

Text as Data

Justin Grimmer

Professor
Department of Political Science
Stanford University

September 10th, 2019

Texts and Geometry

Consider a document-term matrix

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

Texts and Geometry

Consider a document-term matrix

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

Suppose documents live in a **space**

Texts and Geometry

Consider a document-term matrix

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

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

Texts and Geometry

Consider a document-term matrix

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

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

- Provides a **geometry**

Texts and Geometry

Consider a document-term matrix

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

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

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

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

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

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**
- Building block for clustering, supervised learning, and scaling

Texts in Space

Texts in Space

Doc1 = (1, 1, 3, . . . , 5)

Texts in Space

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

$$\text{Doc2} = (2, 0, 0, \dots, 1)$$

Texts in Space

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

$$\text{Doc2} = (2, 0, 0, \dots, 1)$$

$$\text{Doc1}, \text{Doc2} \in \Re^J$$

Texts in Space

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

$$\text{Doc2} = (2, 0, 0, \dots, 1)$$

$$\text{Doc1}, \text{Doc2} \in \Re^J$$

Inner Product between documents:

Texts in Space

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

$$\text{Doc2} = (2, 0, 0, \dots, 1)$$

$$\text{Doc1}, \text{Doc2} \in \Re^J$$

Inner Product between documents:

$$\text{Doc1} \cdot \text{Doc2} = (1, 1, 3, \dots, 5)' (2, 0, 0, \dots, 1)$$

Texts in Space

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

$$\text{Doc2} = (2, 0, 0, \dots, 1)$$

$$\text{Doc1}, \text{Doc2} \in \mathbb{R}^J$$

Inner Product between documents:

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

Texts in Space

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

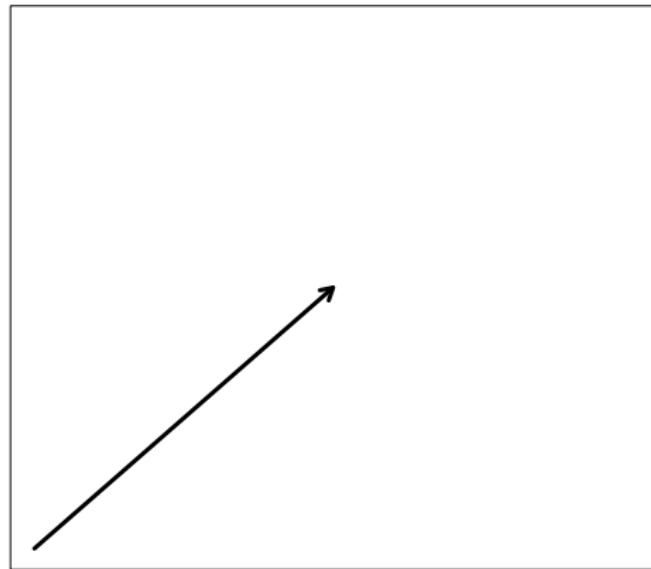
$$\text{Doc2} = (2, 0, 0, \dots, 1)$$

$$\text{Doc1}, \text{Doc2} \in \mathbb{R}^J$$

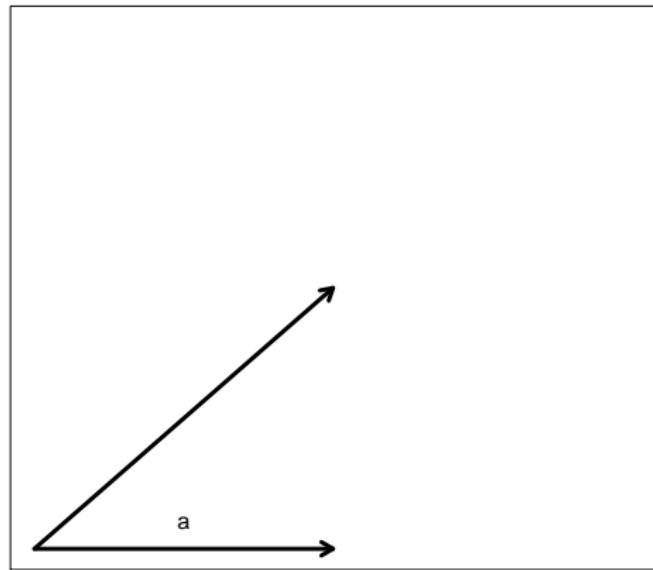
Inner Product between documents:

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

Vector Length

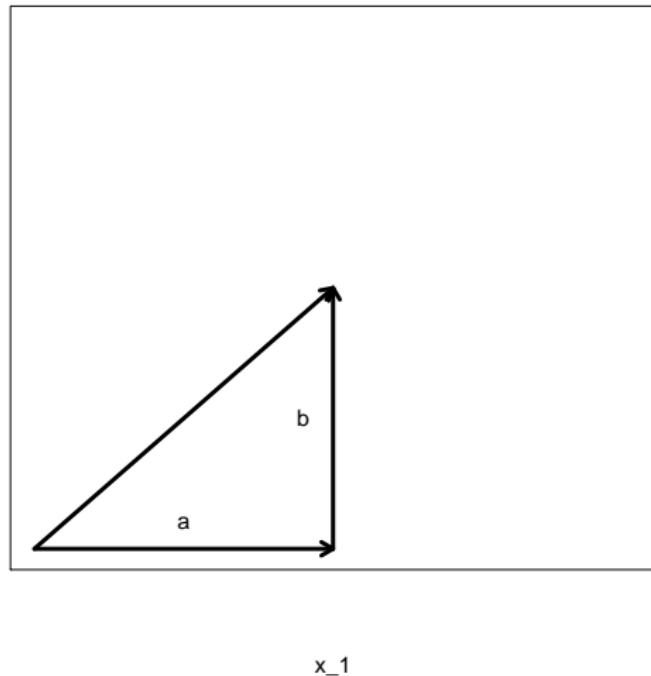


Vector Length



- Pythagorean Theorem:
Side with length a

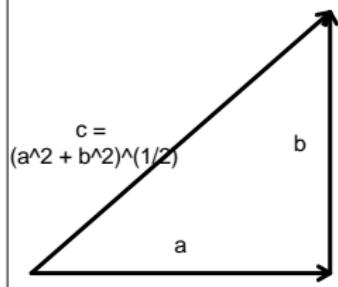
Vector Length



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

Vector Length

x_2

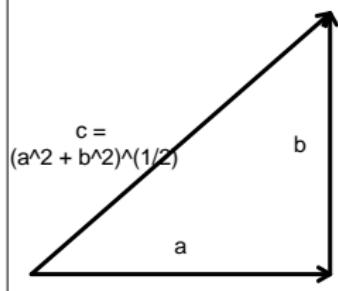


x_1

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

Vector Length

x_2



x_1

- Pythagorean 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 $\mathbf{v} \in \mathbb{R}^J$. Then, we will define its **length** as

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

Measures of Dissimilarity

Initial guess \rightsquigarrow Distance metrics

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



Measures of Dissimilarity

Initial guess \rightsquigarrow Distance metrics

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

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



Measures of Dissimilarity

Initial guess \rightsquigarrow Distance metrics

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

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



Measures of Dissimilarity

Initial guess \rightsquigarrow Distance metrics

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

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



Measures of Dissimilarity

Initial guess \rightsquigarrow Distance metrics

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

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



Measures of Dissimilarity

Initial guess \rightsquigarrow **Distance metrics**

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

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

Explore **distance** functions to compare documents \rightsquigarrow

Measures of Dissimilarity

Initial guess \rightsquigarrow **Distance metrics**

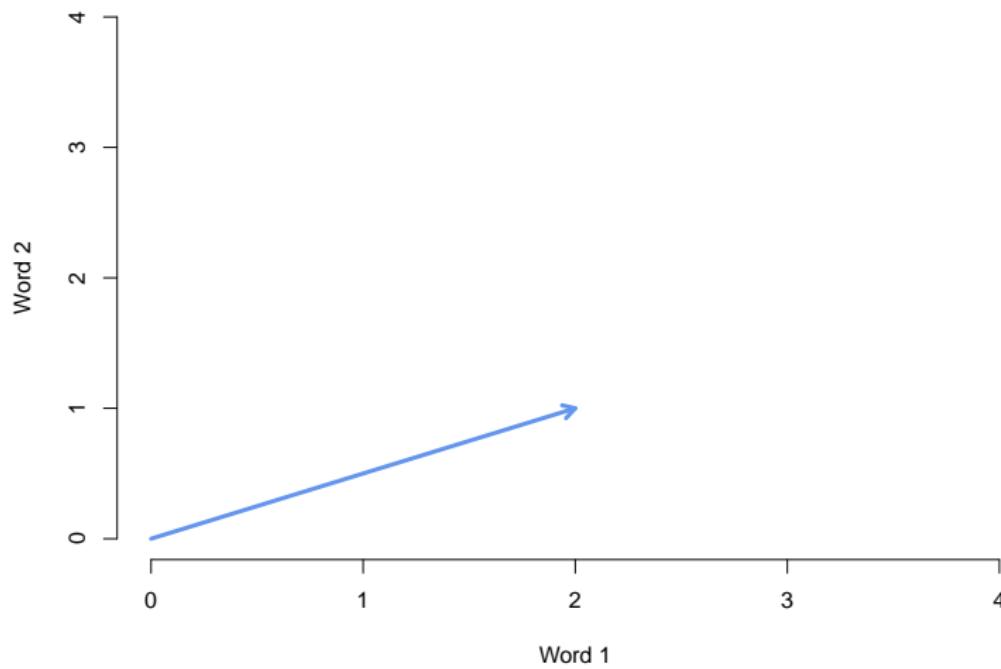
Properties of a metric: (distance function) $d(\cdot, \cdot)$. Consider arbitrary documents $\mathbf{X}_i, \mathbf{X}_j, \mathbf{X}_k$

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

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

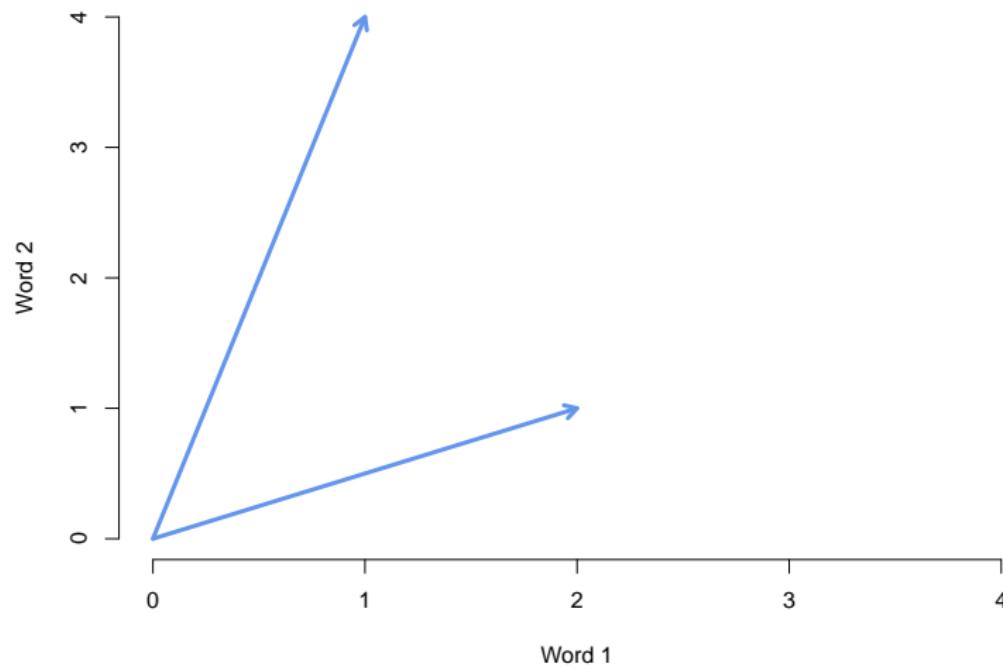
Measuring the Distance Between Documents

Euclidean Distance



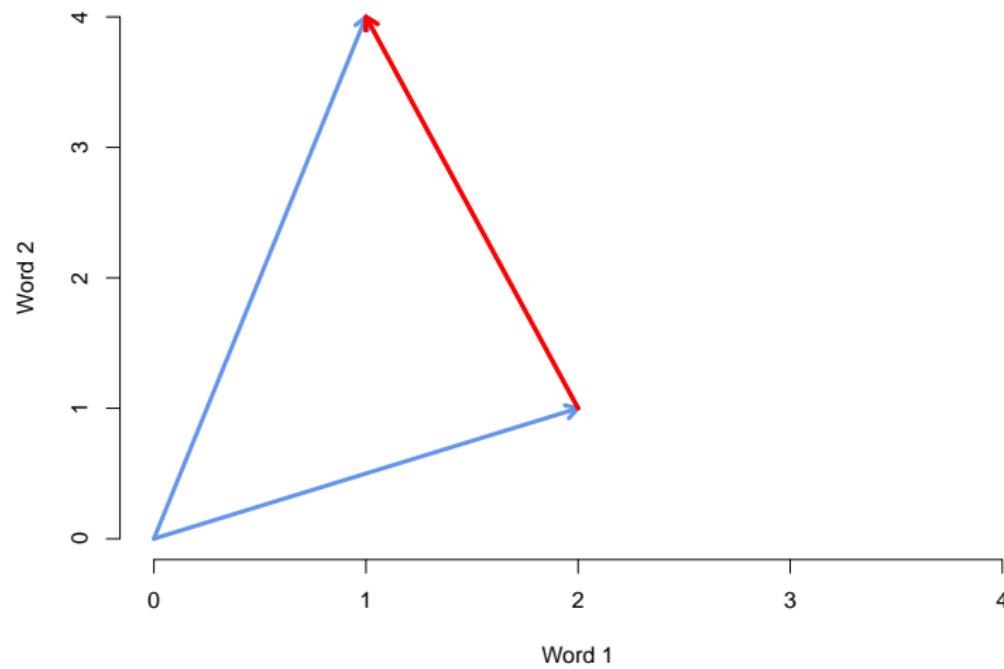
Measuring the Distance Between Documents

Euclidean Distance



Measuring the Distance Between Documents

Euclidean Distance



Measuring the Distance Between Documents

Definition

The Euclidean distance between documents \mathbf{X}_i and \mathbf{X}_j as

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

Measuring the Distance Between Documents

Definition

The Euclidean distance between documents \mathbf{X}_i and \mathbf{X}_j as

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

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

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

Measuring Similarity (and removing document length)

Measuring Similarity (and removing document length)

What properties should similarity measure have?

Measuring Similarity (and removing document length)

What properties should similarity measure have?

- Maximum: document with itself

Measuring Similarity (and removing document length)

What properties should similarity measure have?

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

Measuring Similarity (and removing document length)

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 (and removing document length)

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
- ? $s(a, b) = s(b, a)$.

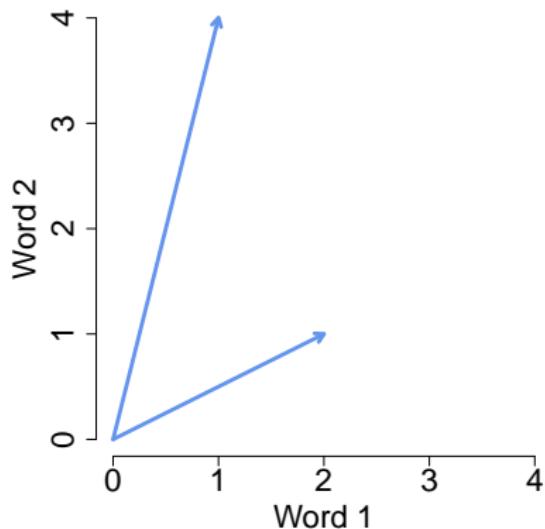
Measuring Similarity (and removing document length)

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
- ? $s(a, b) = s(b, a)$.

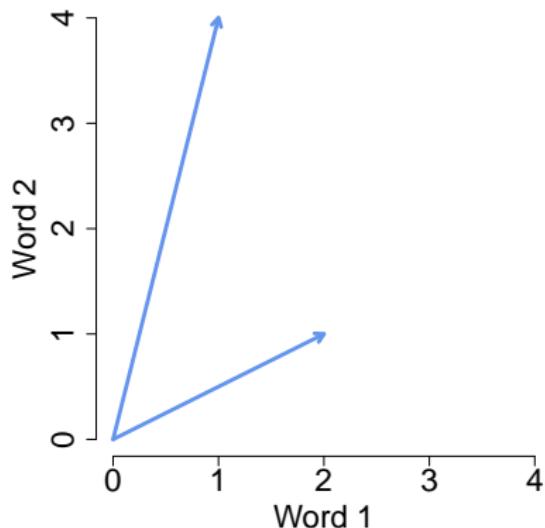
How should additional words be treated?

Measuring Similarity



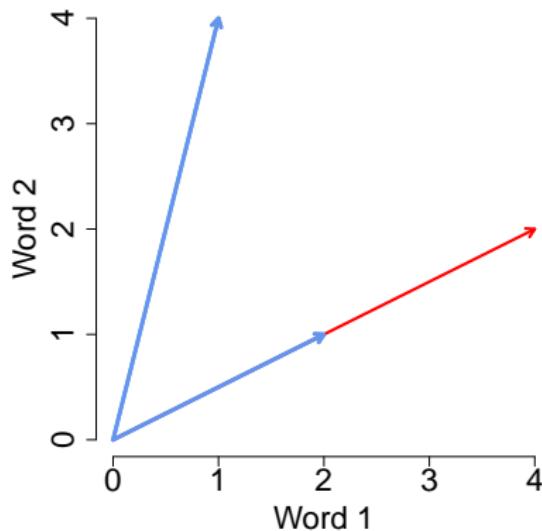
Measure 1: Inner product

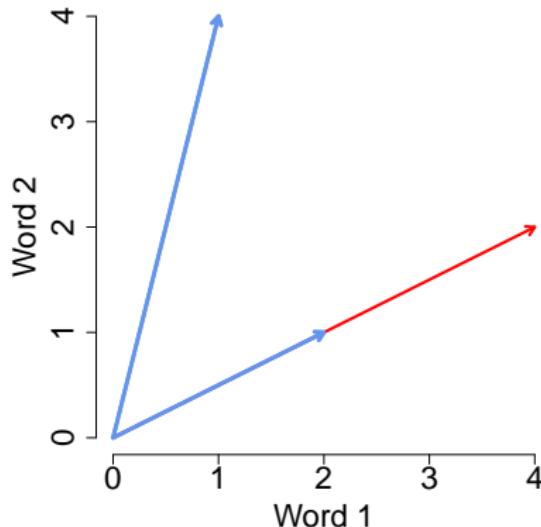
Measuring Similarity



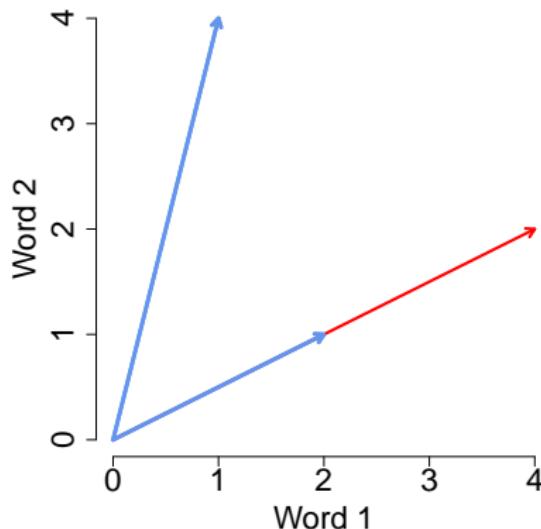
Measure 1: Inner product

$$(2, 1)' \cdot (1, 4) = 6$$



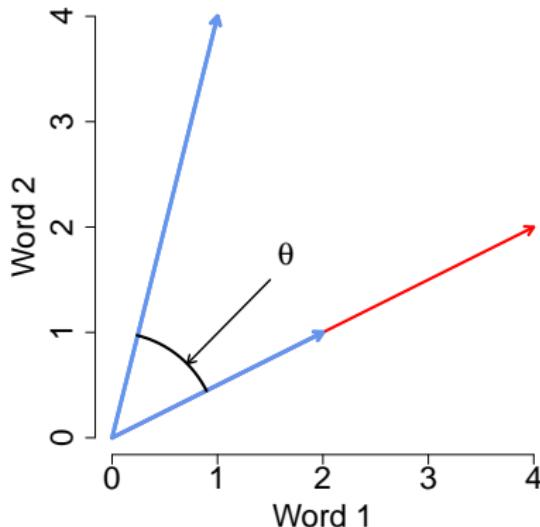


Problem(?): length dependent



Problem(?): length dependent

$$(4, 2)'(1, 4) = 12$$



Problem(?): length dependent

$$(4, 2)'(1, 4) = 12$$

$$a \cdot b = \|a\| \times \|b\| \times \cos \theta$$

Cosine Similarity

Cosine Similarity

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

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

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

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

$$\frac{(2, 1)}{\|(2, 1)\|} = (0.89, 0.45)$$

Cosine Similarity

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

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

$$\frac{(2, 1)}{\|(2, 1)\|} = (0.89, 0.45)$$

$$\frac{(1, 4)}{\|(1, 4)\|} = (0.24, 0.97)$$

Cosine Similarity

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

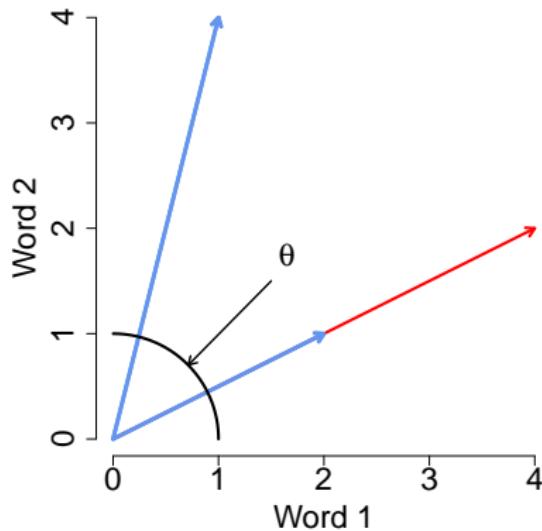
$$\frac{(4, 2)}{\|(4, 2)\|} = (0.89, 0.45)$$

$$\frac{(2, 1)}{\|(2, 1)\|} = (0.89, 0.45)$$

$$\frac{(1, 4)}{\|(1, 4)\|} = (0.24, 0.97)$$

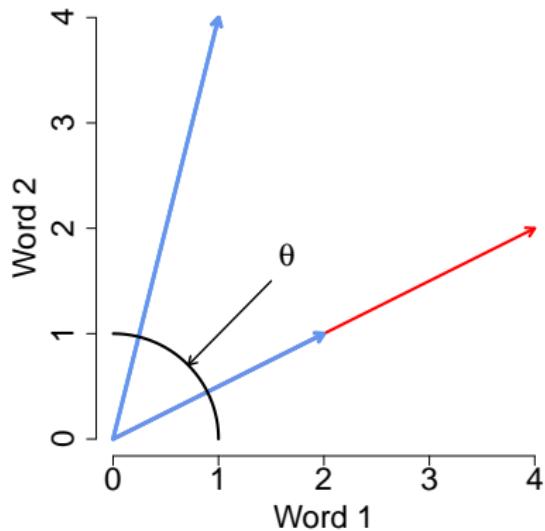
$$(0.89, 0.45)' (0.24, 0.97) = 0.65$$

Cosine Similarity



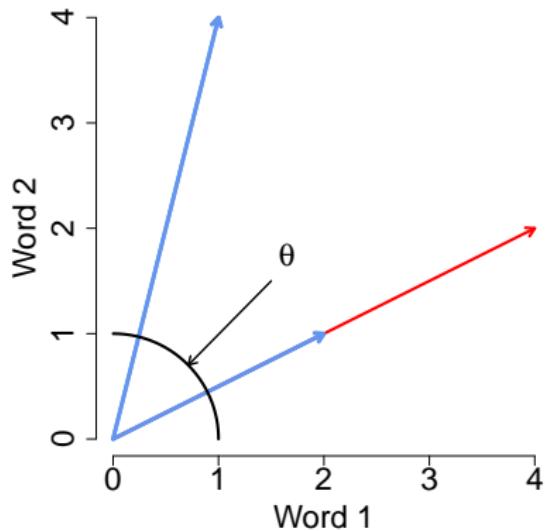
$\cos \theta$: removes document length from similarity measure

Cosine Similarity



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

Cosine Similarity



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

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:

n_j = No. documents in which word j 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:

n_j = No. documents in which word j occurs

$$\text{idf}_j = \log \frac{N}{n_j}$$

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:

n_j = No. documents in which word j occurs

$$\text{idf}_j = \log \frac{N}{n_j}$$

idf = $(\text{idf}_1, \text{idf}_2, \dots, \text{idf}_J)$

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

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

Weighting Words: TF-IDF

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

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

Weighting Words: TF-IDF

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

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

How Does This Matter For Measuring Similarity/Dissimilarity?

Weighting Words: TF-IDF

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

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

How Does This Matter For Measuring Similarity/Dissimilarity?

Inner Product

Weighting Words: TF-IDF

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

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

How Does This Matter For Measuring Similarity/Dissimilarity?

Inner Product

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

Weighting Words: TF-IDF

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

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

How Does This Matter For Measuring Similarity/Dissimilarity?

Inner Product

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

Weighting Words: Inner Product

Define:

Weighting Words: Inner Product

Define:

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

Weighting Words: Inner Product

Define:

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

If we use tf-idf for our documents, then

Weighting Words: Inner Product

Define:

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

If we use tf-idf for our documents, then

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

Final Product

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

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

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

Learning relationships to classify documents

Types of Classification Problems

Topic: What is this text about?

Types of Classification Problems

Topic: What is this text about?

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

Types of Classification Problems

Topic: What is this text about?

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

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

Types of Classification Problems

Topic: What is this text about?

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

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

- Positions on legislation
⇒ { Support, Ambiguous, Oppose }
- Positions on Court Cases
⇒ { Agree with Court, Disagree with Court }
- Liberal/Conservative Blog Posts
⇒ { Liberal, Middle, Conservative, No Ideology Expressed }

Types of Classification Problems

Topic: What is this text about?

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

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

- Positions on legislation
⇒ { Support, Ambiguous, Oppose }
- Positions on Court Cases
⇒ { Agree with Court, Disagree with Court }
- Liberal/Conservative Blog Posts
⇒ { Liberal, Middle, Conservative, No Ideology Expressed }

Style/Tone: How is it said?

Types of Classification Problems

Topic: What is this text about?

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

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

- Positions on legislation
⇒ { Support, Ambiguous, Oppose }
- Positions on Court Cases
⇒ { Agree with Court, Disagree with Court }
- Liberal/Conservative Blog Posts
⇒ { Liberal, Middle, Conservative, No Ideology Expressed }

Style/Tone: How is it said?

- Taunting in floor statements
⇒ { Partisan Taunt, Intra party taunt, Agency taunt, ... }
- Negative campaigning
⇒ { Negative ad, Positive ad }

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 $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{iJ})$.

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 $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{iJ})$.

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

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 $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{iJ})$.

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

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 $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{iJ})$.

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

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 $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{iJ})$.

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

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 $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{iJ})$.

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

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 $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{iJ})$.

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

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 $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{iJ})$.

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

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:

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

Mean Square Error

Suppose θ is some value of the true parameter

Bias:

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

We may care about average distance from truth

Mean Square Error

Suppose θ is some value of the true parameter

Bias:

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

We may care about average distance from truth

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

Mean Square Error

Suppose θ is some value of the true parameter

Bias:

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

We may care about average distance from truth

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

Mean Square Error

Suppose θ is some value of the true parameter

Bias:

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

We may care about average distance from truth

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

Mean Square Error

Suppose θ is some value of the true parameter

Bias:

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

We may care about average distance from truth

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

Mean Square Error

Suppose θ is some value of the true parameter

Bias:

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

We may care about average distance from truth

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

Mean Square Error

Suppose θ is some value of the true parameter

Bias:

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

We may care about average distance from truth

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

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

Ridge Regression

Penalty for model complexity

Ridge Regression

Penalty for model complexity

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

Ridge Regression

Penalty for model complexity

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

Ridge Regression

Penalty for model complexity

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

Ridge Regression

Penalty for model complexity

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

where:

Ridge Regression

Penalty for model complexity

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

where:

- $\beta_0 \rightsquigarrow$ intercept

Ridge Regression

Penalty for model complexity

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

where:

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

Ridge Regression

Penalty for model complexity

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

where:

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

Ridge Regression \rightsquigarrow Optimization

$$\boldsymbol{\beta}^{\text{Ridge}} = \arg \min_{\boldsymbol{\beta}} \{f(\boldsymbol{\beta}, \mathbf{X}, \mathbf{Y})\}$$

Ridge Regression \rightsquigarrow Optimization

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

Ridge Regression \rightsquigarrow Optimization

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

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

Ridge Regression \rightsquigarrow Optimization

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

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

Ridge Regression \rightsquigarrow Optimization

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

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

Lasso Regression Objective Function

Different Penalty for Model Complexity

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

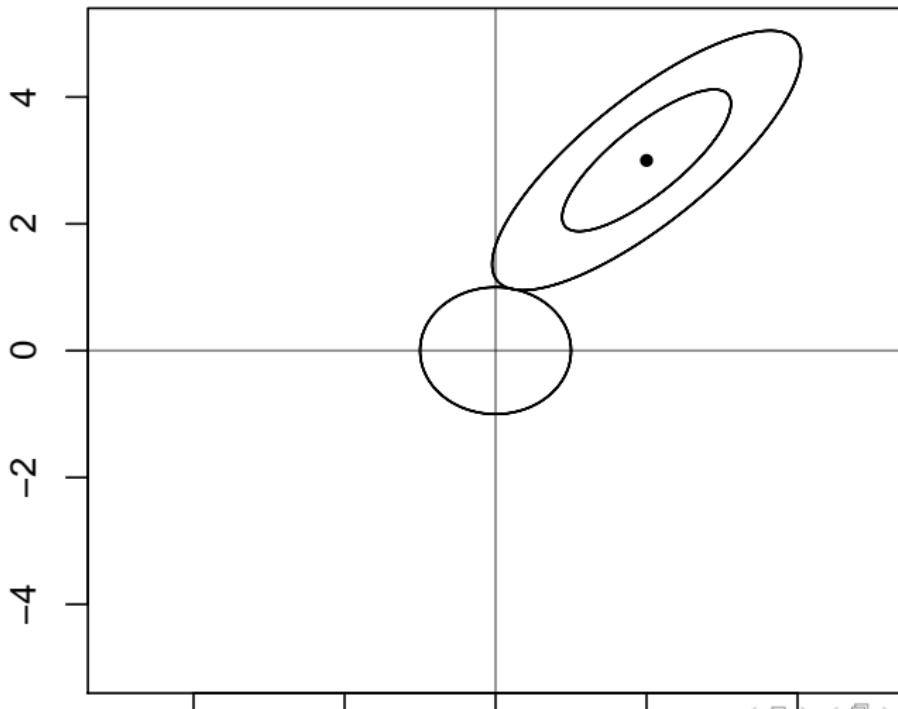
Lasso Regression Objective Function

Different Penalty for Model Complexity

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

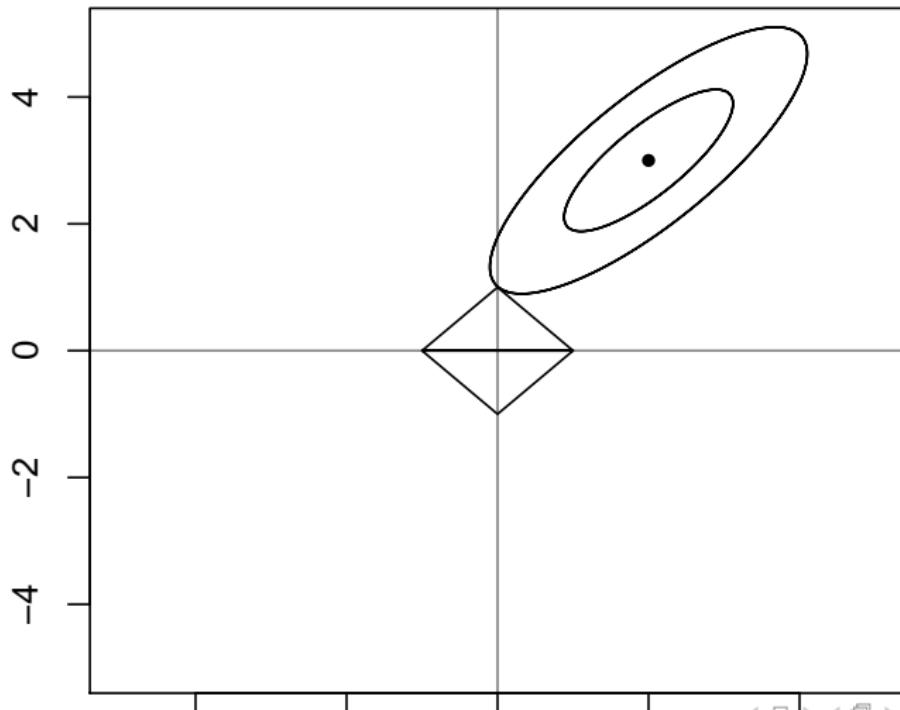
Comparing Ridge and LASSO

Ridge Regression



Comparing Ridge and LASSO

LASSO Regression



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:

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

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:

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

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

Under LASSO

Comparing Ridge and LASSO

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

Under ridge:

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

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

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:

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

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

Under LASSO

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

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

Ridge and LASSO: The Elastic-Net

Combining the two criteria \rightsquigarrow Elastic-Net

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

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

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:

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

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

1 Group2, Group3, Group 4, ..., Group K

Validation ("Test")
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

1 Group2, Group3, Group 4, ..., Group K

2 Group 1, Group3, Group 4, ..., Group K

Validation ("Test")

Group 1

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 $\hat{f}^{-K}(\beta, \mathbf{X})$

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 $\hat{f}^{-K}(\beta, \mathbf{X})$
- 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 $\hat{f}^{-K}(\beta, \mathbf{X})$
- Predict values for K^{th}
- Summarize performance with loss function: $L(\mathbf{Y}_i, \hat{f}^{-k}(\beta, \mathbf{X}))$

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 $\hat{f}^{-K}(\beta, \mathbf{X})$
- Predict values for K^{th}
- Summarize performance with loss function: $L(\mathbf{Y}_i, \hat{f}^{-k}(\beta, \mathbf{X}))$
 - 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 $\hat{f}^{-K}(\beta, \mathbf{X})$
- Predict values for K^{th}
- Summarize performance with loss function: $L(\mathbf{Y}_i, \hat{f}^{-k}(\beta, \mathbf{X}))$
 - Mean square error, Absolute error, Prediction error, ...

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

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 $\hat{f}^{-K}(\beta, \mathbf{X})$
- Predict values for K^{th}
- Summarize performance with loss function: $L(\mathbf{Y}_i, \hat{f}^{-k}(\beta, \mathbf{X}))$
 - Mean square error, Absolute error, Prediction error, ...

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

$$\text{CV(proportions)} =$$

$\frac{1}{K} \sum_{j=1}^K$ 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 $\hat{f}^{-K}(\beta, \mathbf{X})$
- Predict values for K^{th}
- Summarize performance with loss function: $L(\mathbf{Y}_i, \hat{f}^{-k}(\beta, \mathbf{X}))$
 - Mean square error, Absolute error, Prediction error, ...

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

$$\text{CV(proportions)} =$$

$\frac{1}{K} \sum_{j=1}^K$ 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 =  
    binomial) ##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 = class) ##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
```

Ensemble Learning: Intuition

Heuristic (upon which we'll improve):

Ensemble Learning: Intuition

Heuristic (upon which we'll improve): if regressions are **accurate** and **diverse** → ensemble methods improve

Ensemble Learning: Intuition

Heuristic (upon which we'll improve): if regressions are **accurate** and **diverse** → ensemble methods improve

Intuition:

Ensemble Learning: Intuition

Heuristic (upon which we'll improve): if regressions are **accurate** and **diverse** → ensemble methods improve

Intuition:

- Classify observations into two categories (Category 1, Category 2).

Ensemble Learning: Intuition

Heuristic (upon which we'll improve): if regressions are **accurate** and **diverse** → ensemble methods improve

Intuition:

- Classify observations into two categories (Category 1, Category 2).
- True labels: evenly distributed across two categories

Ensemble Learning: Intuition

Heuristic (upon which we'll improve): if regressions are **accurate** and **diverse** → ensemble methods improve

Intuition:

- Classify observations into two categories (Category 1, Category 2).
- True labels: evenly distributed across two categories
- Three classifiers with 75% accuracy, but independent

Ensemble Learning: Intuition

Heuristic (upon which we'll improve): if regressions are **accurate** and **diverse** → ensemble methods improve

Intuition:

- Classify observations into two categories (Category 1, Category 2).
- True labels: evenly distributed across two categories
- Three classifiers with 75% accuracy, but independent
- Implement majority voting rule

Ensemble Learning: Intuition

Heuristic (upon which we'll improve): if regressions are **accurate** and **diverse** → ensemble methods improve

Intuition:

- Classify observations into two categories (Category 1, Category 2).
- True labels: evenly distributed across two categories
- Three classifiers with 75% accuracy, but independent
- Implement majority voting rule

$$\Pr(\text{Correct Guess} | \text{Votes})$$

Ensemble Learning: Intuition

Heuristic (upon which we'll improve): if regressions are **accurate** and **diverse** → ensemble methods improve

Intuition:

- Classify observations into two categories (Category 1, Category 2).
- True labels: evenly distributed across two categories
- Three classifiers with 75% accuracy, but independent
- Implement majority voting rule

$$\Pr(\text{Correct Guess} | \text{Votes}) = \Pr(3 \text{ correct}) + \Pr(2 \text{ correct})$$

Ensemble Learning: Intuition

Heuristic (upon which we'll improve): if regressions are **accurate** and **diverse** → ensemble methods improve

Intuition:

- Classify observations into two categories (Category 1, Category 2).
- True labels: evenly distributed across two categories
- Three classifiers with 75% accuracy, but independent
- Implement majority voting rule

$$\begin{aligned}\Pr(\text{Correct Guess}|\text{Votes}) &= \Pr(3 \text{ correct}) + \Pr(2 \text{ correct}) \\ &= 0.75^3 + 3 \times (0.75^2 \times 0.25)\end{aligned}$$

Ensemble Learning: Intuition

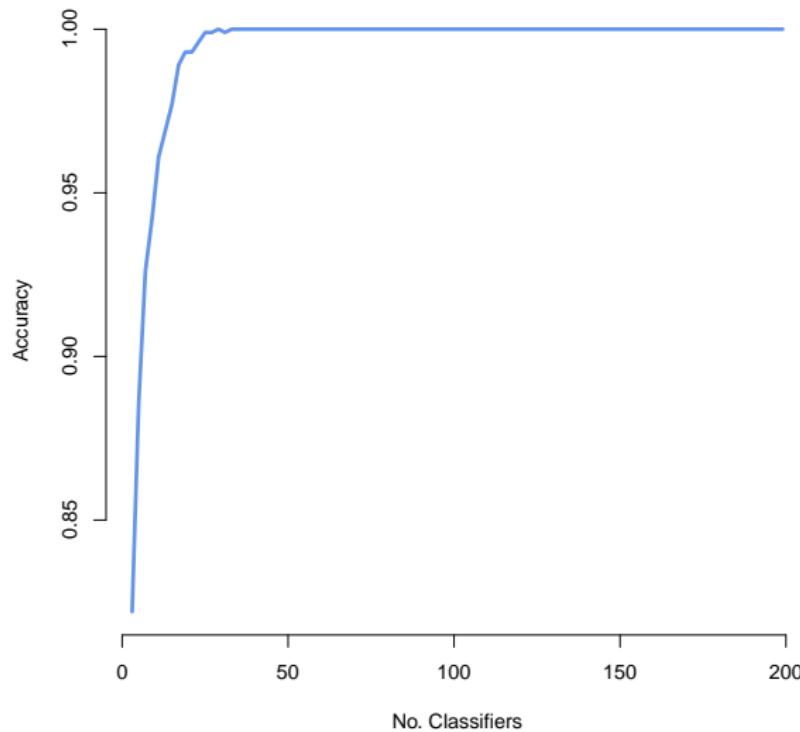
Heuristic (upon which we'll improve): if regressions are **accurate** and **diverse** → ensemble methods improve

Intuition:

- Classify observations into two categories (Category 1, Category 2).
- True labels: evenly distributed across two categories
- Three classifiers with 75% accuracy, but independent
- Implement majority voting rule

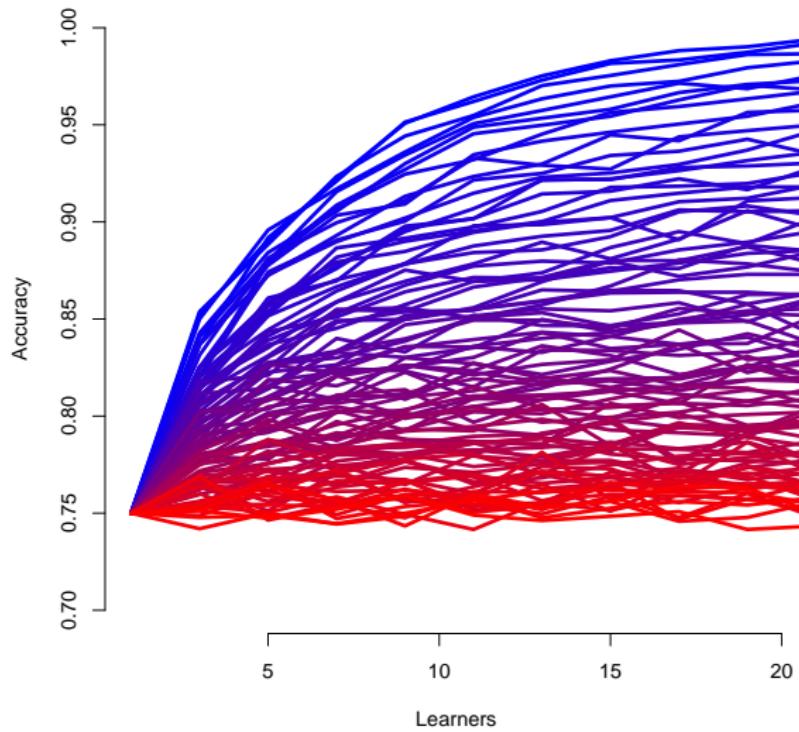
$$\begin{aligned}\Pr(\text{Correct Guess}|\text{Votes}) &= \Pr(3 \text{ correct}) + \Pr(2 \text{ correct}) \\ &= 0.75^3 + 3 \times (0.75^2 \times 0.25) \\ &= 0.844\end{aligned}$$

Ensemble Learning: Intuition



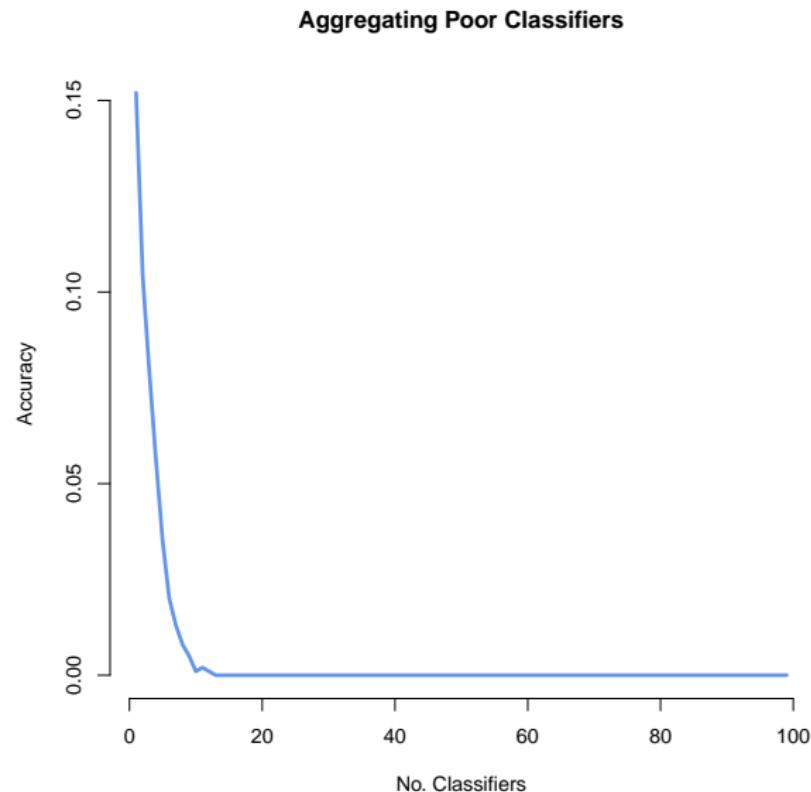
Ensemble Learning: Intuition

Diverse and Accurate matter.



Ensemble Learning: Intuition

Diverse and Accurate matter.



Wisdom of the Crowds:

Goal: estimate an observation's category $\rightsquigarrow Y \in \{0, 1\}$

Wisdom of the Crowds:

Goal: estimate an observation's category $\rightsquigarrow Y \in \{0, 1\}$

Classifiers: (suppose) a sequence of identically distributed (**not necessarily independent**) random variables.

Wisdom of the Crowds:

Goal: estimate an observation's category $\rightsquigarrow Y \in \{0, 1\}$

Classifiers: (suppose) a sequence of identically distributed (**not necessarily independent**) random variables.

Suppose $Y = 1$

Wisdom of the Crowds:

Goal: estimate an observation's category $\rightsquigarrow Y \in \{0, 1\}$

Classifiers: (suppose) a sequence of identically distributed (**not necessarily independent**) random variables.

Suppose $Y = 1$

Guess from classifier m is B_m with $\Pr(B_i = 1) = p > 0.5$.

Wisdom of the Crowds:

Goal: estimate an observation's category $\rightsquigarrow Y \in \{0, 1\}$

Classifiers: (suppose) a sequence of identically distributed (**not necessarily independent**) random variables.

Suppose $Y = 1$

Guess from classifier m is B_m with $\Pr(B_i = 1) = p > 0.5$.

$$\bar{B} = \sum_{m=1}^M \frac{B_m}{M}$$

Wisdom of the Crowds:

Goal: estimate an observation's category $\rightsquigarrow Y \in \{0, 1\}$

Classifiers: (suppose) a sequence of identically distributed (**not necessarily independent**) random variables.

Suppose $Y = 1$

Guess from classifier m is B_m with $\Pr(B_i = 1) = p > 0.5$.

$$\bar{B} = \sum_{m=1}^M \frac{B_m}{M}$$

Wisdom of crows (Condorcet Jury Theorem)

Wisdom of the Crowds:

Goal: estimate an observation's category $\rightsquigarrow Y \in \{0, 1\}$

Classifiers: (suppose) a sequence of identically distributed (**not necessarily independent**) random variables.

Suppose $Y = 1$

Guess from classifier m is B_m with $\Pr(B_i = 1) = p > 0.5$.

$$\bar{B} = \sum_{m=1}^M \frac{B_m}{M}$$

Wisdom of crows (Condorcet Jury Theorem)

$$\lim_{M \rightarrow \infty} P(\bar{B} > 0.5) = 1$$

Wisdom of the Crowds

Suppose B_m have variance σ^2 and pairwise correlation ρ .

Wisdom of the Crowds

Suppose B_m have variance σ^2 and pairwise correlation ρ .
Then,

Wisdom of the Crowds

Suppose B_m have variance σ^2 and pairwise correlation ρ .
Then,

$$\text{var}(\bar{B})$$

Wisdom of the Crowds

Suppose B_m have variance σ^2 and pairwise correlation ρ .
Then,

$$\text{var}(\bar{B}) = \text{var} \left(\sum_{i=1}^M \frac{B_i}{M} \right)$$

Wisdom of the Crowds

Suppose B_m have variance σ^2 and pairwise correlation ρ .
Then,

$$\begin{aligned}\text{var}(\bar{B}) &= \text{var} \left(\sum_{i=1}^M \frac{B_i}{M} \right) \\ &= \frac{1}{M^2} \sum_{i=1}^M \text{var}(B_i) + \frac{2}{M^2} \sum_{i < j} \text{cov}(B_i, B_j)\end{aligned}$$

Wisdom of the Crowds

Suppose B_m have variance σ^2 and pairwise correlation ρ .
Then,

$$\begin{aligned}\text{var}(\bar{B}) &= \text{var} \left(\sum_{i=1}^M \frac{B_i}{M} \right) \\ &= \frac{1}{M^2} \sum_{i=1}^M \text{var}(B_i) + \frac{2}{M^2} \sum_{i < j} \text{cov}(B_i, B_j) \\ &= \frac{M\sigma^2}{M^2} + \frac{2}{M^2} \rho \sigma^2 \binom{M}{2}\end{aligned}$$

Wisdom of the Crowds

Suppose B_m have variance σ^2 and pairwise correlation ρ .
Then,

$$\begin{aligned}\text{var}(\bar{B}) &= \text{var} \left(\sum_{i=1}^M \frac{B_i}{M} \right) \\ &= \frac{1}{M^2} \sum_{i=1}^M \text{var}(B_i) + \frac{2}{M^2} \sum_{i < j} \text{cov}(B_i, B_j) \\ &= \frac{M\sigma^2}{M^2} + \frac{2}{M^2} \rho \sigma^2 \binom{M}{2} \\ &= \underbrace{\rho \sigma^2}_{\text{Resolve with independence}} + \underbrace{\frac{1 - \rho}{M} \sigma^2}_{\text{Resolve with } \uparrow \text{classifiers}}\end{aligned}$$

Bagging: bootstrap aggregation

Creating Weak Classifiers with resampling:

Bagging: bootstrap aggregation

Creating Weak Classifiers with resampling:

- Suppose we have dependent variables \mathbf{Y} and data \mathbf{X}

Bagging: bootstrap aggregation

Creating Weak Classifiers with resampling:

- Suppose we have dependent variables \mathbf{Y} and data \mathbf{X}
- For each bootstrap step m , ($m = 1, 2, \dots, M$) draw N observations with replacement, $\tilde{\mathbf{Y}}_m$, $\tilde{\mathbf{X}}_m$.

Bagging: bootstrap aggregation

Creating Weak Classifiers with resampling:

- Suppose we have dependent variables \mathbf{Y} and data \mathbf{X}
- For each bootstrap step m , ($m = 1, 2, \dots, M$) draw N observations with replacement, $\tilde{\mathbf{Y}}_m$, $\tilde{\mathbf{X}}_m$.
- Train classifier on bootstrapped data,

Bagging: bootstrap aggregation

Creating Weak Classifiers with resampling:

- Suppose we have dependent variables \mathbf{Y} and data \mathbf{X}
- For each bootstrap step m , ($m = 1, 2, \dots, M$) draw N observations with replacement, $\tilde{\mathbf{Y}}_m$, $\tilde{\mathbf{X}}_m$.
- Train classifier on bootstrapped data,

$$\tilde{\mathbf{Y}}_m = f^m(\tilde{\mathbf{X}}_m, \hat{\boldsymbol{\beta}}, \lambda)$$

Bagging: bootstrap aggregation

Creating Weak Classifiers with resampling:

- Suppose we have dependent variables \mathbf{Y} and data \mathbf{X}
- For each bootstrap step m , ($m = 1, 2, \dots, M$) draw N observations with replacement, $\tilde{\mathbf{Y}}_m$, $\tilde{\mathbf{X}}_m$.
- Train classifier on bootstrapped data,

$$\begin{aligned}\tilde{\mathbf{Y}}_m &= f^m(\tilde{\mathbf{X}}_m, \hat{\boldsymbol{\beta}}, \lambda) \\ \hat{f}^m(\mathbf{x}_i, \hat{\boldsymbol{\beta}}, \lambda) &= \text{Classifier from } m^{\text{th}} \text{ iteration at } \mathbf{x}_i\end{aligned}$$

Bagging: bootstrap aggregation

Creating Weak Classifiers with resampling:

- Suppose we have dependent variables \mathbf{Y} and data \mathbf{X}
- For each bootstrap step m , ($m = 1, 2, \dots, M$) draw N observations with replacement, $\tilde{\mathbf{Y}}_m$, $\tilde{\mathbf{X}}_m$.
- Train classifier on bootstrapped data,

$$\begin{aligned}\tilde{\mathbf{Y}}_m &= f^m(\tilde{\mathbf{X}}_m, \hat{\boldsymbol{\beta}}, \lambda) \\ \hat{f}^m(\mathbf{x}_i, \hat{\boldsymbol{\beta}}, \lambda) &= \text{Classifier from } m^{\text{th}} \text{ iteration at } \mathbf{x}_i\end{aligned}$$

- Aggregating across classifiers,

Bagging: bootstrap aggregation

Creating Weak Classifiers with resampling:

- Suppose we have dependent variables \mathbf{Y} and data \mathbf{X}
- For each bootstrap step m , ($m = 1, 2, \dots, M$) draw N observations with replacement, $\tilde{\mathbf{Y}}_m$, $\tilde{\mathbf{X}}_m$.
- Train classifier on bootstrapped data,

$$\begin{aligned}\tilde{\mathbf{Y}}_m &= f^m(\tilde{\mathbf{X}}_m, \hat{\boldsymbol{\beta}}, \lambda) \\ \hat{f}^m(\mathbf{x}_i, \hat{\boldsymbol{\beta}}, \lambda) &= \text{Classifier from } m^{\text{th}} \text{ iteration at } \mathbf{x}_i\end{aligned}$$

- Aggregating across classifiers,

$$f_{\text{bag}}(\mathbf{x}_i) = \frac{1}{M} \sum_{m=1}^M \hat{f}^m(\mathbf{x}_i, \hat{\boldsymbol{\beta}}, \lambda)$$

Bagging: bootstrap aggregation

Creating Weak Classifiers with resampling:

- Suppose we have dependent variables \mathbf{Y} and data \mathbf{X}
- For each bootstrap step m , ($m = 1, 2, \dots, M$) draw N observations with replacement, $\tilde{\mathbf{Y}}_m$, $\tilde{\mathbf{X}}_m$.
- Train classifier on bootstrapped data,

$$\begin{aligned}\tilde{\mathbf{Y}}_m &= f^m(\tilde{\mathbf{X}}_m, \hat{\boldsymbol{\beta}}, \lambda) \\ \hat{f}^m(\mathbf{x}_i, \hat{\boldsymbol{\beta}}, \lambda) &= \text{Classifier from } m^{\text{th}} \text{ iteration at } \mathbf{x}_i\end{aligned}$$

- Aggregating across classifiers,

$$f_{\text{bag}}(\mathbf{x}_i) = \frac{1}{M} \sum_{m=1}^M \hat{f}^m(\mathbf{x}_i, \hat{\boldsymbol{\beta}}, \lambda)$$

- Only leads to a difference in estimate if classifiers are non-linear.

Bagging: bootstrap aggregation

Creating Weak Classifiers with resampling:

- Suppose we have dependent variables \mathbf{Y} and data \mathbf{X}
- For each bootstrap step m , ($m = 1, 2, \dots, M$) draw N observations with replacement, $\tilde{\mathbf{Y}}_m$, $\tilde{\mathbf{X}}_m$.
- Train classifier on bootstrapped data,

$$\begin{aligned}\tilde{\mathbf{Y}}_m &= f^m(\tilde{\mathbf{X}}_m, \hat{\boldsymbol{\beta}}, \lambda) \\ \hat{f}^m(\mathbf{x}_i, \hat{\boldsymbol{\beta}}, \lambda) &= \text{Classifier from } m^{\text{th}} \text{ iteration at } \mathbf{x}_i\end{aligned}$$

- Aggregating across classifiers,

$$f_{\text{bag}}(\mathbf{x}_i) = \frac{1}{M} \sum_{m=1}^M \hat{f}^m(\mathbf{x}_i, \hat{\boldsymbol{\beta}}, \lambda)$$

- Only leads to a difference in estimate if classifiers are non-linear.
- Strong Correlation between classifiers (recall optimal division from previous slide)

Classification and Regression Trees (CART): Intuition

Consider regression $E[Y|\mathbf{x}_i]$.

Classification and Regression Trees (CART): Intuition

Consider regression $E[Y|\mathbf{x}_i]$.

With no assumptions, **stratify** \rightsquigarrow different mean for unique values of \mathbf{x}_i

Classification and Regression Trees (CART): Intuition

Consider regression $E[Y|\mathbf{x}_i]$.

With no assumptions, **stratify** \rightsquigarrow different mean for unique values of \mathbf{x}_i

- Within each strata p , compute average Y

Classification and Regression Trees (CART): Intuition

Consider regression $E[Y|\mathbf{x}_i]$.

With no assumptions, **stratify** \rightsquigarrow different mean for unique values of \mathbf{x}_i

- Within each strata p , compute average Y

$$\bar{Y}|\mathbf{x}_p = \sum_{i=1}^N \frac{I(\mathbf{x}_i = \mathbf{x}_p) Y_i}{\sum_{t=1}^N I(\mathbf{x}_t = \mathbf{x}_p)}$$

Classification and Regression Trees (CART): Intuition

Consider regression $E[Y|\mathbf{x}_i]$.

With no assumptions, **stratify** \rightsquigarrow different mean for unique values of \mathbf{x}_i

- Within each strata p , compute average Y

$$\bar{Y}|\mathbf{x}_p = \sum_{i=1}^N \frac{I(\mathbf{x}_i = \mathbf{x}_p) Y_i}{\sum_{t=1}^N I(\mathbf{x}_t = \mathbf{x}_p)}$$

Implies that for test data we would fit:

Classification and Regression Trees (CART): Intuition

Consider regression $E[Y|\mathbf{x}_i]$.

With no assumptions, **stratify** \rightsquigarrow different mean for unique values of \mathbf{x}_i

- Within each strata p , compute average \bar{Y}

$$\bar{Y}|\mathbf{x}_p = \sum_{i=1}^N \frac{I(\mathbf{x}_i = \mathbf{x}_p) Y_i}{\sum_{t=1}^N I(\mathbf{x}_t = \mathbf{x}_p)}$$

Implies that for test data we would fit:

$$\hat{f}(\mathbf{x}_i) = \sum_{p=1}^P \bar{Y}|\mathbf{x}_p I(\mathbf{x}_i = \mathbf{x}_p)$$

Classification and Regression Trees (CART): Intuition

Consider regression $E[Y|\mathbf{x}_i]$.

With no assumptions, **stratify** \rightsquigarrow different mean for unique values of \mathbf{x}_i

- Within each strata p , compute average \bar{Y}

$$\bar{Y}|\mathbf{x}_p = \sum_{i=1}^N \frac{I(\mathbf{x}_i = \mathbf{x}_p) Y_i}{\sum_{t=1}^N I(\mathbf{x}_t = \mathbf{x}_p)}$$

Implies that for test data we would fit:

$$\begin{aligned}\hat{f}(\mathbf{x}_i) &= \sum_{p=1}^P \bar{Y}|\mathbf{x}_p I(\mathbf{x}_i = \mathbf{x}_p) \\ &= \sum_{p=1}^P c_p I(\mathbf{x}_i = \mathbf{x}_p)\end{aligned}$$

Classification and Regression Trees (CART): Intuition

Consider regression $E[Y|\mathbf{x}_i]$.

With no assumptions, **stratify** \rightsquigarrow different mean for unique values of \mathbf{x}_i

- Within each strata p , compute average \bar{Y}

$$\bar{Y}|\mathbf{x}_p = \sum_{i=1}^N \frac{I(\mathbf{x}_i = \mathbf{x}_p) Y_i}{\sum_{t=1}^N I(\mathbf{x}_t = \mathbf{x}_p)}$$

Implies that for test data we would fit:

$$\begin{aligned}\hat{f}(\mathbf{x}_i) &= \sum_{p=1}^P \bar{Y}|\mathbf{x}_p I(\mathbf{x}_i = \mathbf{x}_p) \\ &= \sum_{p=1}^P c_p I(\mathbf{x}_i = \mathbf{x}_p)\end{aligned}$$

Curse of dimensionality(!!!)

Classification and Regression Trees (CART): Intuition

Consider regression $E[Y|\mathbf{x}_i]$.

With no assumptions, **stratify** \rightsquigarrow different mean for unique values of \mathbf{x}_i

- Within each strata p , compute average \bar{Y}

$$\bar{Y}|\mathbf{x}_p = \sum_{i=1}^N \frac{I(\mathbf{x}_i = \mathbf{x}_p) Y_i}{\sum_{t=1}^N I(\mathbf{x}_t = \mathbf{x}_p)}$$

Implies that for test data we would fit:

$$\begin{aligned}\hat{f}(\mathbf{x}_i) &= \sum_{p=1}^P \bar{Y}|\mathbf{x}_p I(\mathbf{x}_i = \mathbf{x}_p) \\ &= \sum_{p=1}^P c_p I(\mathbf{x}_i = \mathbf{x}_p)\end{aligned}$$

Curse of dimensionality(!!!)

Approximate with **regions** \rightsquigarrow search for splits of data to approximate stratification

Classification and Regression Trees (CART): Objective function

Labels \mathbf{Y}_i and documents \mathbf{x}_i

$$\begin{aligned} E[Y|\mathbf{x}_i] &= \hat{f}(\mathbf{x}_i) \\ &= \sum_{p=1}^P c_p I(\mathbf{x}_i \in R_p) \end{aligned}$$

where:

- R_p describes a **region** \rightsquigarrow node
- c_p describes values of Y_i for document in R_p

Classification and Regression Trees (CART): Optimization function

Suppose we want to minimize sum of squared residuals with each **node**

Classification and Regression Trees (CART): Optimization function

Suppose we want to minimize sum of squared residuals with each **node**

Then c_p = Average Y for documents assigned to R_p

Classification and Regression Trees (CART): Optimization function

Suppose we want to minimize sum of squared residuals with each node

Then c_p = Average Y for documents assigned to R_p

$$\hat{c}_p = \frac{\sum_{i=1}^N Y_i I(\mathbf{x}_i \in R_p)}{\sum_{j=1}^N I(\mathbf{x}_j \in R_p)}$$

Classification and Regression Trees (CART): Optimization function

Suppose we want to minimize sum of squared residuals with each node

Then c_p = Average Y for documents assigned to R_p

$$\hat{c}_p = \sum_{i=1}^N \frac{Y_i I(\mathbf{x}_i \in R_p)}{\sum_{j=1}^N I(\mathbf{x}_j \in R_p)}$$

Determining an optimal partition \rightsquigarrow NP-Hard.

Classification and Regression Trees (CART): Optimization function

Suppose we want to minimize sum of squared residuals with each node

Then c_p = Average Y for documents assigned to R_p

$$\hat{c}_p = \sum_{i=1}^N \frac{Y_i I(\mathbf{x}_i \in R_p)}{\sum_{j=1}^N I(\mathbf{x}_j \in R_p)}$$

Determining an optimal partition \rightsquigarrow NP-Hard.

Suppose we are in some node (perhaps at the start).

Classification and Regression Trees (CART): Optimization function

Suppose we want to minimize sum of squared residuals with each node

Then c_p = Average Y for documents assigned to R_p

$$\hat{c}_p = \sum_{i=1}^N \frac{Y_i I(\mathbf{x}_i \in R_p)}{\sum_{j=1}^N I(\mathbf{x}_j \in R_p)}$$

Determining an optimal partition \rightsquigarrow NP-Hard.

Suppose we are in some node (perhaps at the start).

Greedy algorithm:

Classification and Regression Trees (CART): Optimization function

Suppose we want to minimize sum of squared residuals with each node

Then c_p = Average Y for documents assigned to R_p

$$\hat{c}_p = \sum_{i=1}^N \frac{Y_i I(x_i \in R_p)}{\sum_{j=1}^N I(x_j \in R_p)}$$

Determining an optimal partition \rightsquigarrow NP-Hard.

Suppose we are in some node (perhaps at the start).

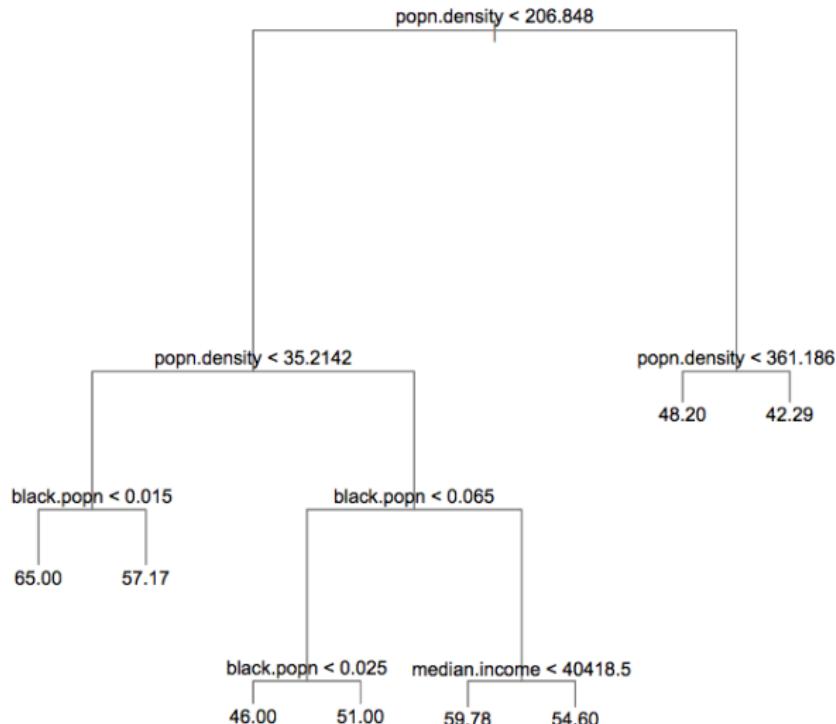
Greedy algorithm:

$$(j^*, s^*) = \arg \min_{j,s} \left[\underbrace{\min_{c_1} \sum_{i=1}^N I(x_{ij} < s) (Y_i - c_1)^2}_{\text{"cost" group 1}} + \underbrace{\min_{c_2} \sum_{i=1}^N I(x_{ij} > s) (Y_i - c_2)^2}_{\text{"cost" group 2}} \right]$$

Classification and Regression Trees (CART): Algorithm

- Start in Node
- Partition according to Greedy algorithm
- Continue until some stopping rule: number of observations per node

CART Picture (Spirling 2008)



Forests and Trees

Recall: accurate (unbiased) and uncorrelated classifiers

Forests and Trees

Recall: accurate (unbiased) and uncorrelated classifiers

- Grow trees deeply \rightsquigarrow unbiased classifiers, though high variance

Forests and Trees

Recall: accurate (unbiased) and uncorrelated classifiers

- Grow trees deeply \rightsquigarrow unbiased classifiers, though high variance
- **Average** \rightsquigarrow reduces variance, but will be correlated

Forests and Trees

Recall: accurate (unbiased) and uncorrelated classifiers

- Grow trees deeply \rightsquigarrow unbiased classifiers, though high variance
- **Average** \rightsquigarrow reduces variance, but will be correlated
- Random forest \rightsquigarrow introduce additional sampling to induce independence \rightsquigarrow Only split on subset of variables

Random Forest Algorithm (ESL, 588)

Random Forest Algorithm (ESL, 588)

- 1) For m bootstrap samples ($m = 1, \dots, M$), draw N observations with replacement, $\tilde{\mathbf{Y}}_m, \tilde{\mathbf{X}}_m$

Random Forest Algorithm (ESL, 588)

- 1) For m bootstrap samples ($m = 1, \dots, M$), draw N observations with replacement, $\tilde{\mathbf{Y}}_m, \tilde{\mathbf{X}}_m$
- 2) Until a minimum node size is reached:

Random Forest Algorithm (ESL, 588)

- 1) For m bootstrap samples ($m = 1, \dots, M$), draw N observations with replacement, $\tilde{\mathbf{Y}}_m, \tilde{\mathbf{X}}_m$
- 2) Until a minimum node size is reached:
 - i) Select z of the J variables \rightsquigarrow introduces independences across the trees

Random Forest Algorithm (ESL, 588)

- 1) For m bootstrap samples ($m = 1, \dots, M$), draw N observations with replacement, $\tilde{\mathbf{Y}}_m, \tilde{\mathbf{X}}_m$
- 2) Until a minimum node size is reached:
 - i) Select z of the J variables \rightsquigarrow introduces independences across the trees
 - ii) Among those z , select the best split node

Random Forest Algorithm (ESL, 588)

- 1) For m bootstrap samples ($m = 1, \dots, M$), draw N observations with replacement, $\tilde{\mathbf{Y}}_m, \tilde{\mathbf{X}}_m$
- 2) Until a minimum node size is reached:
 - i) Select z of the J variables \rightsquigarrow introduces independences across the trees
 - ii) Among those z , select the best split node
 - iii) Split into daughter nodes

Random Forest Algorithm (ESL, 588)

- 1) For m bootstrap samples ($m = 1, \dots, M$), draw N observations with replacement, $\tilde{\mathbf{Y}}_m, \tilde{\mathbf{X}}_m$
- 2) Until a minimum node size is reached:
 - i) Select z of the J variables \rightsquigarrow introduces independences across the trees
 - ii) Among those z , select the best split node
 - iii) Split into daughter nodes
- 3) The result is an ensemble (forest) of trees $\mathcal{T} = (\mathcal{T}_1, \mathcal{T}_2, \dots, \mathcal{T}_M)$,

Random Forest Algorithm (ESL, 588)

- 1) For m bootstrap samples ($m = 1, \dots, M$), draw N observations with replacement, $\tilde{\mathbf{Y}}_m, \tilde{\mathbf{X}}_m$
- 2) Until a minimum node size is reached:
 - i) Select z of the J variables \rightsquigarrow introduces independences across the trees
 - ii) Among those z , select the best split node
 - iii) Split into daughter nodes
- 3) The result is an ensemble (forest) of trees $\mathcal{T} = (T_1, T_2, \dots, T_M)$,

$$\hat{f}(\mathbf{x}_i) = \frac{1}{M} \sum_{m=1}^M T_m(\mathbf{x}_i)$$

Random Forest Algorithm (ESL, 588)

- 1) For m bootstrap samples ($m = 1, \dots, M$), draw N observations with replacement, $\tilde{\mathbf{Y}}_m, \tilde{\mathbf{X}}_m$
- 2) Until a minimum node size is reached:
 - i) Select z of the J variables \rightsquigarrow introduces independences across the trees
 - ii) Among those z , select the best split node
 - iii) Split into daughter nodes
- 3) The result is an ensemble (forest) of trees $\mathcal{T} = (T_1, T_2, \dots, T_M)$,

$$\hat{f}(\mathbf{x}_i) = \frac{1}{M} \sum_{m=1}^M T_m(\mathbf{x}_i)$$

RandomForest \rightsquigarrow Not a silver bullet!

Random Forest Algorithm (ESL, 588)

- 1) For m bootstrap samples ($m = 1, \dots, M$), draw N observations with replacement, $\tilde{\mathbf{Y}}_m, \tilde{\mathbf{X}}_m$
- 2) Until a minimum node size is reached:
 - i) Select z of the J variables \rightsquigarrow introduces independences across the trees
 - ii) Among those z , select the best split node
 - iii) Split into daughter nodes
- 3) The result is an ensemble (forest) of trees $\mathcal{T} = (T_1, T_2, \dots, T_M)$,

$$\hat{f}(\mathbf{x}_i) = \frac{1}{M} \sum_{m=1}^M T_m(\mathbf{x}_i)$$

RandomForest \rightsquigarrow Not a silver bullet!

- With many poor predictors \rightsquigarrow the p selected may be meaningless

Random Forest Algorithm (ESL, 588)

- 1) For m bootstrap samples ($m = 1, \dots, M$), draw N observations with replacement, $\tilde{\mathbf{Y}}_m, \tilde{\mathbf{X}}_m$
- 2) Until a minimum node size is reached:
 - i) Select z of the J variables \rightsquigarrow introduces independences across the trees
 - ii) Among those z , select the best split node
 - iii) Split into daughter nodes
- 3) The result is an ensemble (forest) of trees $\mathcal{T} = (T_1, T_2, \dots, T_M)$,

$$\hat{f}(\mathbf{x}_i) = \frac{1}{M} \sum_{m=1}^M T_m(\mathbf{x}_i)$$

RandomForest \rightsquigarrow Not a silver bullet!

- With many poor predictors \rightsquigarrow the p selected may be meaningless
- Wager and Athey (2015): Random Forest for estimating heterogeneous effects

Super Learning

- 1) Set of hand labeled documents. For each i , ($i = 1, \dots, N_{\text{train}}$)

Super Learning

- 1) Set of hand labeled documents. For each i , ($i = 1, \dots, N_{\text{train}}$)

$$Y_{i,\text{train}} \in \{C_1 C_2, \dots C_K\}$$

Super Learning

- 1) Set of hand labeled documents. For each i , ($i = 1, \dots, N_{\text{train}}$)
$$Y_{i,\text{train}} \in \{C_1 C_2, \dots C_K\}$$
- 2) Estimate relationship between labels and words

Super Learning

- 1) Set of hand labeled documents. For each i , ($i = 1, \dots, N_{\text{train}}$)
 $Y_{i,\text{train}} \in \{C_1 C_2, \dots C_K\}$
- 2) Estimate relationship between labels and words
 - Each document i is a **count vector** of K words

Super Learning

- 1) Set of hand labeled documents. For each i , ($i = 1, \dots, N_{\text{train}}$)
 $Y_{i,\text{train}} \in \{C_1 C_2, \dots C_K\}$
- 2) Estimate relationship between labels and words
 - Each document i is a **count vector** of K words $\mathbf{x}_{i,\text{train}} = (X_{i1}, X_{i2}, \dots, X_{iK})$

Super Learning

- 1) Set of hand labeled documents. For each i , ($i = 1, \dots, N_{\text{train}}$)
 $Y_{i,\text{train}} \in \{C_1, C_2, \dots, C_K\}$
- 2) Estimate relationship between labels and words
 - Each document i is a **count vector** of K words
 $\mathbf{x}_{i,\text{train}} = (X_{i1}, X_{i2}, \dots, X_{iK})$

$$\Pr(Y_i = C_k | \mathbf{x}_i)$$

Super Learning

- 1) Set of hand labeled documents. For each i , ($i = 1, \dots, N_{\text{train}}$)
 $Y_{i,\text{train}} \in \{C_1 C_2, \dots C_K\}$
- 2) Estimate relationship between labels and words
 - Each document i is a **count vector** of K words
 $\mathbf{x}_{i,\text{train}} = (X_{i1}, X_{i2}, \dots, X_{iK})$

$$\Pr(Y_i = C_k | \mathbf{x}_i)_{\text{train}}$$

Super Learning

- 1) Set of hand labeled documents. For each i , ($i = 1, \dots, N_{\text{train}}$)
 $Y_{i,\text{train}} \in \{C_1, C_2, \dots, C_K\}$
- 2) Estimate relationship between labels and words
 - Each document i is a **count vector** of K words
 $\mathbf{x}_{i,\text{train}} = (X_{i1}, X_{i2}, \dots, X_{iK})$

$$\Pr(Y_i = C_k | \mathbf{x}_i)_{\text{train}} = \hat{g}(\mathbf{x}_i)_{\text{train}}$$

- Identify systematic relationship between words, labels

Super Learning

- 1) Set of hand labeled documents. For each i , ($i = 1, \dots, N_{\text{train}}$)
 $Y_{i,\text{train}} \in \{C_1 C_2, \dots C_K\}$
- 2) Estimate relationship between labels and words
 - Each document i is a **count vector** of K words
 $\mathbf{x}_{i,\text{train}} = (X_{i1}, X_{i2}, \dots, X_{iK})$

$$\Pr(Y_i = C_k | \mathbf{x}_i)_{\text{train}} = \hat{g}(\mathbf{x}_i)_{\text{train}}$$

- Identify systematic relationship between words, labels \rightsquigarrow Data and **assumptions**

Super Learning

- 1) Set of hand labeled documents. For each i , ($i = 1, \dots, N_{\text{train}}$)
 $Y_{i,\text{train}} \in \{C_1 C_2, \dots C_K\}$
- 2) Estimate relationship between labels and words
 - Each document i is a **count vector** of K words
 $\mathbf{x}_{i,\text{train}} = (X_{i1}, X_{i2}, \dots, X_{iK})$

$$\Pr(Y_i = C_k | \mathbf{x}_i)_{\text{train}} = \hat{g}(\mathbf{x}_i)_{\text{train}}$$

- Identify systematic relationship between words, labels \rightsquigarrow Data and **assumptions**
 - LASSO (Tibshirani 1996): **sparsity**

Super Learning

- 1) Set of hand labeled documents. For each i , ($i = 1, \dots, N_{\text{train}}$)
 $Y_{i,\text{train}} \in \{C_1 C_2, \dots C_K\}$
- 2) Estimate relationship between labels and words
 - Each document i is a **count vector** of K words
 - $\mathbf{x}_{i,\text{train}} = (X_{i1}, X_{i2}, \dots, X_{iK})$

$$\Pr(Y_i = C_k | \mathbf{x}_i)_{\text{train}} = \hat{g}(\mathbf{x}_i)_{\text{train}}$$

- Identify systematic relationship between words, labels \rightsquigarrow Data and **assumptions**
 - LASSO (Tibshirani 1996): **sparsity**
 - KRLS (Hainmueller and Hazlett 2013): **dense**, flexible surface

Super Learning

- 1) Set of hand labeled documents. For each i , ($i = 1, \dots, N_{\text{train}}$)
 $Y_{i,\text{train}} \in \{C_1 C_2, \dots C_K\}$
- 2) Estimate relationship between labels and words
 - Each document i is a **count vector** of K words
 - $\mathbf{x}_{i,\text{train}} = (X_{i1}, X_{i2}, \dots, X_{iK})$

$$\Pr(Y_i = C_k | \mathbf{x}_i)_{\text{train}} = \hat{g}(\mathbf{x}_i)_{\text{train}}$$

- Identify systematic relationship between words, labels \rightsquigarrow Data and **assumptions**
 - LASSO (Tibshirani 1996): **sparsity**
 - KRLS (Hainmueller and Hazlett 2013): **dense**, flexible surface
 - Ridge, Elastic-Net, SVM, Random Forests, BART, ...

Super Learning

- 1) Set of hand labeled documents. For each i , ($i = 1, \dots, N_{\text{train}}$)
 $Y_{i,\text{train}} \in \{C_1 C_2, \dots C_K\}$
- 2) Estimate relationship between labels and words
 - Each document i is a **count vector** of K words
 $\mathbf{x}_{i,\text{train}} = (X_{i1}, X_{i2}, \dots, X_{iK})$

$$\Pr(Y_i = C_k | \mathbf{x}_i)_{\text{train}} = \hat{g}(\mathbf{x}_i)_{\text{train}}$$

- Identify systematic relationship between words, labels \rightsquigarrow Data and **assumptions**
 - LASSO (Tibshirani 1996): **sparsity**
 - KRLS (Hainmueller and Hazlett 2013): **dense**, flexible surface
 - Ridge, Elastic-Net, SVM, Random Forests, BART, ...
- Which model? Difficult to know before hand

Super Learning

- 1) Set of hand labeled documents. For each i , ($i = 1, \dots, N_{\text{train}}$)
 $Y_{i,\text{train}} \in \{C_1 C_2, \dots C_K\}$
- 2) Estimate relationship between labels and words
 - Each document i is a **count vector** of K words
 $\mathbf{x}_{i,\text{train}} = (X_{i1}, X_{i2}, \dots, X_{iK})$

$$\Pr(Y_i = C_k | \mathbf{x}_i)_{\text{train}} = \hat{g}(\mathbf{x}_i)_{\text{train}}$$

- Identify systematic relationship between words, labels \rightsquigarrow Data and **assumptions**
 - LASSO (Tibshirani 1996): **sparsity**
 - KRLS (Hainmueller and Hazlett 2013): **dense**, flexible surface
 - Ridge, Elastic-Net, SVM, Random Forests, BART, ...
- Which model? Difficult to know before hand
- Assess out of sample performance with **cross validation**

Super Learning

- 1) Set of hand labeled documents. For each i , ($i = 1, \dots, N_{\text{train}}$)
 $Y_{i,\text{train}} \in \{C_1 C_2, \dots C_K\}$
- 2) Estimate relationship between labels and words
 - Each document i is a **count vector** of K words
 $\mathbf{x}_{i,\text{train}} = (X_{i1}, X_{i2}, \dots, X_{iK})$

$$\Pr(Y_i = C_k | \mathbf{x}_i)_{\text{train}} = \hat{g}(\mathbf{x}_i)_{\text{train}}$$

- Identify systematic relationship between words, labels \rightsquigarrow Data and **assumptions**
 - LASSO (Tibshirani 1996): **sparsity**
 - KRLS (Hainmueller and Hazlett 2013): **dense**, flexible surface
 - Ridge, Elastic-Net, SVM, Random Forests, BART, ...
- Which model? Difficult to know before hand
- Assess out of sample performance with **cross validation**

Weighted ensemble: weights determined by (unique) out of sample predictive performance

Committee Methods:

Fit many methods, average with equal weights

- Voting (classification)
- Averaging (predictions)

Problem: many poor methods may overwhelm high quality fit (remember earlier figures)

Solution: learn weights via cross validation

Weighted Ensemble to Classify Documents

- Suppose we have M ($m = 1, \dots, M$) models.

Weighted Ensemble to Classify Documents

- Suppose we have M ($m = 1, \dots, M$) models.

$$\Pr(Y_i = C_1 | \mathbf{x}_i)_{\text{train}} = \sum_{m=1}^M \hat{\pi}_m \hat{g}_m(\mathbf{x}_i)$$

Weighted Ensemble to Classify Documents

- Suppose we have M ($m = 1, \dots, M$) models.

$$\Pr(Y_i = C_1 | \mathbf{x}_i)_{\text{train}} = \sum_{m=1}^M \hat{\pi}_m \hat{g}_m(\mathbf{x}_i)$$

Weighted Ensemble to Classify Documents

- Suppose we have M ($m = 1, \dots, M$) models.

$$\Pr(Y_i = C_1 | \mathbf{x}_i)_{\text{train}} = \sum_{m=1}^M \hat{\pi}_m \hat{g}_m(\mathbf{x}_i)$$

Weighted Ensemble to Classify Documents

- Suppose we have M ($m = 1, \dots, M$) models.

$$\Pr(Y_i = C_1 | \mathbf{x}_i)_{\text{train}} = \sum_{m=1}^M \hat{\pi}_m \hat{g}_m(\mathbf{x}_i)$$

- Estimate weights ($\hat{\pi}_m$)

Weighted Ensemble to Classify Documents

- Suppose we have M ($m = 1, \dots, M$) models.

$$\Pr(Y_i = C_1 | \mathbf{x}_i)_{\text{train}} = \sum_{m=1}^M \hat{\pi}_m \hat{g}_m(\mathbf{x}_i)$$

- Estimate weights ($\hat{\pi}_m$)
 - K-fold cross validation: generate M out of sample predictions for each document in training set

Weighted Ensemble to Classify Documents

- Suppose we have M ($m = 1, \dots, M$) models.

$$\Pr(Y_i = C_1 | \mathbf{x}_i)_{\text{train}} = \sum_{m=1}^M \hat{\pi}_m \hat{g}_m(\mathbf{x}_i)$$

- Estimate weights ($\hat{\pi}_m$)

- K-fold cross validation: generate M out of sample predictions for each document in training set

$$\hat{\mathbf{Y}}_i = (\hat{Y}_{i1}, \hat{Y}_{i2}, \dots, \hat{Y}_{iM})$$

Weighted Ensemble to Classify Documents

- Suppose we have M ($m = 1, \dots, M$) models.

$$\Pr(Y_i = C_1 | \mathbf{x}_i)_{\text{train}} = \sum_{m=1}^M \hat{\pi}_m \hat{g}_m(\mathbf{x}_i)$$

- Estimate weights ($\hat{\pi}_m$)

- K-fold cross validation: generate M out of sample predictions for each document in training set

$$\hat{\mathbf{Y}}_i = (\hat{Y}_{i1}, \hat{Y}_{i2}, \dots, \hat{Y}_{iM})$$

- Estimate weights with constrained regression:

Weighted Ensemble to Classify Documents

- Suppose we have M ($m = 1, \dots, M$) models.

$$\Pr(Y_i = C_1 | \mathbf{x}_i)_{\text{train}} = \sum_{m=1}^M \hat{\pi}_m \hat{g}_m(\mathbf{x}_i)$$

- Estimate weights ($\hat{\pi}_m$)

- K-fold cross validation: generate M out of sample predictions for each document in training set

$$\hat{\mathbf{Y}}_i = (\hat{Y}_{i1}, \hat{Y}_{i2}, \dots, \hat{Y}_{iM})$$

- Estimate weights with constrained regression:

$$Y_i = \sum_{m=1}^M \pi_m \hat{Y}_{im} + \epsilon_i$$

Weighted Ensemble to Classify Documents

- Suppose we have M ($m = 1, \dots, M$) models.

$$\Pr(Y_i = C_1 | \mathbf{x}_i)_{\text{train}} = \sum_{m=1}^M \hat{\pi}_m \hat{g}_m(\mathbf{x}_i)$$

- Estimate weights ($\hat{\pi}_m$)

- K-fold cross validation: generate M out of sample predictions for each document in training set

$$\hat{\mathbf{Y}}_i = (\hat{Y}_{i1}, \hat{Y}_{i2}, \dots, \hat{Y}_{iM})$$

- Estimate weights with constrained regression:

$$Y_i = \sum_{m=1}^M \pi_m \hat{Y}_{im} + \epsilon_i$$

where we impose constraints: $\pi_m \geq 0$ and $\sum_{m=1}^M \pi_m = 1$.

Weighted Ensemble to Classify Documents

- Suppose we have M ($m = 1, \dots, M$) models.

$$\Pr(Y_i = C_1 | \mathbf{x}_i)_{\text{train}} = \sum_{m=1}^M \hat{\pi}_m \hat{g}_m(\mathbf{x}_i)$$

- Estimate weights ($\hat{\pi}_m$)

- K-fold cross validation: generate M out of sample predictions for each document in training set

$$\hat{\mathbf{Y}}_i = (\hat{Y}_{i1}, \hat{Y}_{i2}, \dots, \hat{Y}_{iM})$$

- Estimate weights with constrained regression:

$$Y_i = \sum_{m=1}^M \pi_m \hat{Y}_{im} + \epsilon_i$$

where we impose constraints: $\pi_m \geq 0$ and $\sum_{m=1}^M \pi_m = 1$.

- Result $\hat{\pi}_m$ for each method

Weighted Ensemble to Classify Documents

- Suppose we have M ($m = 1, \dots, M$) models.

$$\Pr(Y_i = C_1 | \mathbf{x}_i)_{\text{train}} = \sum_{m=1}^M \hat{\pi}_m \hat{g}_m(\mathbf{x}_i)$$

- Estimate weights ($\hat{\pi}_m$)
- Estimate $\hat{g}_m(\mathbf{x}_i) \rightsquigarrow$ Apply all M models to entire training set

Weighted Ensemble to Classify Documents

- Suppose we have M ($m = 1, \dots, M$) models.

$$\Pr(Y_i = C_1 | \mathbf{x}_i)_{\text{train}} = \sum_{m=1}^M \hat{\pi}_m \hat{g}_m(\mathbf{x}_i)$$

- Estimate weights ($\hat{\pi}_m$)
 - Estimate $\hat{g}_m(\mathbf{x}_i) \rightsquigarrow$ Apply all M models to entire training set
- 3) For each document i in test set, $\mathbf{x}_{i,\text{test}}$

Weighted Ensemble to Classify Documents

- Suppose we have M ($m = 1, \dots, M$) models.

$$\Pr(Y_i = C_1 | \mathbf{x}_{i,\text{test}})_{\text{train}} = \sum_{m=1}^M \hat{\pi}_m \hat{g}_m(\mathbf{x}_{i,\text{test}})$$

- Estimate weights ($\hat{\pi}_m$)
 - Estimate $\hat{g}_m(\mathbf{x}_i) \rightsquigarrow$ Apply all M models to entire training set
- 3) For each document i in test set, $\mathbf{x}_{i,\text{test}}$

Weighted Ensemble to Classify Documents

- Suppose we have M ($m = 1, \dots, M$) models.

$$\Pr(Y_i = C_1 | \mathbf{x}_{i,\text{test}})_{\text{train}} = \sum_{m=1}^M \hat{\pi}_m \hat{g}_m(\mathbf{x}_{i,\text{test}})$$

- Estimate weights ($\hat{\pi}_m$)
 - Estimate $\hat{g}_m(\mathbf{x}_i) \rightsquigarrow$ Apply all M models to entire training set
- 3) For each document i in test set, $\mathbf{x}_{i,\text{test}}$
(Classify if above threshold)

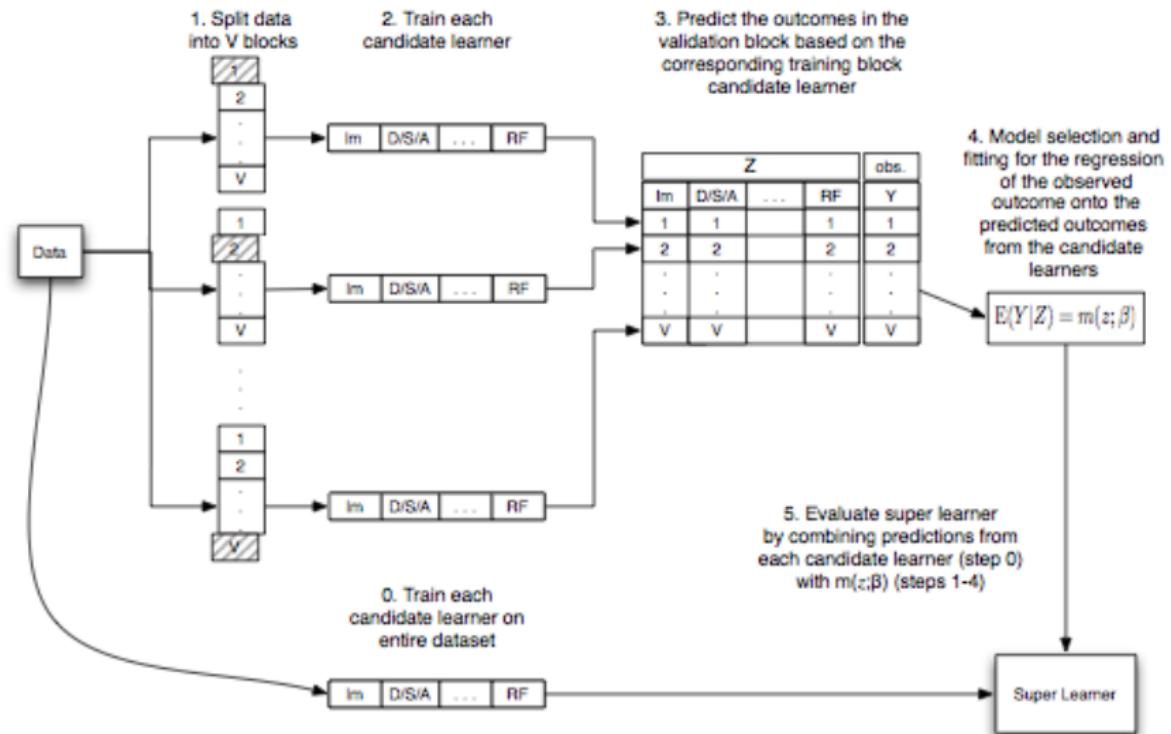


Figure 1: Flow Diagram for Super Learner

Why Super Learn?

van der Laan et al (2007) prove:

- **Asymptotically**: super learners will perform as well the **best** candidates for data
- **Oracle**: performs like the best possible method among candidate methods
 - Asymptotically outperforms constituent methods
 - Performs as well as optimal combinations of those methods

Practical questions:

- Final regression:
 - Logistic
 - Linear
 - Could super learn again!
- How Many Folds?
 - van der Laan et al's proofs rely on growing folds with N (but slowly)
 - Use 10-fold cross validation for simulations

Impression of Influence

Estimate: $Y_i \in \{\text{Credit}, \text{Not Credit}\}$

Impression of Influence

Estimate: $Y_i \in \{\text{Credit, Not Credit}\}$

- Triple hand code 800 press releases

Impression of Influence

Estimate: $Y_i \in \{\text{Credit}, \text{Not Credit}\}$

- Triple hand code 800 press releases
- Resolve disagreement with voting \rightsquigarrow few disagreements

Impression of Influence

Estimate: $Y_i \in \{\text{Credit}, \text{Not Credit}\}$

- Triple hand code 800 press releases
- Resolve disagreement with voting \rightsquigarrow few disagreements

Use five classifiers to form Ensemble (cross validating within each to tune parameters)

Impression of Influence

Estimate: $Y_i \in \{\text{Credit}, \text{Not Credit}\}$

- Triple hand code 800 press releases
- Resolve disagreement with voting \rightsquigarrow few disagreements

Use five classifiers to form Ensemble (cross validating within each to tune parameters)

- LASSO

Impression of Influence

Estimate: $Y_i \in \{\text{Credit}, \text{Not Credit}\}$

- Triple hand code 800 press releases
- Resolve disagreement with voting \rightsquigarrow few disagreements

Use five classifiers to form Ensemble (cross validating within each to tune parameters)

- LASSO
- Elastic-Net

Impression of Influence

Estimate: $Y_i \in \{\text{Credit}, \text{Not Credit}\}$

- Triple hand code 800 press releases
- Resolve disagreement with voting \rightsquigarrow few disagreements

Use five classifiers to form Ensemble (cross validating within each to tune parameters)

- LASSO
- Elastic-Net
- Random Forest

Impression of Influence

Estimate: $Y_i \in \{\text{Credit}, \text{Not Credit}\}$

- Triple hand code 800 press releases
- Resolve disagreement with voting \rightsquigarrow few disagreements

Use five classifiers to form Ensemble (cross validating within each to tune parameters)

- LASSO
- Elastic-Net
- Random Forest
- A Support Vector Machine

Impression of Influence

Estimate: $Y_i \in \{\text{Credit}, \text{Not Credit}\}$

- Triple hand code 800 press releases
- Resolve disagreement with voting \rightsquigarrow few disagreements

Use five classifiers to form Ensemble (cross validating within each to tune parameters)

- LASSO
- Elastic-Net
- Random Forest
- A Support Vector Machine
- Kernel Regularized Least Squares (KRLS, Hainmueller and Hazlett 2014)

Impression of Influence

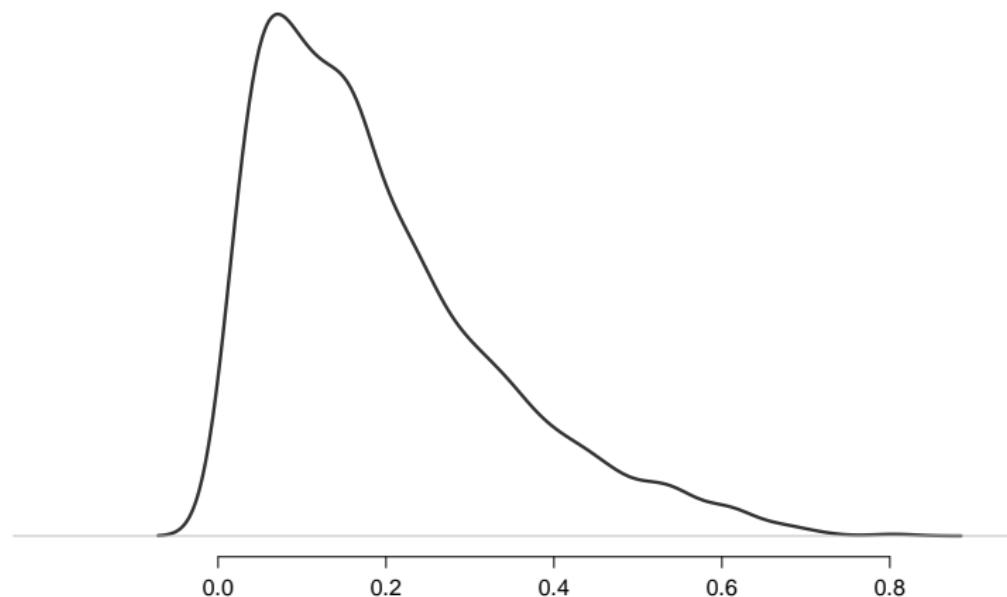
Estimate: $Y_i \in \{\text{Credit}, \text{Not Credit}\}$

- Triple hand code 800 press releases
- Resolve disagreement with voting \rightsquigarrow few disagreements

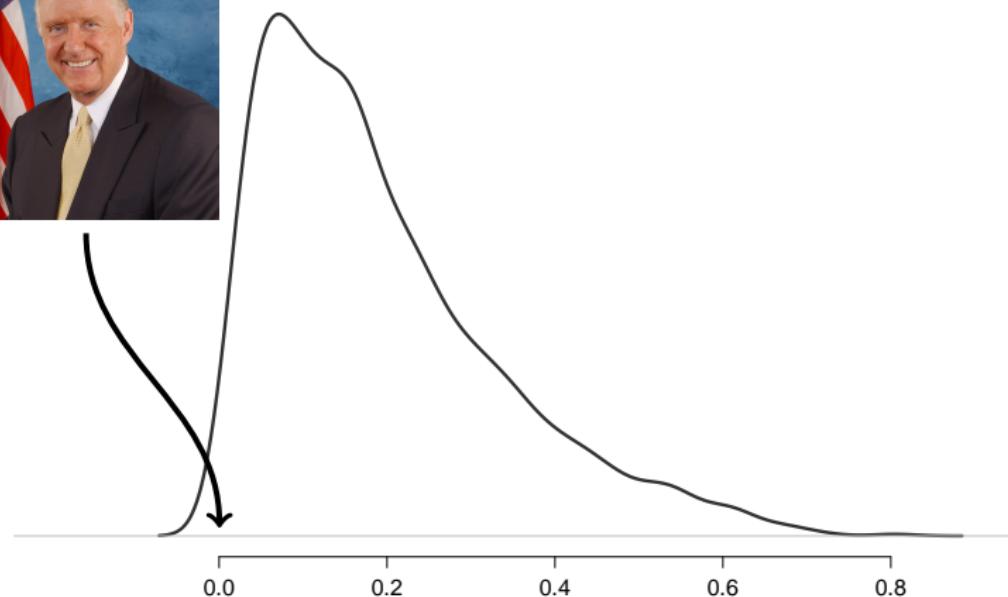
Use five classifiers to form Ensemble (cross validating within each to tune parameters)

- LASSO 0
- Elastic-Net 23%
- Random Forest 61%
- A Support Vector Machine 16%
- Kernel Regularized Least Squares (KRLS, Hainmueller and Hazlett 2014) 0

Strategic Credit Claiming to Build a Personal Vote



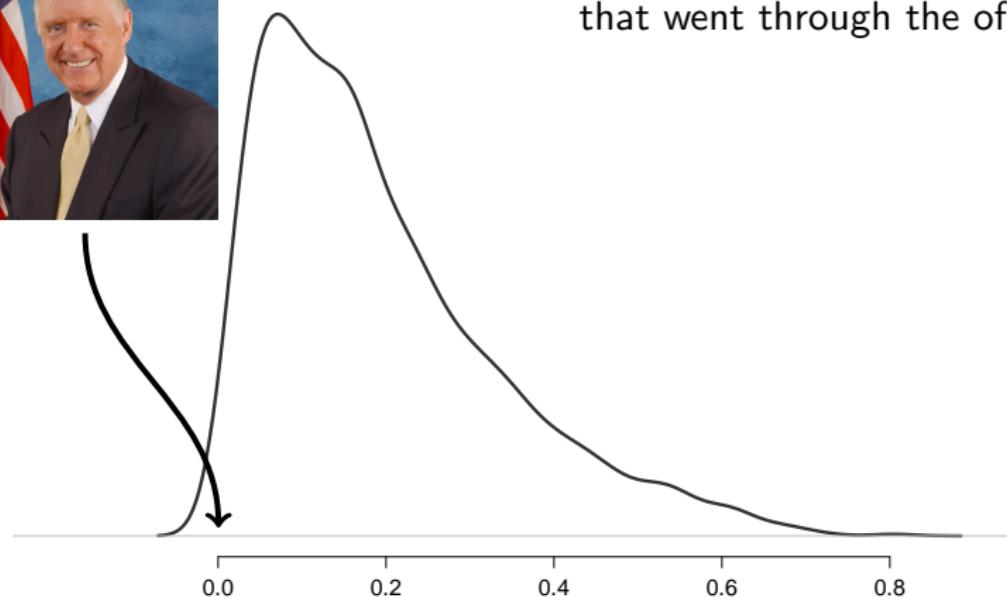
Strategic Credit Claiming to Build a Personal Vote



Strategic Credit Claiming to Build a Personal Vote



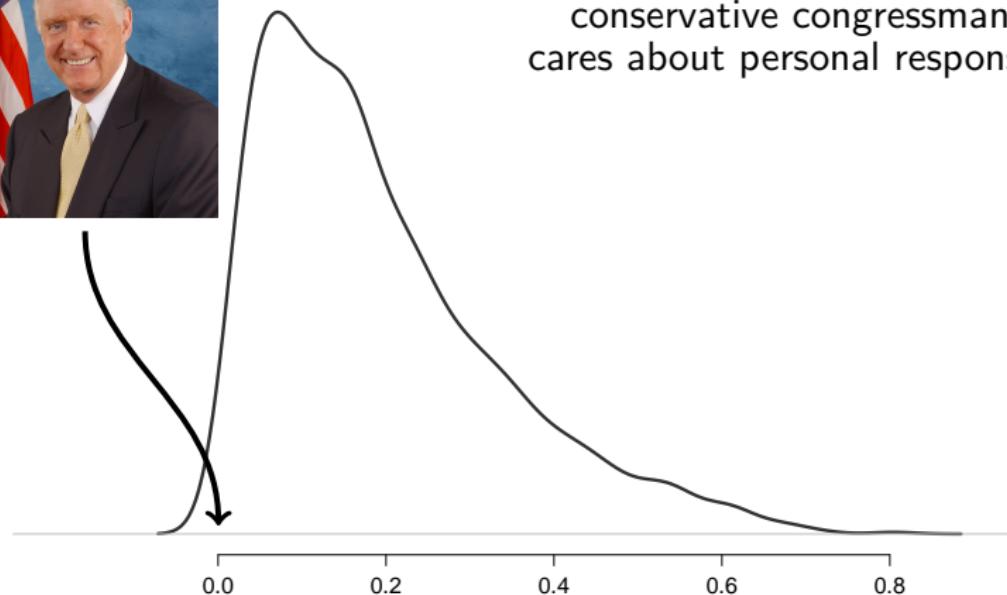
John McGroff: "voted for every spending bill that went through the office"



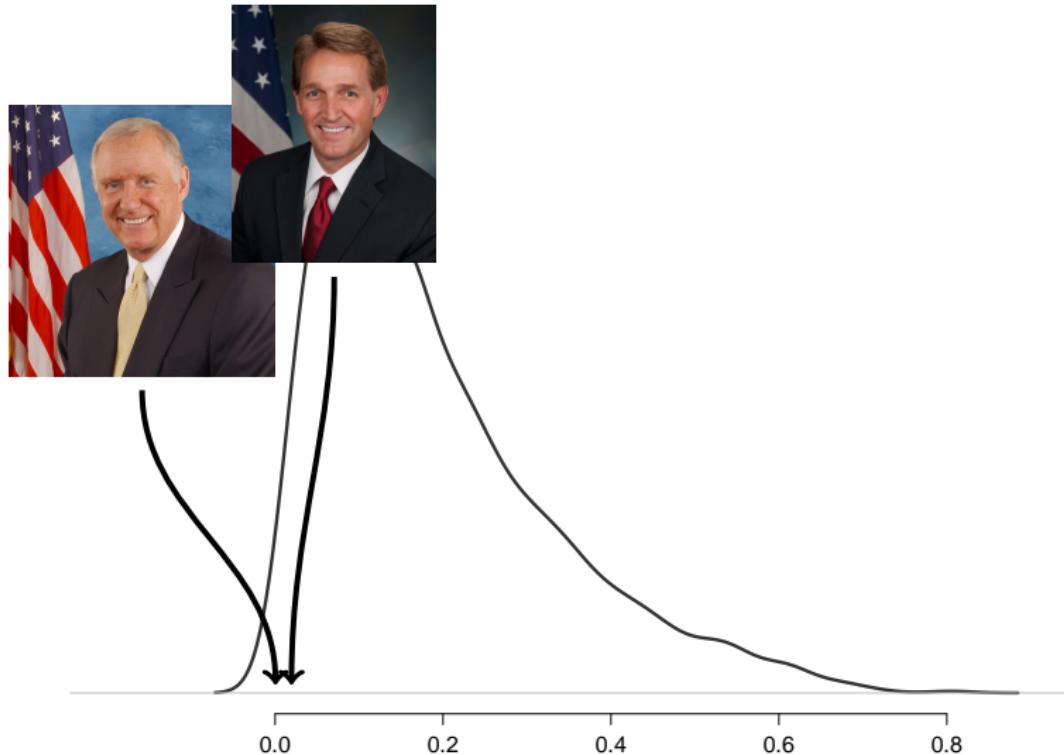
Strategic Credit Claiming to Build a Personal Vote



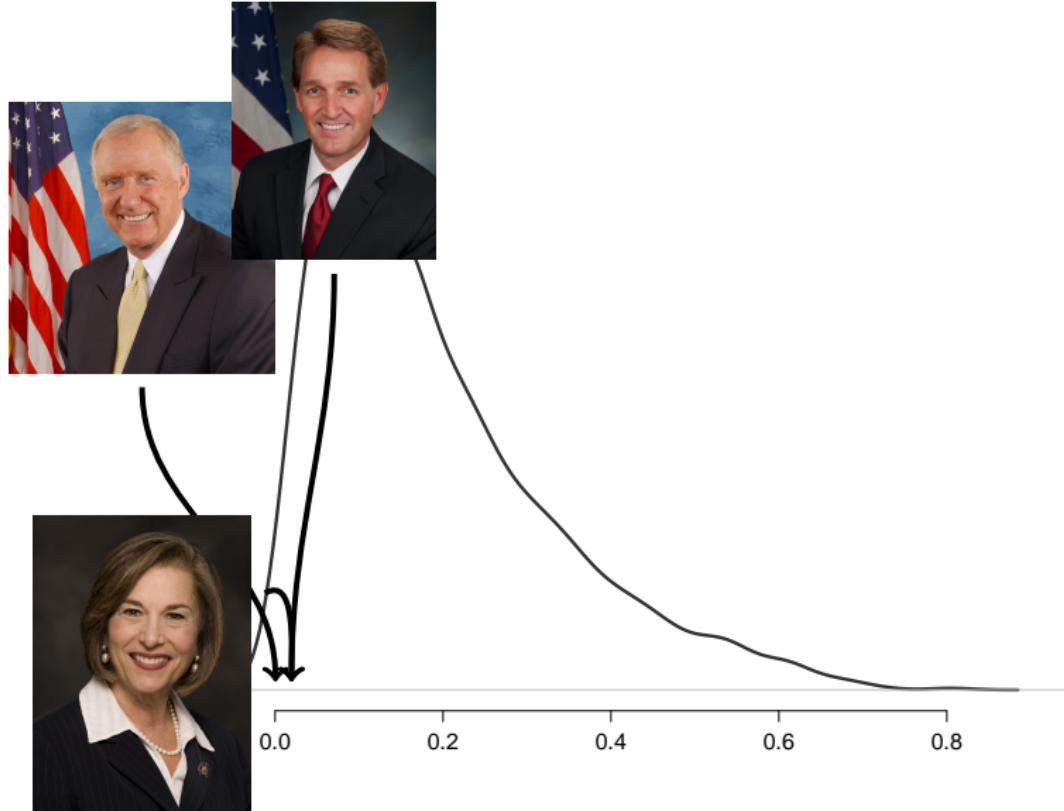
John McGroff: "Not the actions of a fiscally conservative congressman who cares about personal responsibility"



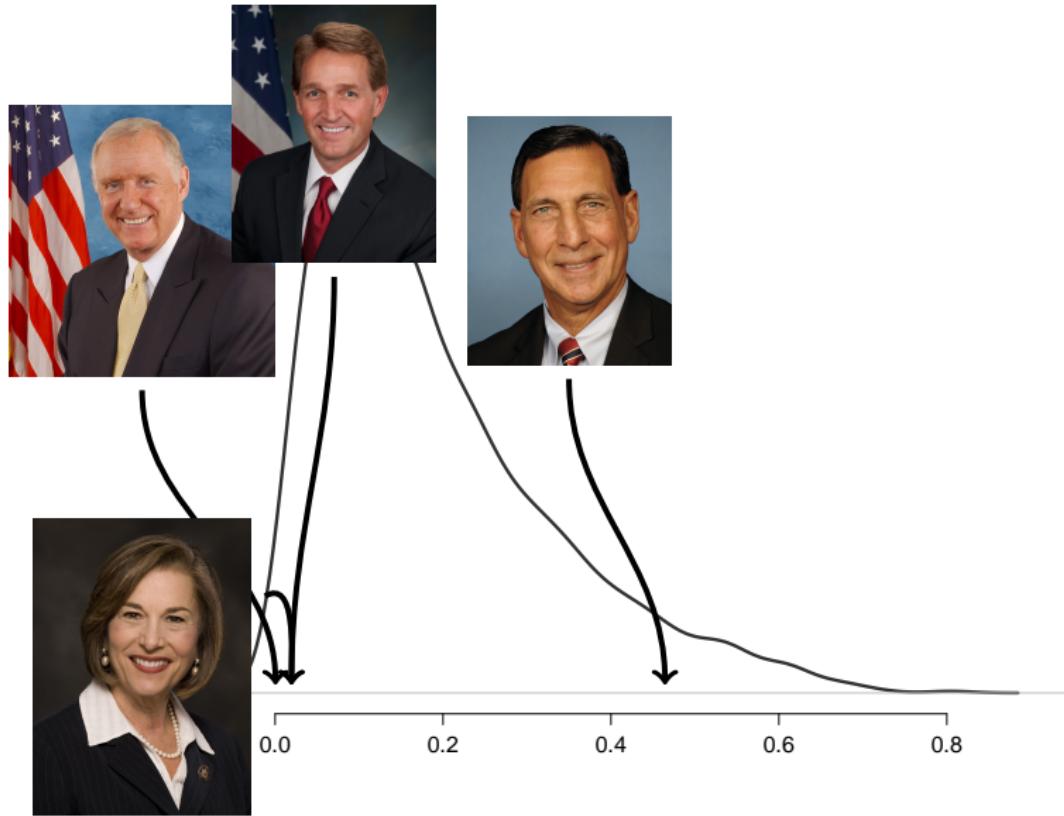
Strategic Credit Claiming to Build a Personal Vote



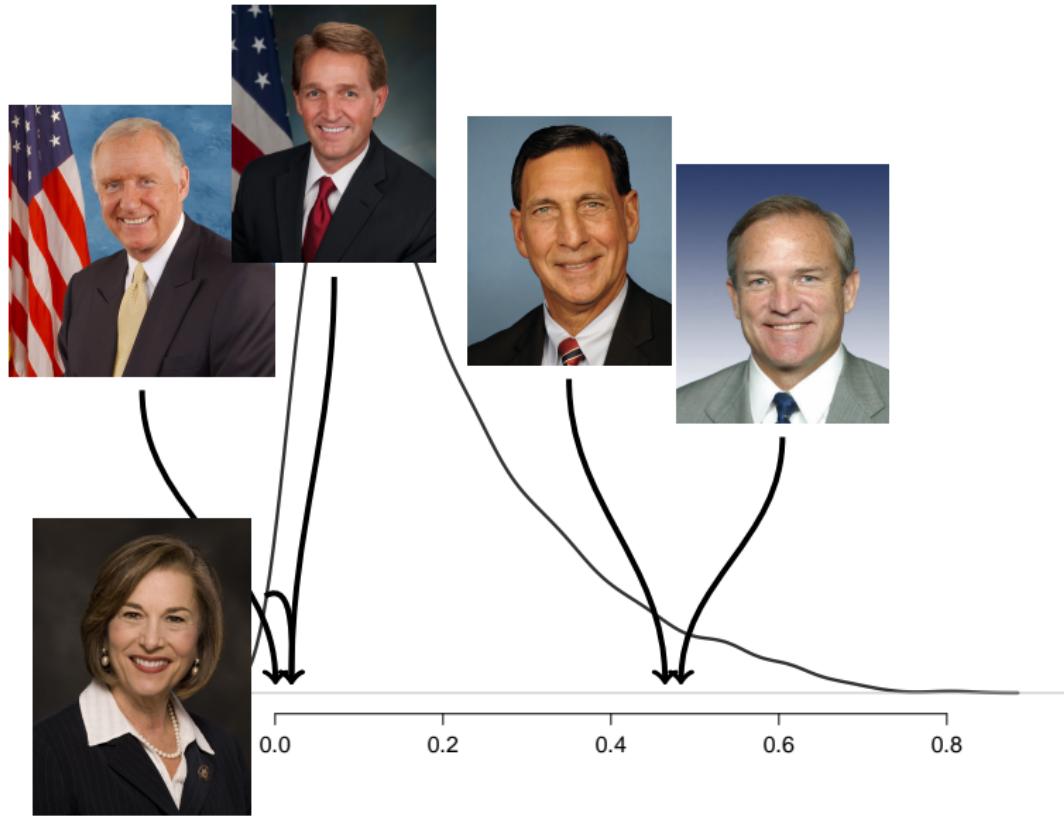
Strategic Credit Claiming to Build a Personal Vote



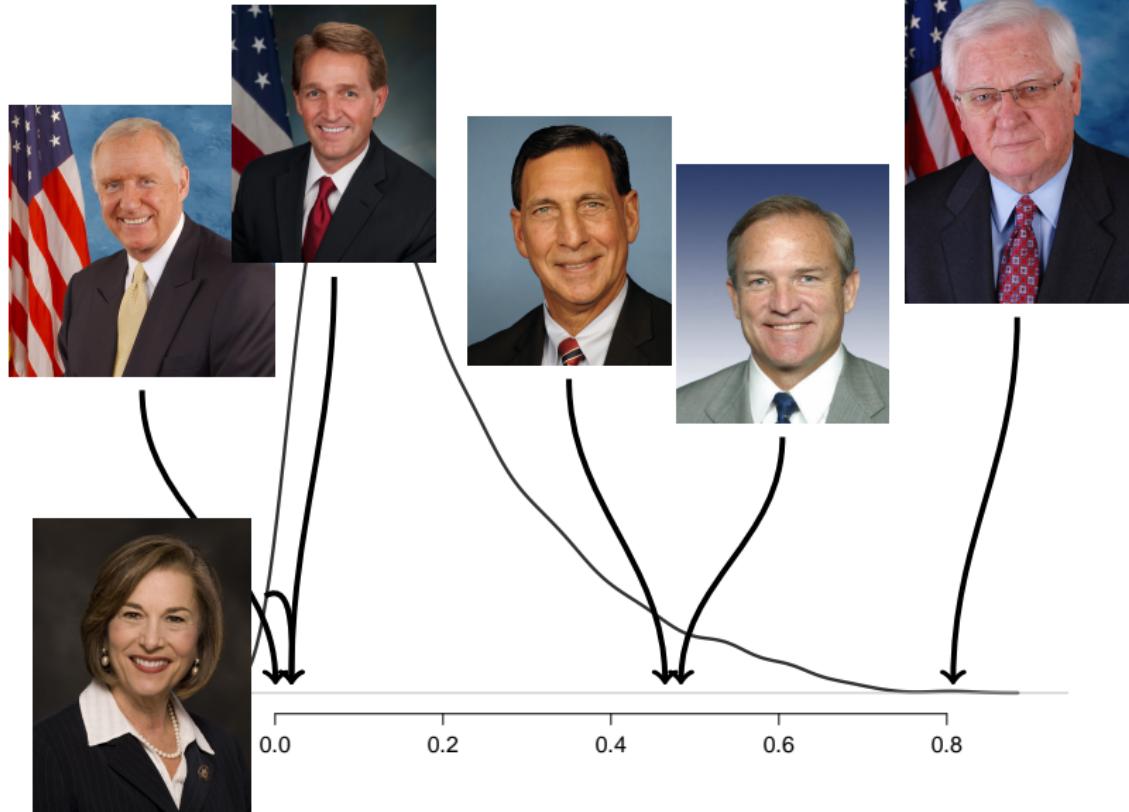
Strategic Credit Claiming to Build a Personal Vote



Strategic Credit Claiming to Build a Personal Vote

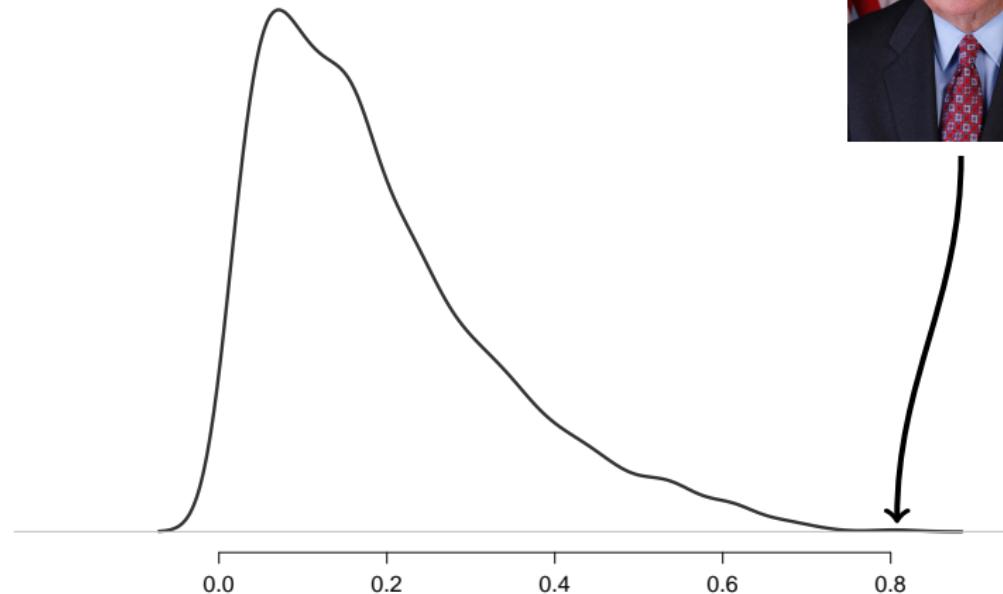


Strategic Credit Claiming to Build a Personal Vote



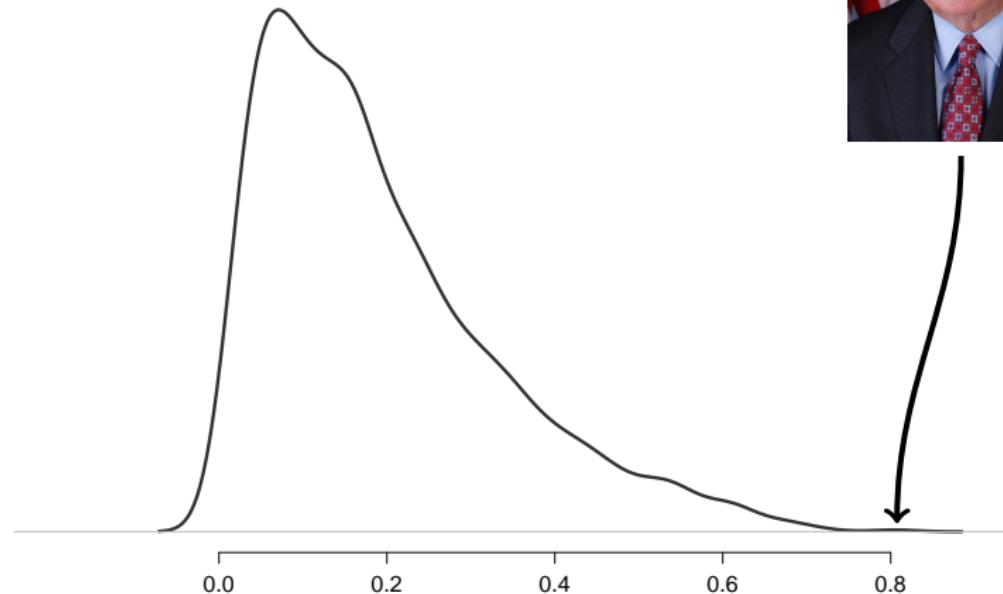
Strategic Credit Claiming to Build a Personal Vote

"We just can't afford luxuries like ideology"



Strategic Credit Claiming to Build a Personal Vote

Lexington Herald-Leader: Prince of Pork



Other Reasons to Ensemble (Dietterich 2000)

Statistical

Other Reasons to Ensemble (Dietterich 2000)

Statistical

- With little data, many algorithms offer similar performance

Other Reasons to Ensemble (Dietterich 2000)

Statistical

- With little data, many algorithms offer similar performance
- Ensemble ensures we avoid **wrong** model in test set

Other Reasons to Ensemble (Dietterich 2000)

Statistical

- With little data, many algorithms offer similar performance
- Ensemble ensures we avoid **wrong** model in test set

Computational

Other Reasons to Ensemble (Dietterich 2000)

Statistical

- With little data, many algorithms offer similar performance
- Ensemble ensures we avoid **wrong** model in test set

Computational

- Methods stuck in local modes

Other Reasons to Ensemble (Dietterich 2000)

Statistical

- With little data, many algorithms offer similar performance
- Ensemble ensures we avoid **wrong** model in test set

Computational

- Methods stuck in local modes
- Result: no one run provides best model

Other Reasons to Ensemble (Dietterich 2000)

Statistical

- With little data, many algorithms offer similar performance
- Ensemble ensures we avoid **wrong** model in test set

Computational

- Methods stuck in local modes
- Result: no one run provides best model
- Averages of runs may perform better

Other Reasons to Ensemble (Dietterich 2000)

Statistical

- With little data, many algorithms offer similar performance
- Ensemble ensures we avoid **wrong** model in test set

Computational

- Methods stuck in local modes
- Result: no one run provides best model
- Averages of runs may perform better

Complex “true” functional forms

Other Reasons to Ensemble (Dietterich 2000)

Statistical

- With little data, many algorithms offer similar performance
- Ensemble ensures we avoid **wrong** model in test set

Computational

- Methods stuck in local modes
- Result: no one run provides best model
- Averages of runs may perform better

Complex “true” functional forms

- One method may be unable to approximate true DGP

Other Reasons to Ensemble (Dietterich 2000)

Statistical

- With little data, many algorithms offer similar performance
- Ensemble ensures we avoid **wrong** model in test set

Computational

- Methods stuck in local modes
- Result: no one run provides best model
- Averages of runs may perform better

Complex “true” functional forms

- One method may be unable to approximate true DGP
- Mixtures of methods may approximate better

Other Reasons to Ensemble (Dietterich 2000)

Statistical

- With little data, many algorithms offer similar performance
- Ensemble ensures we avoid **wrong** model in test set

Computational

- Methods stuck in local modes
- Result: no one run provides best model
- Averages of runs may perform better

Complex “true” functional forms

- One method may be unable to approximate true DGP
- Mixtures of methods may approximate better