j1} \times \text{idf}_1, X_{j2} \times \text{idf}_2, \ldots, X_{jJ} \times \text{idf}_J)$$

How Does This Matter For Measuring Similarity/Dissimilarity?

Weighting Words: TF-IDF

$$\mathbf{X}_{i,\text{idf}} \equiv \underbrace{\mathbf{X}_i}_{\text{tf}} \times \textbf{idf} = (X_{i1} \times \text{idf}_1, X_{i2} \times \text{idf}_2, \ldots, X_{iJ} \times \text{idf}_J)$$

$$\mathbf{X}_{j,\text{idf}} \equiv \mathbf{X}_j \times \textbf{idf} = (X_{j1} \times \text{idf}_1, X_{j2} \times \text{idf}_2, \ldots, X_{jJ} \times \text{idf}_J)$$

How Does This Matter For Measuring Similarity/Dissimilarity?
Inner Product

Weighting Words: TF-IDF

$$\mathbf{X}_{i,\text{idf}} \equiv \underbrace{\mathbf{X}_i}_{\text{tf}} \times \textbf{idf} = (X_{i1} \times \text{idf}_1, X_{i2} \times \text{idf}_2, \ldots, X_{iJ} \times \text{idf}_J)$$

$$\mathbf{X}_{j,\text{idf}} \equiv \mathbf{X}_j \times \textbf{idf} = (X_{j1} \times \text{idf}_1, X_{j2} \times \text{idf}_2, \ldots, X_{jJ} \times \text{idf}_J)$$

How Does This Matter For Measuring Similarity/Dissimilarity?
Inner Product

$$\mathbf{X}_{i,\text{idf}} \cdot \mathbf{X}_{j,\text{idf}} = (\mathbf{X}_i \times \textbf{idf})^{'}(\mathbf{X}_j \times \textbf{idf})$$

Weighting Words: TF-IDF

$$\mathbf{X}_{i,\text{idf}} \equiv \underbrace{\mathbf{X}_i}_{\text{tf}} \times \textbf{idf} = (X_{i1} \times \text{idf}_1, X_{i2} \times \text{idf}_2, \ldots, X_{iJ} \times \text{idf}_J)$$

$$\mathbf{X}_{j,\text{idf}} \equiv \mathbf{X}_j \times \textbf{idf} = (X_{j1} \times \text{idf}_1, X_{j2} \times \text{idf}_2, \ldots, X_{jJ} \times \text{idf}_J)$$

How Does This Matter For Measuring Similarity/Dissimilarity?

Inner Product

$$\begin{aligned}
\mathbf{X}_{i,\text{idf}} \cdot \mathbf{X}_{j,\text{idf}} &= (\mathbf{X}_i \times \textbf{idf})'(\mathbf{X}_j \times \textbf{idf}) \\
&= (\text{idf}_1^2 \times X_{i1} \times X_{j1}) + (\text{idf}_2^2 \times X_{i2} \times X_{j2}) + \\
&\quad \ldots + (\text{idf}_J^2 \times X_{iJ} \times X_{jJ})
\end{aligned}$$

Weighting Words: Inner Product

Define:

Weighting Words: Inner Product

Define:

$$\boldsymbol{\Sigma} = \begin{pmatrix} \text{idf}_1^2 & 0 & 0 & \ldots & 0 \\ 0 & \text{idf}_2^2 & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & \text{idf}_J^2 \end{pmatrix}$$

Weighting Words: Inner Product

Define:

$$\boldsymbol{\Sigma} = \begin{pmatrix} \text{idf}_1^2 & 0 & 0 & \ldots & 0 \\ 0 & \text{idf}_2^2 & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & \text{idf}_J^2 \end{pmatrix}$$

If we use tf-idf for our documents, then

Weighting Words: Inner Product

Define:

$$\boldsymbol{\Sigma} = \begin{pmatrix} \text{idf}_1^2 & 0 & 0 & \dots & 0 \\ 0 & \text{idf}_2^2 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \text{idf}_J^2 \end{pmatrix}$$

If we use tf-idf for our documents, then

$$\begin{aligned} d_2(\boldsymbol{X}_i, \boldsymbol{X}_j) &= \sqrt{\sum_{m=1}^{J} (x_{im,\text{idf}} - x_{jm,\text{idf}})^2} \\ &= \sqrt{(\boldsymbol{X}_i - \boldsymbol{X}_j)' \boldsymbol{\Sigma} (\boldsymbol{X}_i - \boldsymbol{X}_j)} \end{aligned}$$

Final Product

Applying some measure of distance, similarity (if symmetric) yields:

$$\mathbf{D} = \begin{pmatrix} 0 & d(1,2) & d(1,3) & \ldots & d(1,N) \\ d(2,1) & 0 & d(2,3) & \ldots & d(2,N) \\ d(3,1) & d(3,2) & 0 & \ldots & d(3,N) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ d(N,1) & d(N,2) & d(N,3) & \ldots & 0 \end{pmatrix}$$

Lower Triangle contains unique information $N(N-1)/2$

Spirling and Indian Treaties

Spirling (2013): model Treaties between US and Native Americans

Spirling and Indian Treaties

Spirling (2013): model Treaties between US and Native Americans
Why?

Spirling and Indian Treaties

Spirling (2013): model Treaties between US and Native Americans

Why?

- American political development

Spirling and Indian Treaties

Spirling (2013): model Treaties between US and Native Americans
Why?

- American political development
- IR Theories of Treaties and Treaty Violations

Spirling and Indian Treaties

Spirling (2013): model Treaties between US and Native Americans
Why?

- American political development
- IR Theories of Treaties and Treaty Violations
- Comparative studies of indigenous/colonialist interaction

Spirling and Indian Treaties

Spirling (2013): model Treaties between US and Native Americans
Why?

- American political development
- IR Theories of Treaties and Treaty Violations
- Comparative studies of indigenous/colonialist interaction
- Political Science question: how did Native Americans lose land so quickly?

Spirling and Indian Treaties

Spirling (2013): model Treaties between US and Native Americans
Why?

- American political development
- IR Theories of Treaties and Treaty Violations
- Comparative studies of indigenous/colonialist interaction
- Political Science question: how did Native Americans lose land so quickly?

Paper does a lot. We're going to focus on

Spirling and Indian Treaties

Spirling (2013): model Treaties between US and Native Americans
Why?

- American political development
- IR Theories of Treaties and Treaty Violations
- Comparative studies of indigenous/colonialist interaction
- Political Science question: how did Native Americans lose land so quickly?

Paper does a lot. We're going to focus on

- Today: Text representation and similarity calculation

Spirling and Indian Treaties

Spirling (2013): model Treaties between US and Native Americans

Why?

- American political development
- IR Theories of Treaties and Treaty Violations
- Comparative studies of indigenous/colonialist interaction
- Political Science question: how did Native Americans lose land so quickly?

Paper does a lot. We're going to focus on

- Today: Text representation and similarity calculation
- Tuesday: Projecting to low dimensional space

Spirling and Indian Treaties

How do we preserve word order and semantic language?
After stemming, stopping, bag of wording:

- `Peace Between Us`
- `No Peace Between Us`

are identical.
Spirling uses complicated representation of texts to preserve word order $\rightsquigarrow$ broad application

`Peace Between Us`

`Analyzes K-substrings`

Spirling and Indian Treaties

How do we preserve word order and semantic language?
After stemming, stopping, bag of wording:

```
- Peace Between Us

- No Peace Between Us
```

are identical.
Spirling uses complicated representation of texts to preserve word order $\rightsquigarrow$ broad application

`Peace Between Us`

`Analyzes K-substrings`

Spirling and Indian Treaties

How do we preserve word order and semantic language?
After stemming, stopping, bag of wording:

```
- Peace Between Us
- No Peace Between Us
```

are identical.
Spirling uses complicated representation of texts to preserve word order $\rightsquigarrow$ broad application

`Peace Between Us`

`Analyzes K-substrings`

Spirling and Indian Treaties

How do we preserve word order and semantic language?
After stemming, stopping, bag of wording:

```
- Peace Between Us
- No Peace Between Us
```

are identical.
Spirling uses complicated representation of texts to preserve word order $\rightsquigarrow$ broad application

`Peace Between Us`

`Analyzes K-substrings`

Spirling and Indian Treaties

How do we preserve word order and semantic language?
After stemming, stopping, bag of wording:

```
- Peace Between Us
- No Peace Between Us
```

are identical.
Spirling uses complicated representation of texts to preserve word order $\rightsquigarrow$ broad application

`Peace Between Us`

`Analyzes K-substrings`

Spirling and Indian Treaties

How do we preserve word order and semantic language?
After stemming, stopping, bag of wording:

- `Peace Between Us`
- `No Peace Between Us`

are identical.
Spirling uses complicated representation of texts to preserve word order $\rightsquigarrow$ broad application

`Peace Between Us`

`Analyzes K-substrings`

Spirling and Indian Treaties

How do we preserve word order and semantic language?
After stemming, stopping, bag of wording:

```
- Peace Between Us

- No Peace Between Us
```

are identical.
Spirling uses complicated representation of texts to preserve word order $\rightsquigarrow$ broad application

`Peace Between Us`

`Analyzes K-substrings`

Spirling and Indian Treaties

How do we preserve word order and semantic language?
After stemming, stopping, bag of wording:

```
- Peace Between Us
- No Peace Between Us
```

are identical.
Spirling uses complicated representation of texts to preserve word order $\rightsquigarrow$ broad application

`Peace Between Us`

`Analyzes K-substrings`

Spirling and Indian Treaties

How do we preserve word order and semantic language?
After stemming, stopping, bag of wording:

```
- Peace Between Us
- No Peace Between Us
```

are identical.
Spirling uses complicated representation of texts to preserve word order $\rightsquigarrow$ broad application

`Peace Between Us`

`Analyzes K-substrings`

Spirling and Indian Treaties

How do we preserve word order and semantic language?
After stemming, stopping, bag of wording:

```
- Peace Between Us
- No Peace Between Us
```

are identical.
Spirling uses complicated representation of texts to preserve word order $\rightsquigarrow$ broad application

`Peace Between Us`

`Analyzes K-substrings`

Spirling and Indian Treaties

How do we preserve word order and semantic language?
After stemming, stopping, bag of wording:

```
- Peace Between Us
- No Peace Between Us
```

are identical.
Spirling uses complicated representation of texts to preserve word order $\rightsquigarrow$ broad application

`Peace Between Us`

`Analyzes K-substrings`

Spirling and Indian Treaties

How do we preserve word order and semantic language?

After stemming, stopping, bag of wording:

```
- Peace Between Us
- No Peace Between Us
```

are identical.

Spirling uses complicated representation of texts to preserve word order $\rightsquigarrow$ broad application

`Peace Between Us`

`Analyzes K-substrings`

Spirling and Indian Treaties

How do we preserve word order and semantic language?
After stemming, stopping, bag of wording:

```
- Peace Between Us
- No Peace Between Us
```

are identical.
Spirling uses complicated representation of texts to preserve word order $\rightsquigarrow$ broad application

`Peace Between Us`

`Analyzes K-substrings`

Kernel Trick

Kernel Trick

- **Kernel Methods**: Represent texts, measure similarity

Kernel Trick

- Kernel Methods: Represent texts, measure similarity simultaneously

Kernel Trick

- Kernel Methods: Represent texts, measure similarity simultaneously
- Compare only substrings in both documents (without explicitly quantifying entire documents)

Kernel Trick

- Kernel Methods: Represent texts, measure similarity simultaneously
- Compare only substrings in both documents (without explicitly quantifying entire documents)
- Problem solved:

Kernel Trick

- Kernel Methods: Represent texts, measure similarity simultaneously
- Compare only substrings in both documents (without explicitly quantifying entire documents)
- Problem solved:
    - Arthur gives all his money to Justin

Kernel Trick

- Kernel Methods: Represent texts, measure similarity simultaneously
- Compare only substrings in both documents (without explicitly quantifying entire documents)
- Problem solved:
  - Arthur gives all his money to Justin
  - Justin gives all his money to Arthur

Kernel Trick

- Kernel Methods: Represent texts, measure similarity simultaneously
- Compare only substrings in both documents (without explicitly quantifying entire documents)
- Problem solved:
    - Arthur gives all his money to Justin
    - Justin gives all his money to Arthur
    - Discard word order: same sentence

Kernel Trick

- Kernel Methods: Represent texts, measure similarity simultaneously
- Compare only substrings in both documents (without explicitly quantifying entire documents)
- Problem solved:
    - Arthur gives all his money to Justin
    - Justin gives all his money to Arthur
    - Discard word order: same sentence Kernel : different sentences.

Kernel Trick

- Kernel Methods: Represent texts, measure similarity simultaneously
- Compare only substrings in both documents (without explicitly quantifying entire documents)
- Problem solved:
  - Arthur gives all his money to Justin
  - Justin gives all his money to Arthur
  - Discard word order: same sentence Kernel : different sentences.

Uses Kernel methods to measure similarity

Kernel Trick

- Kernel Methods: Represent texts, measure similarity simultaneously
- Compare only substrings in both documents (without explicitly quantifying entire documents)
- Problem solved:
    - Arthur gives all his money to Justin
    - Justin gives all his money to Arthur
    - Discard word order: same sentence Kernel : different sentences.

Uses Kernel methods to measure similarity

Kernel Trick

- Kernel Methods: Represent texts, measure similarity simultaneously
- Compare only substrings in both documents (without explicitly quantifying entire documents)
- Problem solved:
  - Arthur gives all his money to Justin
  - Justin gives all his money to Arthur
  - Discard word order: same sentence Kernel : different sentences.

Uses Kernel methods to measure similarity

Kernel Trick

- Kernel Methods: Represent texts, measure similarity simultaneously
- Compare only substrings in both documents (without explicitly quantifying entire documents)
- Problem solved:
  - Arthur gives all his money to Justin
  - Justin gives all his money to Arthur
  - Discard word order: same sentence Kernel : different sentences.

Uses Kernel methods to measure similarity

Kernel Trick

- Kernel Methods: Represent texts, measure similarity simultaneously
- Compare only substrings in both documents (without explicitly quantifying entire documents)
- Problem solved:
  - Arthur gives all his money to Justin
  - Justin gives all his money to Arthur
  - Discard word order: same sentence Kernel : different sentences.

Uses Kernel methods to measure similarity

Similarity and Dissimilarity of Many Things

Throughout the course we'll measure similarity between documents
We'll also (implicitly) study similarity of probability distributions
Develop a measure of distribution dissimilarity

Similarity of Probability Distributions

Definition

Suppose P is a continuous random variable with density $p : \Re \rightarrow \Re$ and Q is a continuous random variable with density $q : \Re \rightarrow q$.
We can define the KL-Divergence between P and Q as

$$KL(P||Q) \quad = \quad \int_{-\infty}^{\infty} p(x) \log \frac{p(x)}{q(x)} dx$$

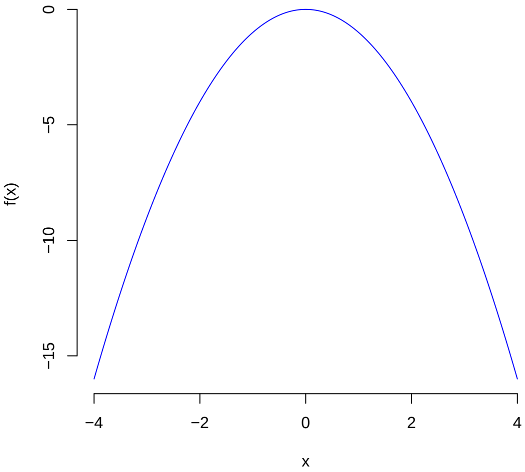
Assessing Similarity of Other Things

KL-divergence measures dissimilarity between two distributions.

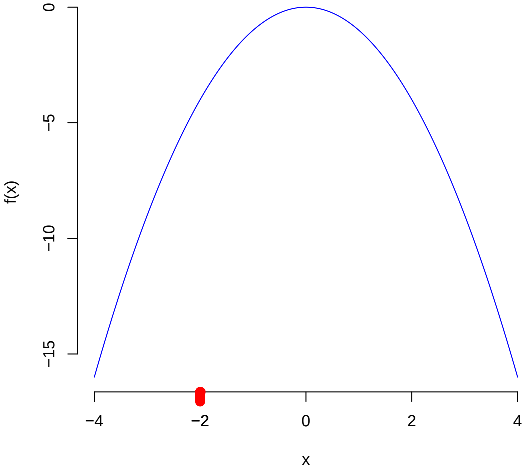
Consider a function. $f(x) = -x^2$.

Consider a function. $f(x) = -x^2$.
Maps numbers to other numbers.

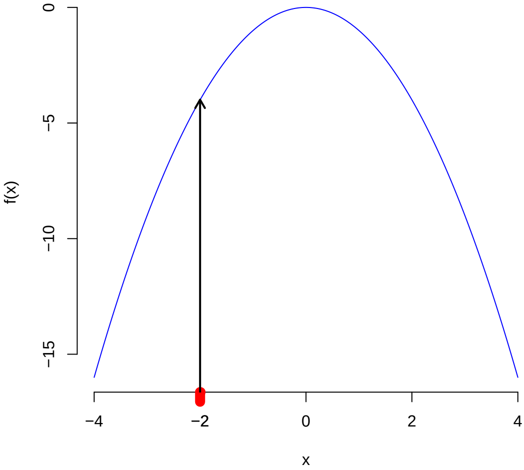
Consider a function. $f(x) = -x^2$.
Maps numbers to other numbers.



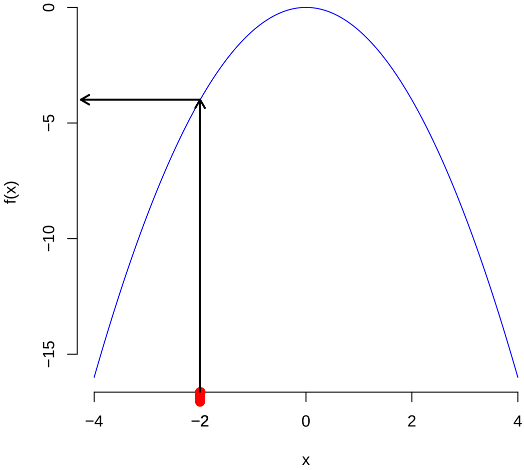
Take some input (-2 here)



Then obtain the value of $f(-2)$



Then obtain the value of $f(-2) = -4$



KL($q||p$) is a functional.

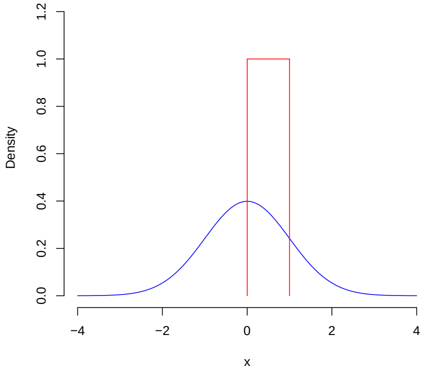
KL$(q||p)$ is a functional. A functional takes functions as inputs, returns a real number.

KL($q||p$) is a functional. A functional takes functions as inputs, returns a real number.
KL($q||p$) maps from sets of distributions $q \in \mathcal{Q}$ and $p \in \mathcal{P}$ to positive real numbers.

KL($q||p$) is a functional. A functional takes functions as inputs, returns a real number.

KL($q||p$) maps from sets of distributions $q \in \mathcal{Q}$ and $p \in \mathcal{P}$ to positive real numbers.

For example, we could set q = Uniform(0,1) and p = Normal(0, 1)

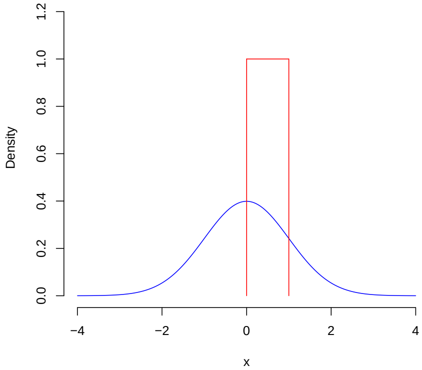


KL($q||p$) is a functional. A functional takes functions as inputs, returns a real number.

KL($q||p$) maps from sets of distributions $q \in \mathcal{Q}$ and $p \in \mathcal{P}$ to positive real numbers.

For example, we could set q = Uniform(0,1) and p = Normal(0, 1)

KL(Uniform(0,1)||Normal(0,1)) = 1.09



If q and p are the same distribution then $\text{KL}(q||p) = 0$.

If q and p are the same distribution then $\text{KL}(q||p) = 0$.
Variational Approximation (topic models!): approximate one distribution p, with another, simpler distribution q.

If q and p are the same distribution then $\text{KL}(q||p) = 0$.
Variational Approximation (topic models!): approximate one distribution p, with another, simpler distribution q.
Then make this approximation the best possible–minimize the KL-divergence.

A simple example.

A simple example.

Approximate a Normal(0,1) with symmetric Uniform distribution, Uniform(-b, b).

A simple example.

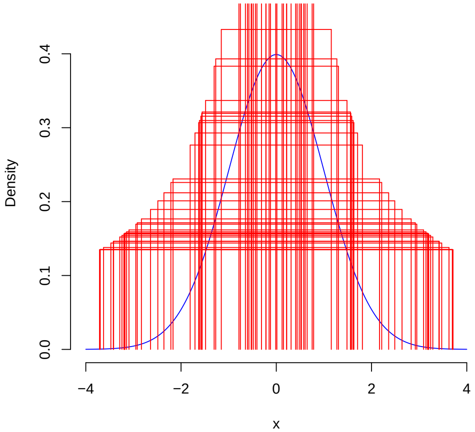
Approximate a Normal(0,1) with symmetric Uniform distribution, Uniform(-b, b).

Choose b to min. KL(Uniform(-b, b)|| Normal(0,1))

A simple example.

Approximate a Normal(0,1) with symmetric Uniform distribution, Uniform(-b, b).

Choose b to min. KL(Uniform(-b, b)|| Normal(0,1))

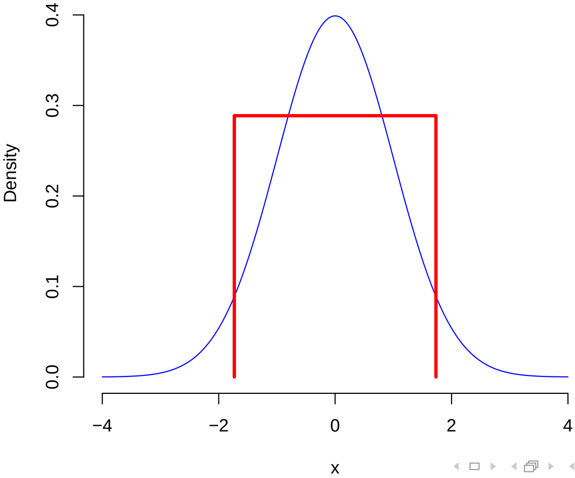


Answer:

Answer:
$b = \sqrt{3}$

Answer:

$b = \sqrt{3}$



1) Documents in vector space $\rightsquigarrow$ geometry of texts
2) Many methods to measure similarity and dissimilarity