Text Data Mining

Jaime Arguello
jarguell@email.unc.edu

Outline: Predictive and Exploratory Analysis

Concepts, Instances, and Features

Human Annotation

Text Representation

Learning Algorithms

Evaluation metrics

Experimentation

Clustering

Hands-on Exercise

Outline: Predictive and Exploratory Analysis

Concepts, Instances, and Features

Human Annotation

Text Representation

Learning Algorithms

Evaluation metrics

Experimentation

Clustering

Hands-on Exercise

What is Text Data Mining?

- The science and practice of developing and evaluating computer programs that automatically detect or discover interesting and useful things in collections of natural language text

Related Fields

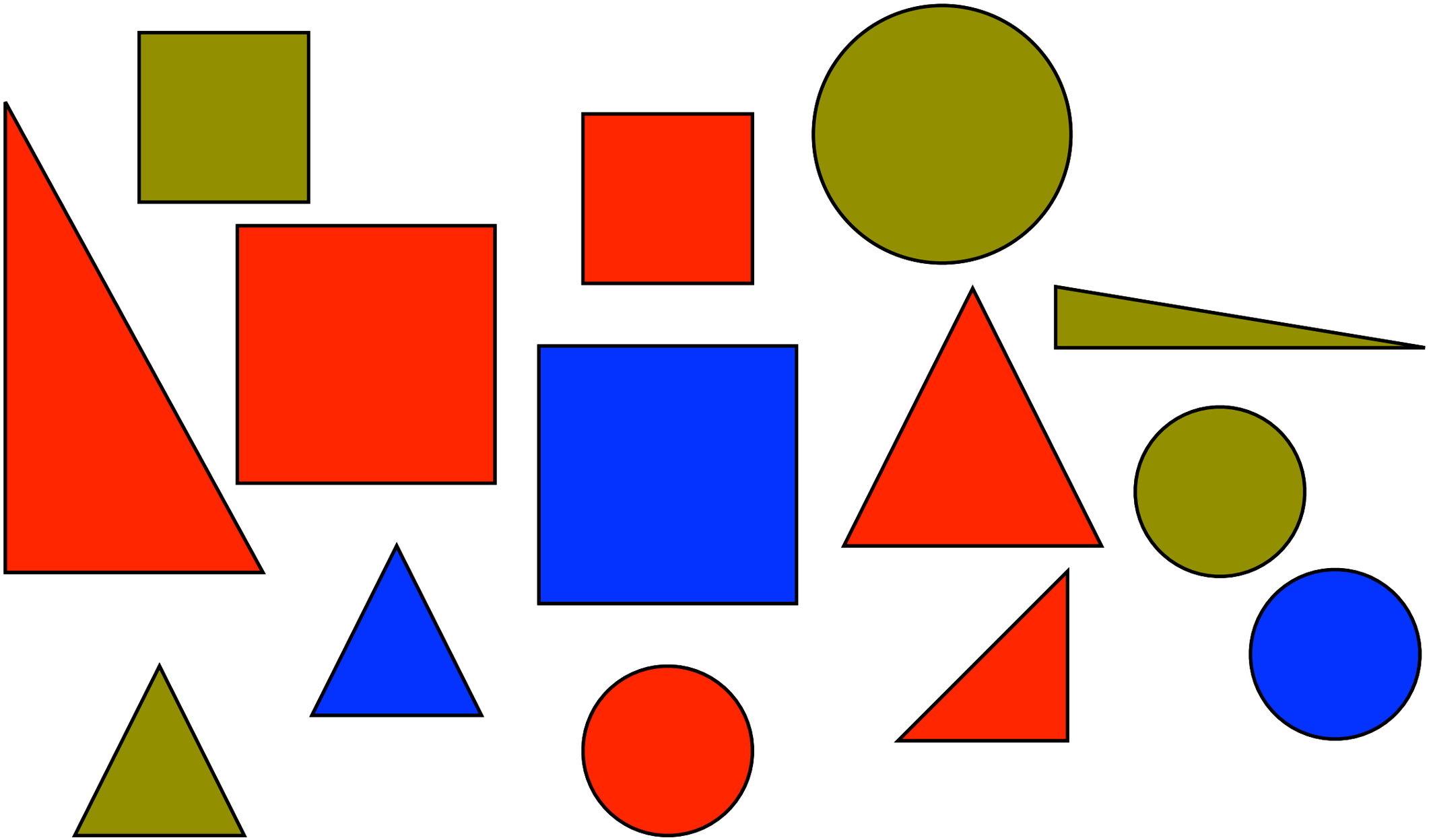
- Machine Learning: developing computer programs that improve their performance with “experience”
- Data Mining: developing methods that discover patterns within large structured datasets
- Statistics: developing methods for the interpretation of data and testing of hypotheses

Text Data Mining in this Module

- Predictive Analysis of Text
  - developing computer programs that automatically <u>detect</u> concepts within a span of text
- Exploratory Analysis of Text:
  - developing computer programs that automatically <u>discover</u> patterns or trends in text collections

Predictive Analysis

example: recognizing triangles



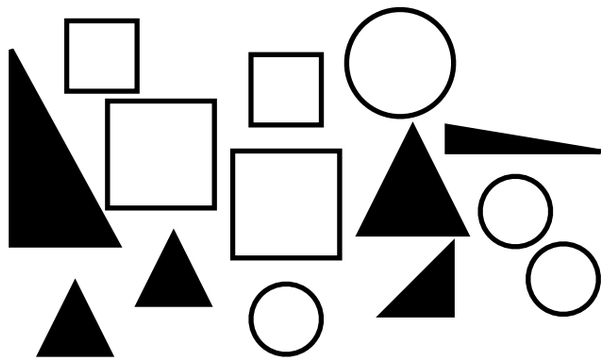
Predictive Analysis
example: recognizing triangles

- We could imagine writing a “triangle detector” by hand:
  - if shape has three sides, then shape = triangle.
  - otherwise, shape = other
- Alternatively, we could use supervised machine learning!

Predictive Analysis

example: recognizing triangles

training

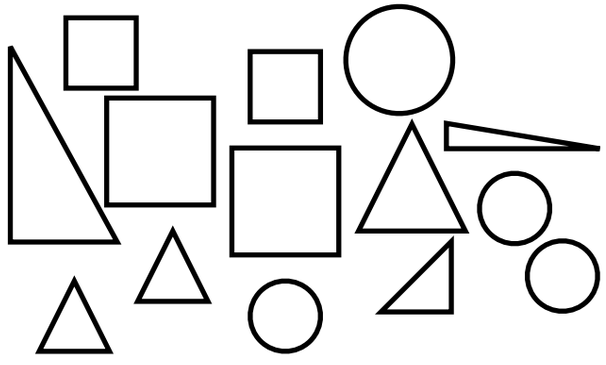


labeled examples

machine learning algorithm

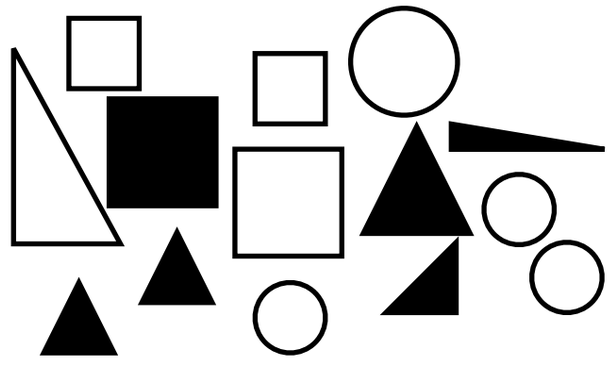
model

testing



new, unlabeled examples

model



predictions

Predictive Analysis

concepts, instances, and features

color	size	# slides	equal sides	...	label
red	big	3	no	...	yes
green	big	3	yes	...	yes
blue	small	inf	yes	...	no
blue	small	4	yes	...	no
⋮	⋮	⋮	⋮	⋮	⋮
red	big	3	yes	...	yes

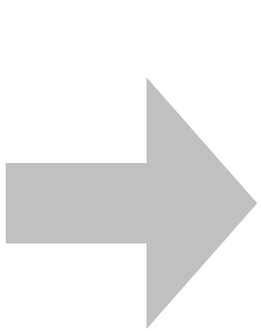
Predictive Analysis

concepts, instances, and features

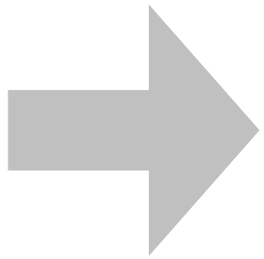
training

color	size	sides	equal sides	...	label
red	big	3	no	...	yes
green	big	3	yes	...	yes
blue	small	inf	yes	...	no
blue	small	4	yes	...	no
⋮	⋮	⋮	⋮	⋮	⋮
red	big	3	yes	...	yes

labeled examples



machine learning algorithm

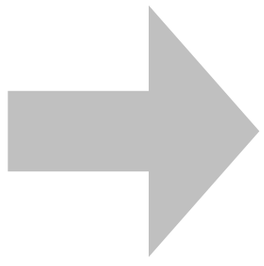


model

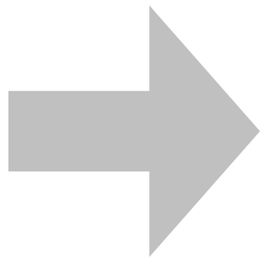
testing

color	size	sides	equal sides	...	label
red	big	3	no	...	???
green	big	3	yes	...	???
blue	small	inf	yes	...	???
blue	small	4	yes	...	???
⋮	⋮	⋮	⋮	⋮	???
red	big	3	yes	...	???

new, unlabeled examples



model



color	size	sides	equal sides	...	label
red	big	3	no	...	yes
green	big	3	yes	...	yes
blue	small	inf	yes	...	no
blue	small	4	yes	...	no
⋮	⋮	⋮	⋮	⋮	⋮
red	big	3	yes	...	yes

predictions

Predictive Analysis

basic ingredients

1. Training data: a set of examples of the concept we want to automatically recognize
2. Representation: a set of features that we believe are useful in recognizing the desired concept
3. Learning algorithm: a computer program that uses the training data to learn a predictive model of the concept

Predictive Analysis
basic ingredients

4. Model: a mathematical function that describes a predictive relationship between the feature values and the presence/absence of the concept

5. Test data: a set of previously unseen examples used to estimate the model's effectiveness

6. Performance metric: a statistic used measure the predictive effectiveness of the model

Predictive Analysis
applications

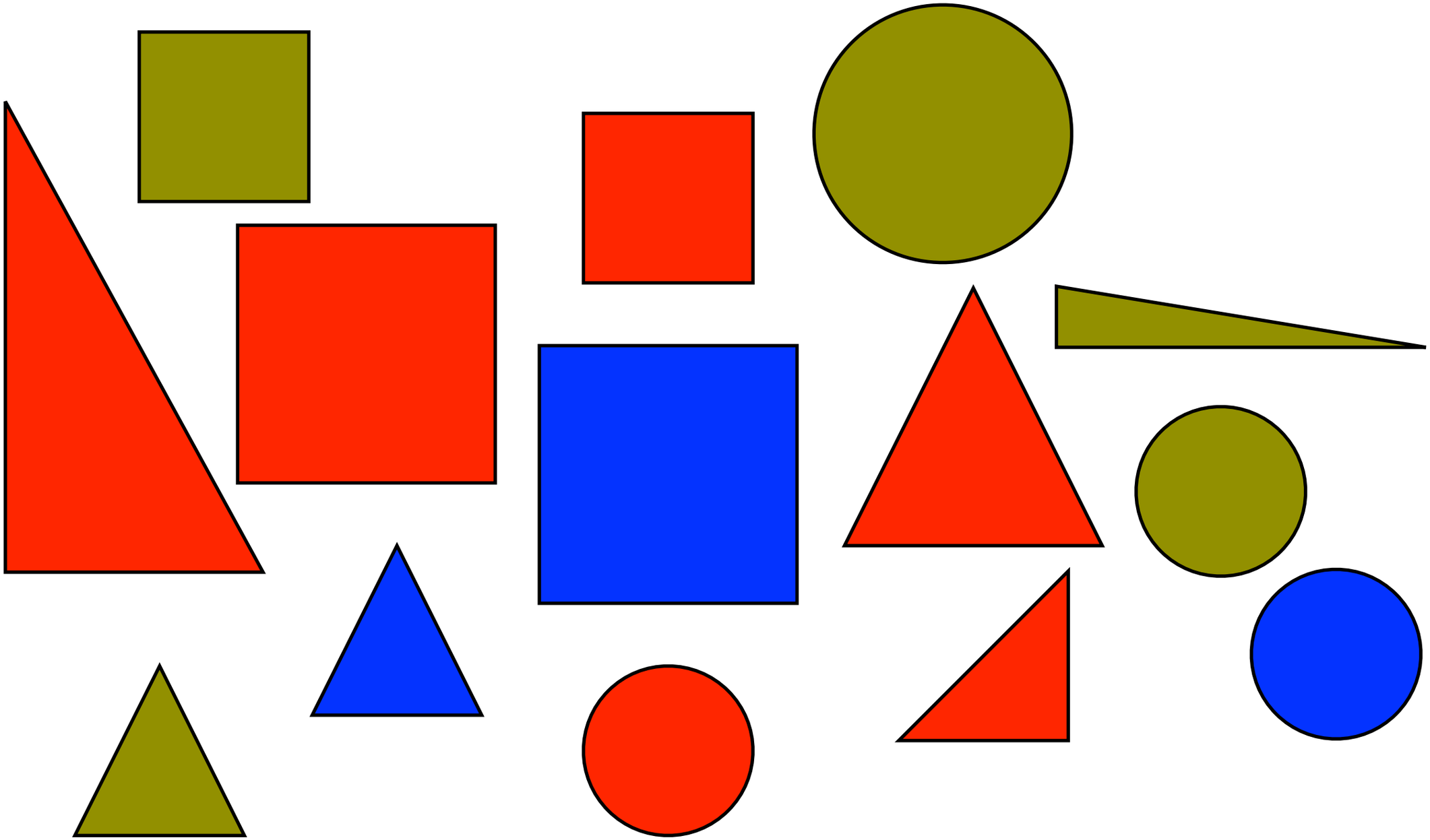
- Topic categorization
- Opinion mining
- Sentiment analysis
- Bias or viewpoint detection
- Discourse analysis (e.g., student retention)
- Forecasting and nowcasting
- Document retention (?)

What Could Possibly Go Wrong?

1. Bad feature representation
2. Bad data + misleading correlations
3. Noisy labels for training and testing
4. Bad learning algorithm
5. Misleading evaluation metric

Training data + Representation

what could possibly go wrong?



Training data + Representation

what could possibly go wrong?

color	size	90 deg. angle	equal sides	...	label
red	big	yes	no	...	yes
green	big	no	yes	...	yes
blue	small	no	yes	...	no
blue	small	yes	yes	...	no
⋮	⋮	⋮	⋮	⋮	⋮
red	big	no	yes	...	yes

Training data + Representation

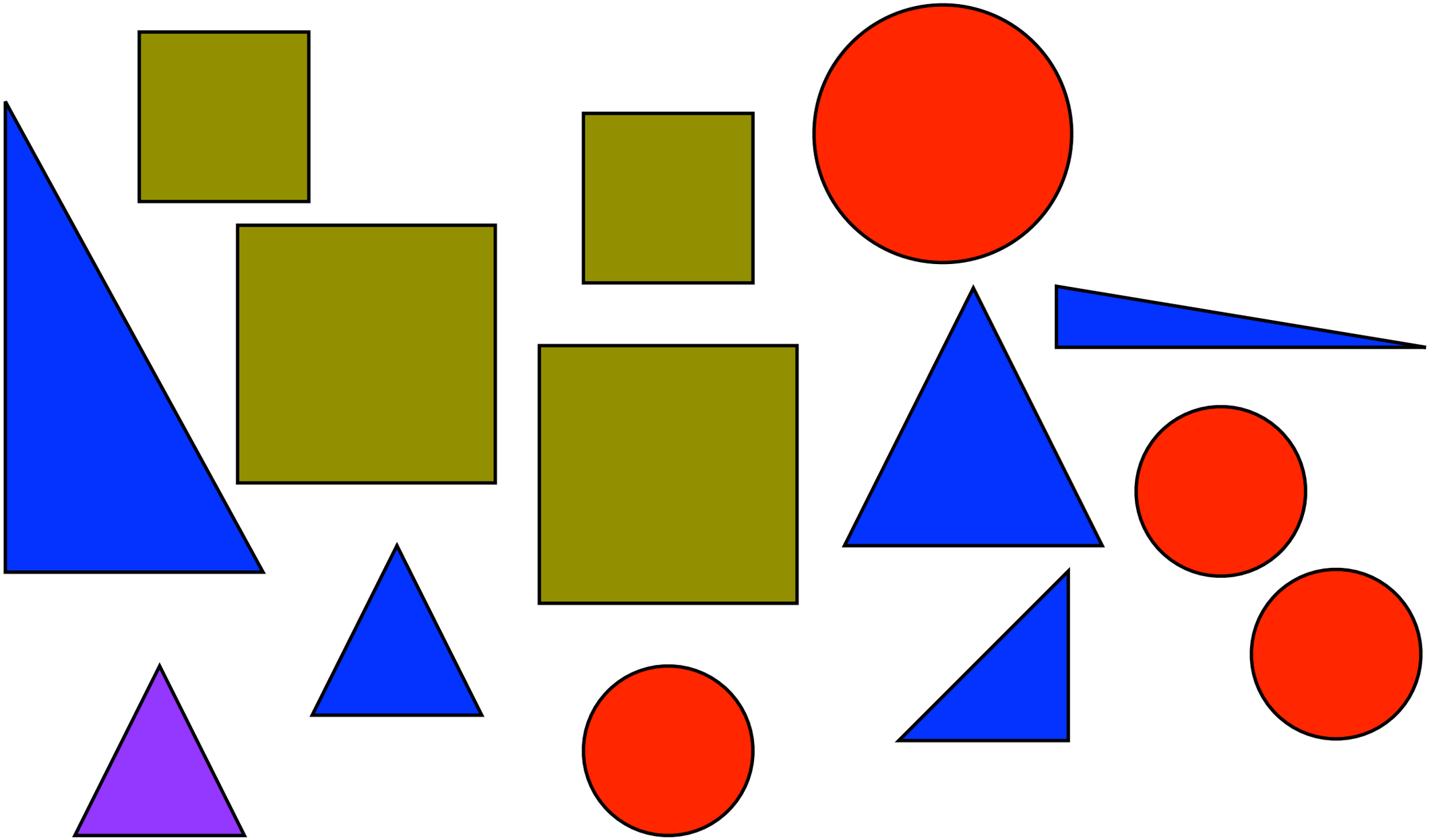
what could possibly go wrong?

color	size	90 deg. angle	equal sides	...	label
red	big	yes	no	...	yes
green	big	no	yes	...	yes
blue	small	no	yes	...	no
blue	small	yes	yes	...	no
⋮	⋮	⋮	⋮	⋮	⋮
red	big	no	yes	...	yes

1. bad feature representation!

Training data + Representation

what could possibly go wrong?



Training data + Representation

what could possibly go wrong?

color	size	# slides	equal sides	...	label
blue	big	3	no	...	yes
blue	big	3	yes	...	yes
red	small	inf	yes	...	no
green	small	4	yes	...	no
⋮	⋮	⋮	⋮	⋮	⋮
blue	big	3	yes	...	yes

Training data + Representation

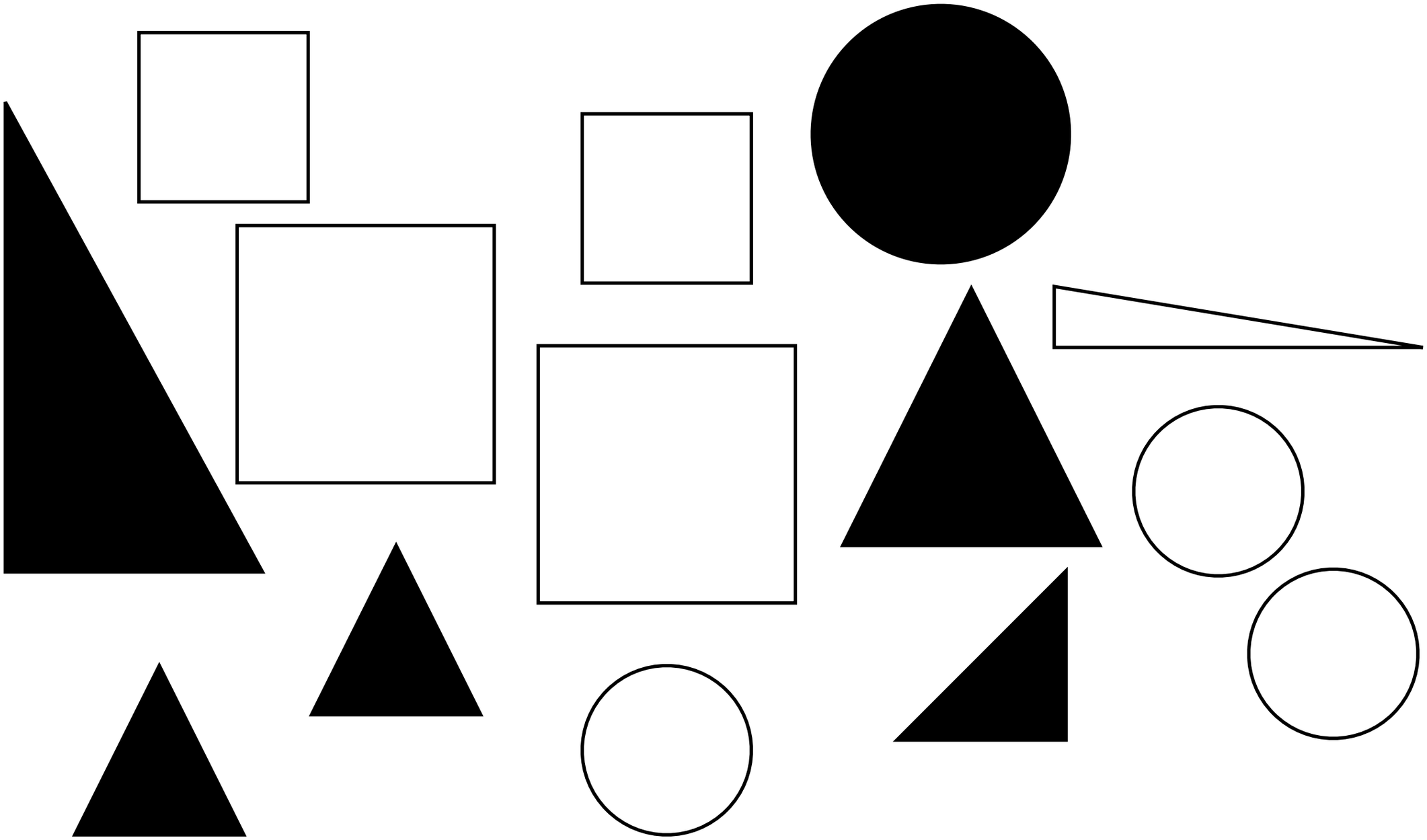
what could possibly go wrong?

color	size	# slides	equal sides	...	label
blue	big	3	no	...	yes
blue	big	3	yes	...	yes
red	small	inf	yes	...	no
green	small	4	yes	...	no
⋮	⋮	⋮	⋮	⋮	⋮
blue	big	3	yes	...	yes

2. bad data + misleading correlations

Training data + Representation

what could possibly go wrong?



Training data + Representation

what could possibly go wrong?

color	size	# slides	equal sides	...	label
white	big	3	no	...	yes
white	big	3	no	...	no
white	small	inf	yes	...	yes
white	small	4	yes	...	no
⋮	⋮	⋮	⋮	⋮	⋮
white	big	3	yes	...	yes

3. noisy training data!

Learning Algorithm + Model

what could possibly go wrong?

- Linear classifier

$$y = \begin{cases} 1 & \text{if } w_0 + \sum_{j=1}^{n} w_j x_j > 0 \\ 0 & \text{otherwise} \end{cases}$$

Learning Algorithm + Model

what could possibly go wrong?

- Linear classifier

$$y = \begin{cases} 1 & \text{if } w_0 + \sum_{j=1}^{n} w_j x_j > 0 \\ 0 & \text{otherwise} \end{cases}$$

parameters learned by the model

predicted value (e.g., 1 = positive, 0 = negative)

Learning Algorithm + Model

what could possibly go wrong?

test instance

f_1	f_2	f_3
0.5	1.0	0.2

model weights

w_0	w_1	w_2	w_3
2.0	-5.0	2.0	1.0

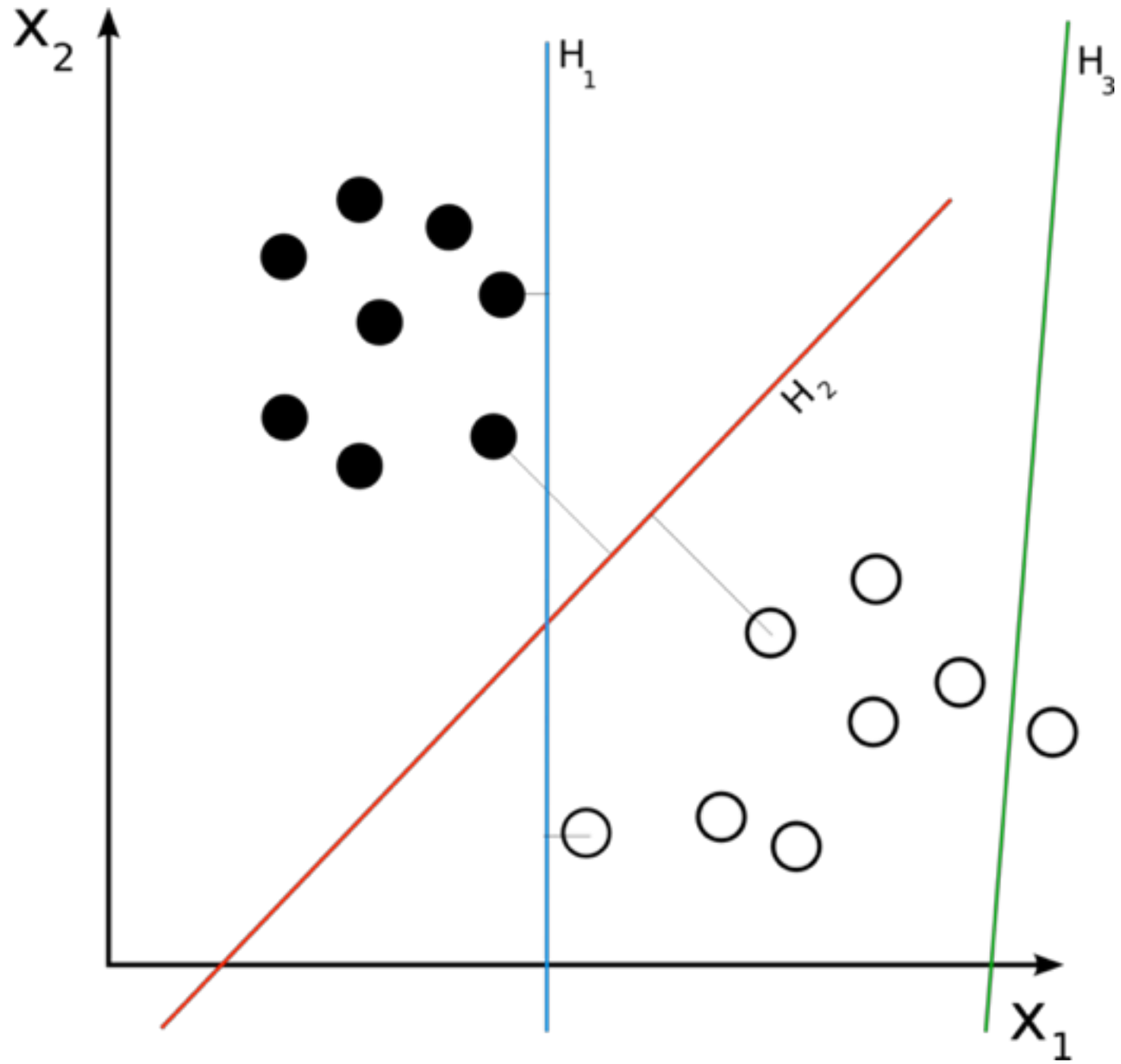
output = 2.0 + (0.50 x -5.0) + (1.0 x 2.0) + (0.2 x 1.0)

output = 1.7

output prediction = positive

Learning Algorithm + Model

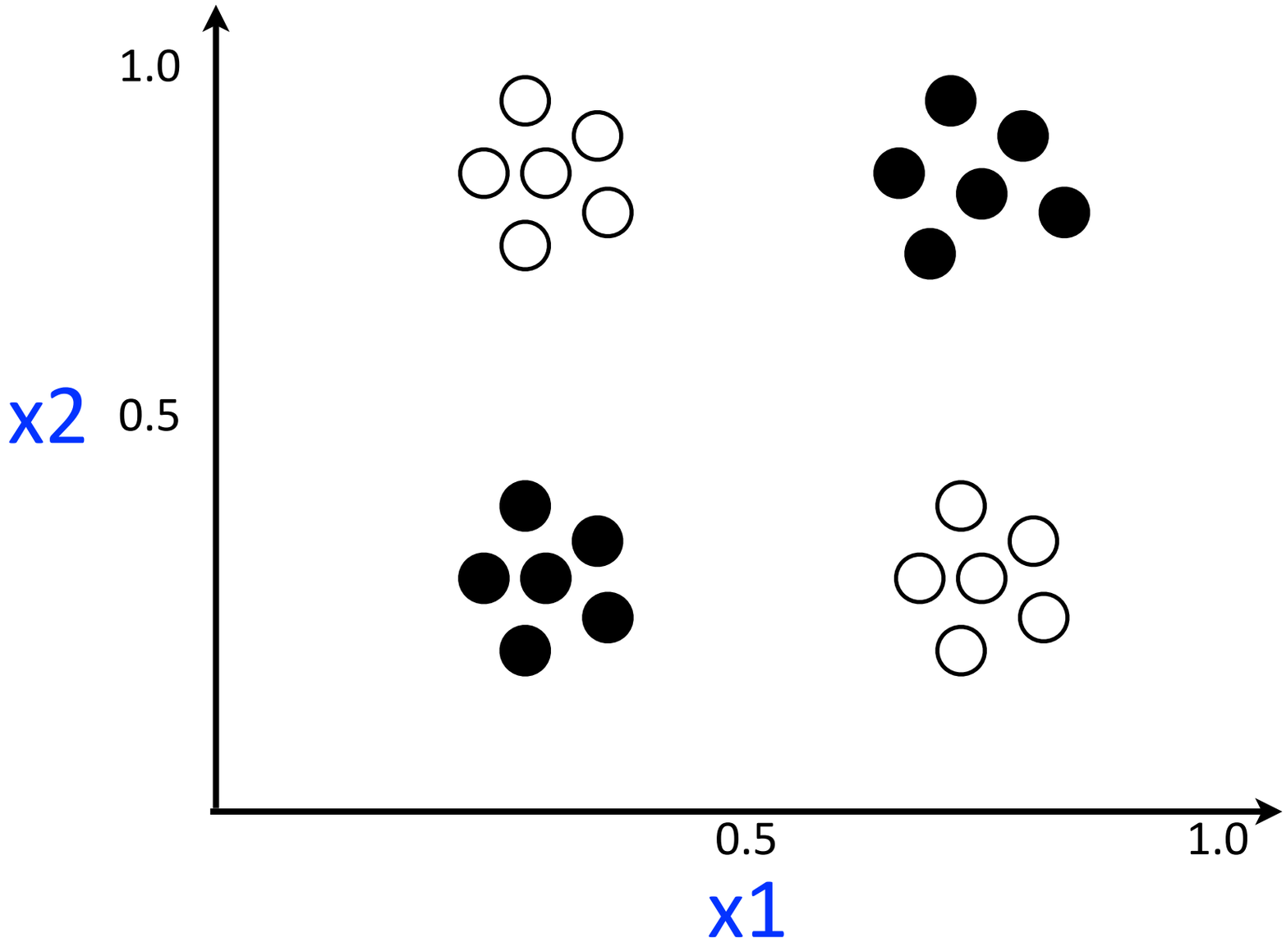
what could possibly go wrong?



(source: http://en.wikipedia.org/wiki/File:Svm_separating_hyperplanes.png)

Learning Algorithm + Model

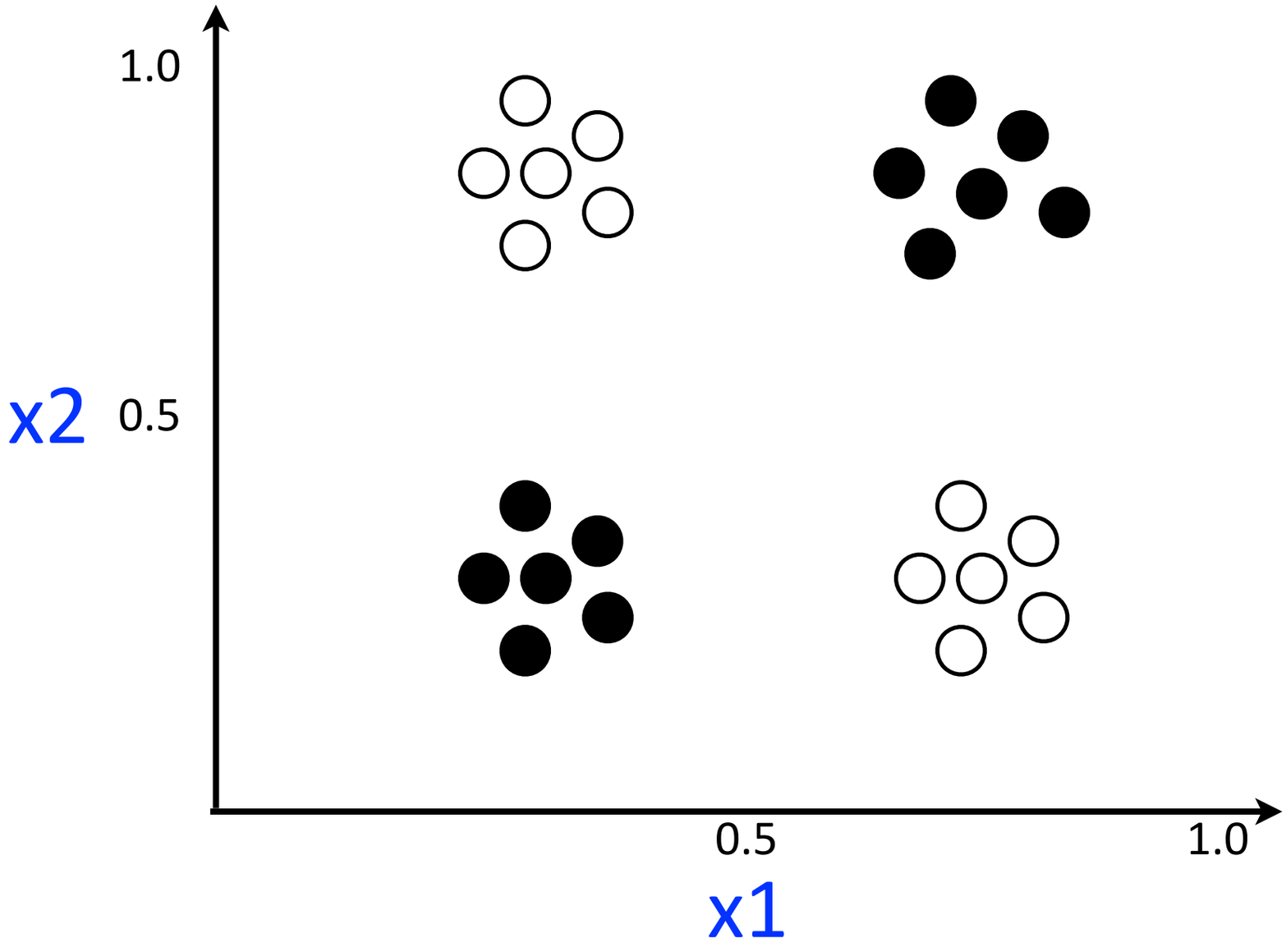
what could possibly go wrong?



- Would a linear classifier do well on positive (black) and negative (white) data that looks like this?

Learning Algorithm + Model

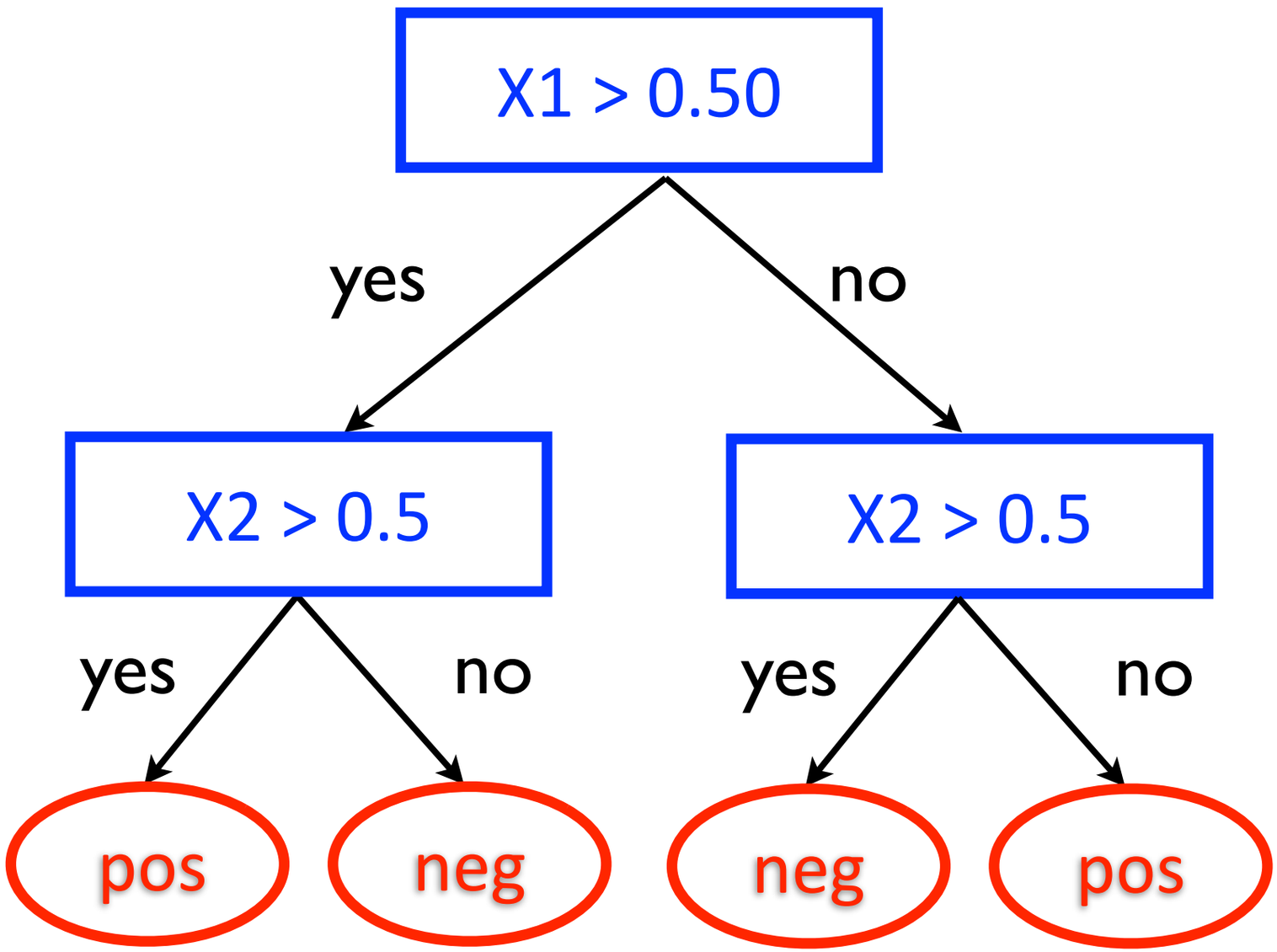
what could possibly go wrong?



4. Bad learning algorithm!

Learning Algorithm + Model

what could possibly go wrong?



Evaluation Metric

what could possibly go wrong?

- Most evaluation metrics can be understood using a contingency table

predicted \ true	triangle	other
triangle	A	B
other	C	D

- What number(s) do we want to maximize?
- What number(s) do we want to minimize?

Evaluation Metric

what could possibly go wrong?

- True positives (A): number of triangles correctly predicted as triangles
- False positives (B): number of "other" incorrectly predicted as triangles
- False negatives (C): number of triangles incorrectly predicted as "other"
- True negatives (D): number of "other" correctly predicted as "other"

predicted \ true	triangle	other
triangle	A	B
other	C	D

Evaluation Metric

what could possibly go wrong?

- Accuracy: percentage of predictions that are correct (i.e., true positives and true negatives)

$$\frac{(? + ?)}{(? + ? + ? + ?)}$$

predicted \ true	triangle	other
triangle	A	B
other	C	D

Evaluation Metric

what could possibly go wrong?

- Accuracy: percentage of predictions that are correct (i.e., true positives and true negatives)

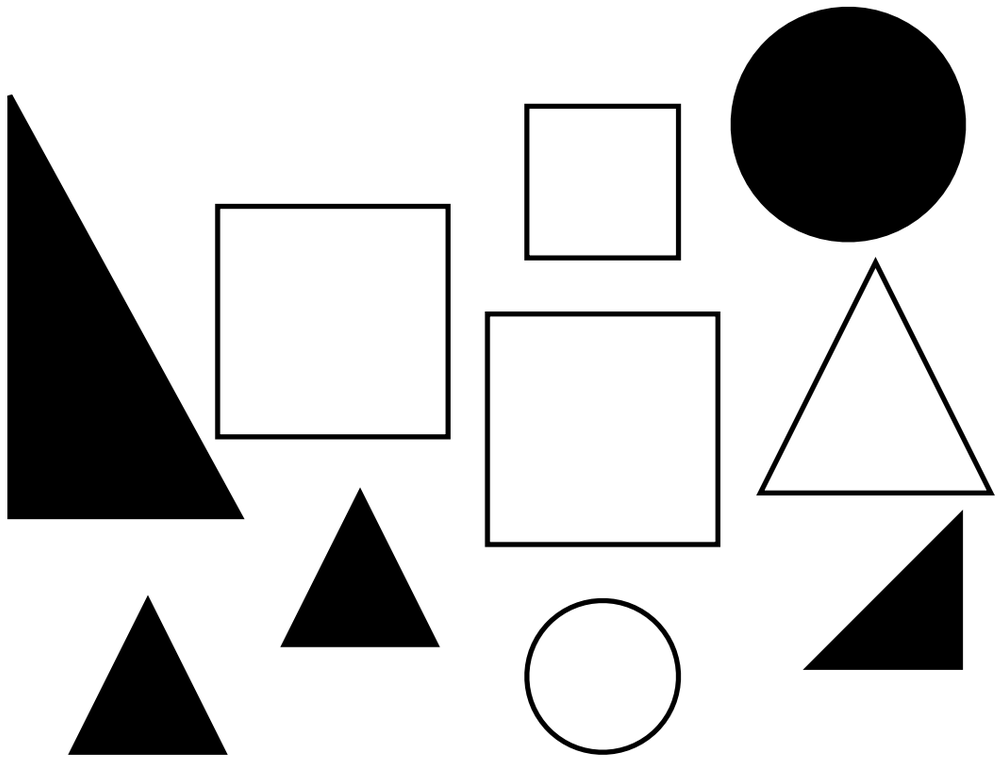
$$\frac{(A + D)}{(A + B + C + D)}$$

predicted \ true	triangle	other
triangle	A	B
other	C	D

Evaluation Metric

what could possibly go wrong?

- Accuracy: percentage of predictions that are correct (i.e., true positives and true negatives)

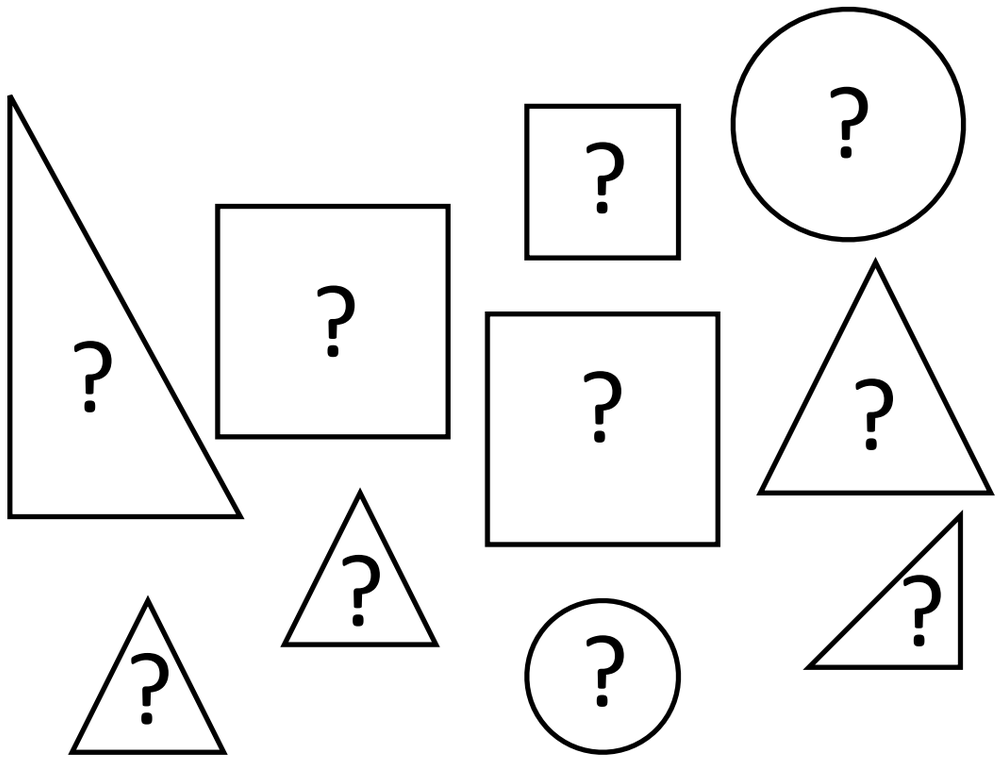


- What is the accuracy of this model?

Evaluation Metric

what could possibly go wrong?

- Interpreting the value of a metric on a particular data set requires some thinking ...

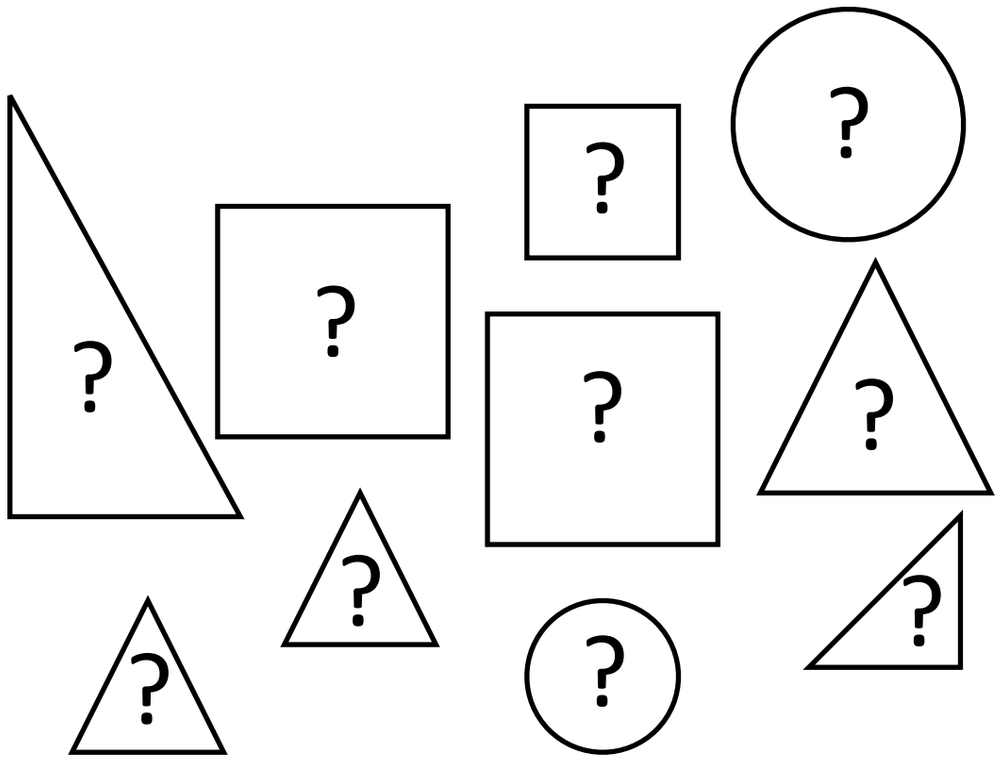


- On this dataset, what would be the expected accuracy of a model that does NO learning (degenerate baseline)?

Evaluation Metric

what could possibly go wrong?

- Interpreting the value of a metric on a particular data set requires some thinking ...



5. Misleading interpretation of a metric value!

What Could Possibly Go Wrong?

1. Bad feature representation
2. Bad data + misleading correlations
3. Noisy labels for training and testing
4. Bad learning algorithm
5. Misleading evaluation metric

Outline: Predictive and Exploratory Analysis

Concepts, Instances, and Features

Human Annotation

Text Representation

Learning Algorithms

Evaluation metrics

Experimentation

Clustering

Hands-on Exercise

Human Annotation

concepts

- Learning algorithms can recognize some concepts better than others

- What are some properties of concepts that are easier to recognize?

Human Annotation

concepts

- Option 1: can a human recognize the concept?
- Option 2: can two or more humans recognize the concept independently and do they agree?
- Option 2 is better.
- In fact, models are sometimes evaluated as an independent assessor
- How does the model's performance compare to the performance of one assessor with respect to another?
  - One assessor produces the "ground truth" and the other produces the "predictions"

Human Annotation

agreement: percent agreement

- Percent agreement: percentage of instances for which both assessors agree that the concept occurs or does not occur

	yes	no
yes	A	B
no	C	D

$$\frac{(? + ?)}{(? + ? + ? + ?)}$$

Human Annotation

agreement: percent agreement

- Percent agreement: percentage of instances for which both assessors agree that the concept occurs or does not occur

	yes	no
yes	A	B
no	C	D

$$\frac{(A + D)}{(A + B + C + D)}$$

Human Annotation

agreement: percent agreement

- Percent agreement: percentage of instances for which both assessors agree that the concept occurs or does not occur

	yes	no	
yes	5	5	10
no	15	75	90
	20	80	

% agreement = ???

Human Annotation

agreement: percent agreement

- Percent agreement: percentage of instances for which both assessors agree that the concept occurs or does not occur

	yes	no	
yes	5	5	10
no	15	75	90
	20	80	

% agreement = (5 + 75) / 100 = 80%

Human Annotation

agreement: percent agreement

- Problem: percent agreement does not account for agreement due to random chance.
- How can we compute the expected agreement due to random chance?
  - Option 1: assume unbiased assessors
  - Option 2: assume biased assessors

Human Annotation

kappa agreement: chance-corrected % agreement

- Option 1: unbiased assessors

	yes	no	
yes	??	??	50
no	??	??	50
	50	50	

Human Annotation

kappa agreement: chance-corrected % agreement

- Option 1: unbiased assessors

	yes	no	
yes	25	25	50
no	25	25	50
	50	50	

Human Annotation

kappa agreement: chance-corrected % agreement

- Option 1: unbiased assessors

	yes	no	
yes	25	25	50
no	25	25	50
	50	50	

random chance % agreement = ???

Human Annotation

kappa agreement: chance-corrected % agreement

- Option 1: unbiased assessors

	yes	no	
yes	25	25	50
no	25	25	50
	50	50	

random chance % agreement = (25 + 25)/100 = 50%

Human Annotation

kappa agreement: chance-corrected % agreement

- Kappa agreement: percent agreement after correcting for the expected agreement due to random chance

$$\mathcal{K} = \frac{P(a) - P(e)}{1 - P(e)}$$

- P(a) = percent of observed agreement
- P(e) = percent of agreement due to random chance

Human Annotation

kappa agreement: chance-corrected % agreement

- Kappa agreement: percent agreement after correcting for the expected agreement due to unbiased chance

	yes	no	
yes	5	5	10
no	15	75	90
	20	80	

	yes	no	
yes	25	25	50
no	25	25	50
	50	50	

$$P(a) = \frac{5+75}{100} = 0.80$$

$$P(e) = \frac{25+25}{100} = 0.50$$

$$\mathcal{K} = \frac{P(a)-P(e)}{1-P(e)} = \frac{0.80-0.50}{1-0.50} = 0.60$$

Human Annotation

kappa agreement: chance-corrected % agreement

- Option 2: biased assessors

	yes	no	
yes	5	5	10
no	15	75	90
	20	80	

biased chance % agreement = ???

Human Annotation

kappa agreement: chance-corrected % agreement

- Kappa agreement: percent agreement after correcting for the expected agreement due to biased chance

	yes	no	
yes	5	5	10
no	15	75	90
	20	80	

$$P(a) = \frac{5+75}{100} = 0.80 \qquad P(e) = \left(\frac{10}{100} \times \frac{20}{100}\right) + \left(\frac{90}{100} \times \frac{80}{100}\right) = 0.74$$

$$\mathcal{K} = \frac{P(a)-P(e)}{1-P(e)} = \frac{0.80-0.74}{1-0.74} = 0.23$$

Human Annotation

data annotation process

- **INPUT**: unlabeled data, annotators, coding manual
- **OUTPUT**: labeled data

1. using the latest coding manual, have <u>all</u> annotators label some previously unseen portion of the data (~10%)
2. measure inter-annotator agreement (Kappa)
3. **IF** agreement < X, **THEN**:
   - refine coding manual using disagreements to resolve inconsistencies and clarify definitions
   - return to 1

   ELSE

   - have annotators label the remainder of the data <u>independently</u> and **EXIT**

Human Annotation

data annotation process

- What is good (Kappa) agreement?
- It depends on who you ask
- According to Landis and Koch, 1977:
  - 0.81 - 1.00: almost perfect
  - 0.61 - 0.70: substantial
  - 0.41 - 0.60: moderate
  - 0.21 - 0.40: fair
  - 0.00 - 0.20: slight
  - < 0.00: no agreement

Outline: Predictive and Exploratory Analysis

Concepts, Instances, and Features

Human Annotation

Text Representation

Learning Algorithms

Evaluation metrics

Experimentation

Clustering

Hands-on Exercise

Text Representation

predicting health-related documents

features **concept**

instances

w_1	w_2	w_3	...	w_n	label
1	1	0	...	0	health
0	0	0	...	0	other
0	0	0	...	0	other
0	1	0	...	1	other
⋮	⋮	⋮	...	0	⋮
1	0	0	...	1	health

Text Representation

predicting positive/negative reviews

features **concept**

instances

w_1	w_2	w_3	...	w_n	label
1	1	0	...	0	positive
0	0	0	...	0	negative
0	0	0	...	0	negative
0	1	0	...	1	negative
⋮	⋮	⋮	...	0	⋮
1	0	0	...	1	positive

Text Representation

predicting liberal/conservative bias

features **concept**

instances

w_1	w_2	w_3	...	w_n	label
1	1	0	...	0	liberal
0	0	0	...	0	conservative
0	0	0	...	0	conservative
0	1	0	...	1	conservative
⋮	⋮	⋮	...	0	⋮
1	0	0	...	1	liberal

Bag of Words Text Representation

- Features correspond to terms in the vocabulary
  - vocabulary: the set of distinct terms appearing in at least one training (positive or negative) instance
  - important: the training and test data must have the same representation!
- Position information and word order is lost
  - dog bites man = man bites dog
- Simple, but effective

Text Processing

- Down-casing: converting text to lower-case
- Tokenization: splitting text into terms or tokens
  - for example: splitting on sequences non-alpha-numeric characters

Text Processing

Steve Carpenter cannot make horror movies. First of all, the casting was very wrong for this movie. The only decent part was the brown haired girl from Buffy the Vampire Slayer. This movies has no gore(usually a key ingredient to a horror movie), no action, no acting, and no suspense(also a key ingredient). Wes Bentley is a good actor but he is so dry and plain in this that it's sad. There were a few parts that were supposed to be funny(continuing the teen horror/ comedy movies) and no one laughed in the audience. I thought that this movie was rated R, and I didn't pay attention and realized it had been changed to PG-13. Anyway, see this movie if you liked I Still Know What You Did Last Summer. That's the only type of person who would find this movie even remotely scary. And seriously, this is to you Steve Carpenter, stop making horror movies. This movie makes Scream look like Texas Chainsaw Massacre.

Text Processing

down-casing

steve carpenter cannot make horror movies. first of all, the casting was very wrong for this movie. the only decent part was the brown haired girl from buffy the vampire slayer. this movies has no gore(usually a key ingredient to a horror movie), no action, no acting, and no suspense(also a key ingredient). wes bentley is a good actor but he is so dry and plain in this that it's sad. there were a few parts that were supposed to be funny(continuing the teen horror/ comedy movies) and no one laughed in the audience. i thought that this movie was rated r, and i didn't pay attention and realized it had been changed to pg-13. anyway, see this movie if you liked i still know what you did last summer. that's the only type of person who would find this movie even remotely scary. and seriously, this is to you steve carpenter, stop making horror movies. this movie makes scream look like texas chainsaw massacre.

Text Processing
tokenization

steve carpenter cannot make horror movies first of all the casting was very wrong for this movie the only decent part was the brown haired girl from buffy the vampire slayer this movies has no gore usually a key ingredient to a horror movie no action no acting and no suspense also a key ingredient wes bentley is a good actor but he is so dry and plain in this that it s sad there were a few parts that were supposed to be funny continuing the teen horror comedy movies and no one laughed in the audience i thought that this movie was rated r and i didn t pay attention and realized it had been changed to pg 13 anyway see this movie if you liked i still know what you did last summer that s the only type of person who would find this movie even remotely scary and seriously this is to you steve carpenter stop making horror movies this movie makes scream look like texas chainsaw massacre

Bag of Words Text Representation

- Which vocabulary terms should we include as features?
- All of them?
  - why might this be a good idea?
  - why might this be a bad idea?

Bag of Words Text Representation

Steve Carpenter cannot make horror movies. First of all, the casting was very wrong for this movie. The only decent part was the brown haired girl from Buffy the Vampire Slayer. This movies has no gore(usually a key ingredient to a horror movie), no action, no acting, and no suspense(also a key ingredient). Wes Bentley is a good actor but he is so dry and plain in this that it's sad. There were a few parts that were supposed to be funny(continuing the teen horror/ comedy movies) and no one laughed in the audience. I thought that this movie was rated R, and I didn't pay attention and realized it had been changed to PG-13. Anyway, see this movie if you liked I Still Know What You Did Last Summer. That's the only type of person who would find this movie even remotely scary. And seriously, this is to you Steve Carpenter, stop making horror movies. This movie makes Scream look like Texas Chainsaw Massacre.

Bag of Words Text Representation

- Hands-on Exercise training set:
  - Number of Instances: 2,000
  - Number of unique terms: 25,637
  - Number of term occurrences: 472,012
- Is there a danger in having 12 times more features than instances?
- We should reduce the feature representation to the most meaningful ones

Feature Selection

- Objective: reduce the feature set to only the most potentially useful
- Unsupervised Feature Selection
  - does not require training data
  - potentially useful features are selected using term statistics
- Supervised Feature Selection
  - requires training data (e.g., positive/negative labels)
  - potentially useful features are selected using co-occurrence statistics between terms and the target label

Unsupervised Feature Selection

Statistical Properties of Text

- As we all know, language use is highly varied
- There are many ways to convey the same information
- However, there are statistical properties of text that are predictable across domains, and even across languages!
- These can help us determine which terms are less likely to be useful (without requiring training labels)

Hands-on Exercise Training Set
statistical properties of text

- Hands-on Exercise training set:
  - Number of Instances: 2,000
  - Number of unique terms: 25,637
  - Number of term occurrences: 472,012

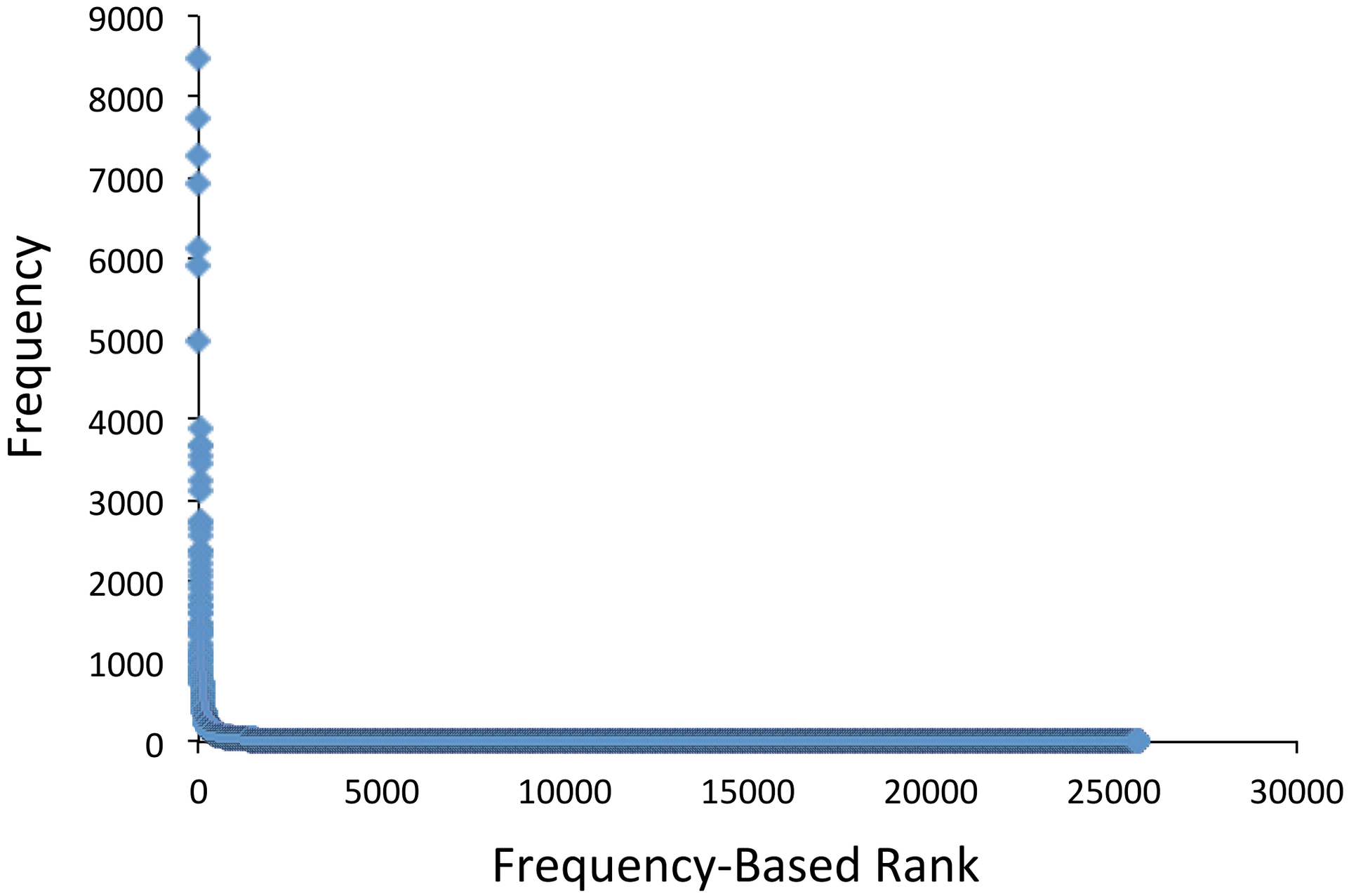
Hands-on Exercise Training Set

term-frequencies

rank	term	frequency	rank	term	frequency
1	the	26638	11	that	5915
2	and	13125	12	s	4975
3	a	12949	13	was	3900
4	of	11715	14	as	3677
5	to	10861	15	movie	3666
6	is	8475	16	for	3540
7	it	7740	17	with	3441
8	in	7259	18	but	3236
9	i	6926	19	film	3124
10	this	6132	20	on	2743

Hands-on Exercise Training Set

term-frequencies



Zipf's Law

- Term-frequency decreases rapidly as a function of rank
- How rapidly?
- Zipf's Law:

$$P_t = \frac{c}{r_t}$$

- P_t = proportion of the data corresponding to term **t**
- **c** = constant
- For English **c** = 0.1 (more or less)
- What does this mean?

Zipf's Law

$$P_t = \frac{c}{r_t} \qquad \mathbf{c} = 0.1$$

- The most frequent term accounts for 10% of the text
- The second most frequent term accounts for 5%
- The third most frequent term accounts for about 3%
- Together, the top 10 account for about 30%
- Together, the top 20 account for about 36%
- Together, the top 50 account for about 45%
  - that's nearly half the text!
- What else does Zipf's law tell us?

Zipf's Law

- With some crafty algebraic manipulation, it also tells us that the faction of terms that occur **n** times is given by:

$$\frac{1}{n(n+1)}$$

- So, what fraction of the terms occur only once?

Zipf's Law

- With some crafty manipulation, it also tells us that the faction of terms that occur **n** times is given by:

$$\frac{1}{n(n+1)}$$

- About half the terms occur only once!
- About 75% of the terms occur 3 times or less!
- About 83% of the terms occur 5 times or less!
- About 90% of the terms occur 10 times or less!

Zipf's Law

Hands-on Exercise training set

- With some crafty manipulation, it also tells us that the faction of terms that occur **n** times is given by:

$$\frac{1}{n(n+1)}$$

- About half the terms occur only once! (43.8%)
- About 75% of the terms occur 3 times or less! (67.5%)
- About 83% of the terms occur 5 times or less! (76.7%)
- About 90% of the terms occur 10 times or less! (86.0%)

Zipf's Law

- Note: the fraction of terms that occur **n** times or less is given by:

$$\sum_{i}^{n} \frac{1}{i(i+1)}$$

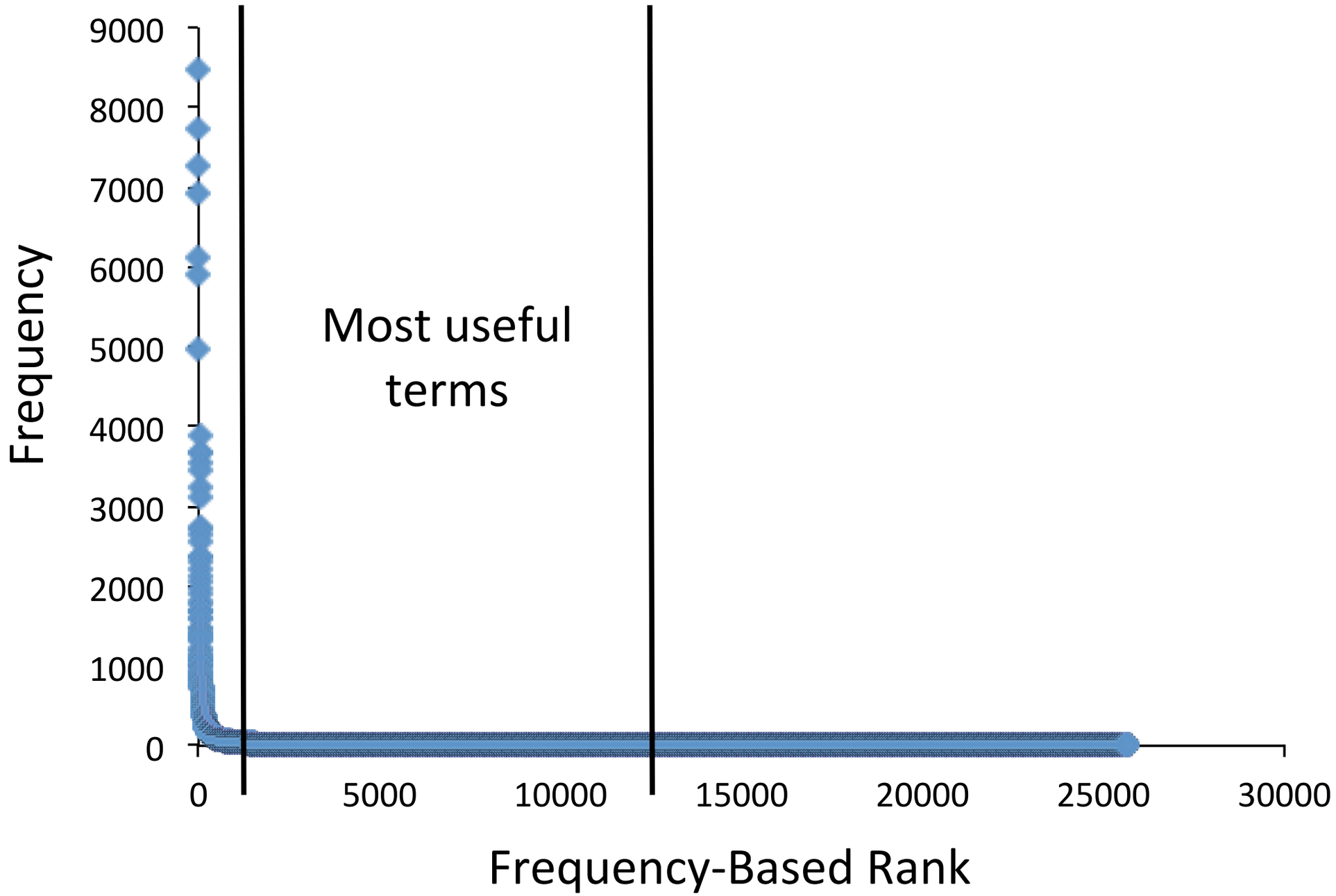
- That is, we have to add the fraction of terms that appear 1, 2, 3, ... up to **n** times

Zipf's Law
Implications for Feature Selection

- The most frequent terms can be ignored
  - assumption: terms that are poor discriminators between instances are likely to be poor discriminators for the target class (e.g., positive/negative sentiment)
- The least frequent terms can be ignored
  - assumption: terms that occur rarely in the training set do not provide enough evidence for learning a model and will occur rarely in the test set

Zipf's Law

Implications for Feature Selection



Zipf's Law
Implications for Feature Selection

- The most frequent terms can be ignored
  - ignore the most frequent 50 terms
  - will account for about 50% of all term occurrences
- The least frequent terms can be ignored
  - ignore terms that occur 5 times or less
  - will account for about 80% of the vocabulary

Verifying Zipf's Law
visualization

Zipf's Law $$f = \frac{k}{r}$$

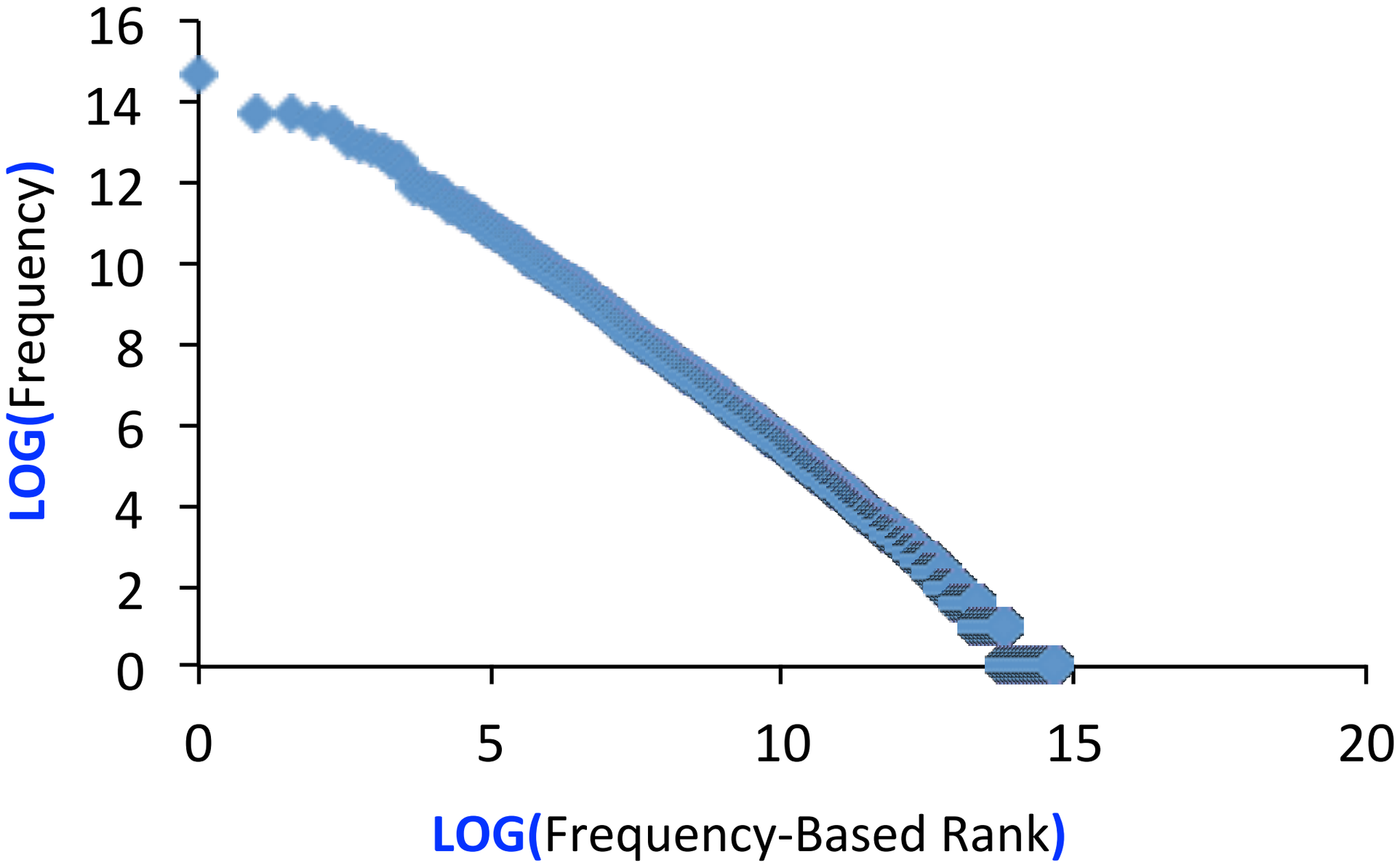
... still Zipf's Law $$\log(f) = \log(\frac{k}{r})$$

... still Zipf's Law $$\log(f) = \log(k) - \log(r)$$

- If Zipf's law holds true, we should be able to plot **log(f)** vs. **log(r)** and see a straight light with a slope of -1

Zipf's Law

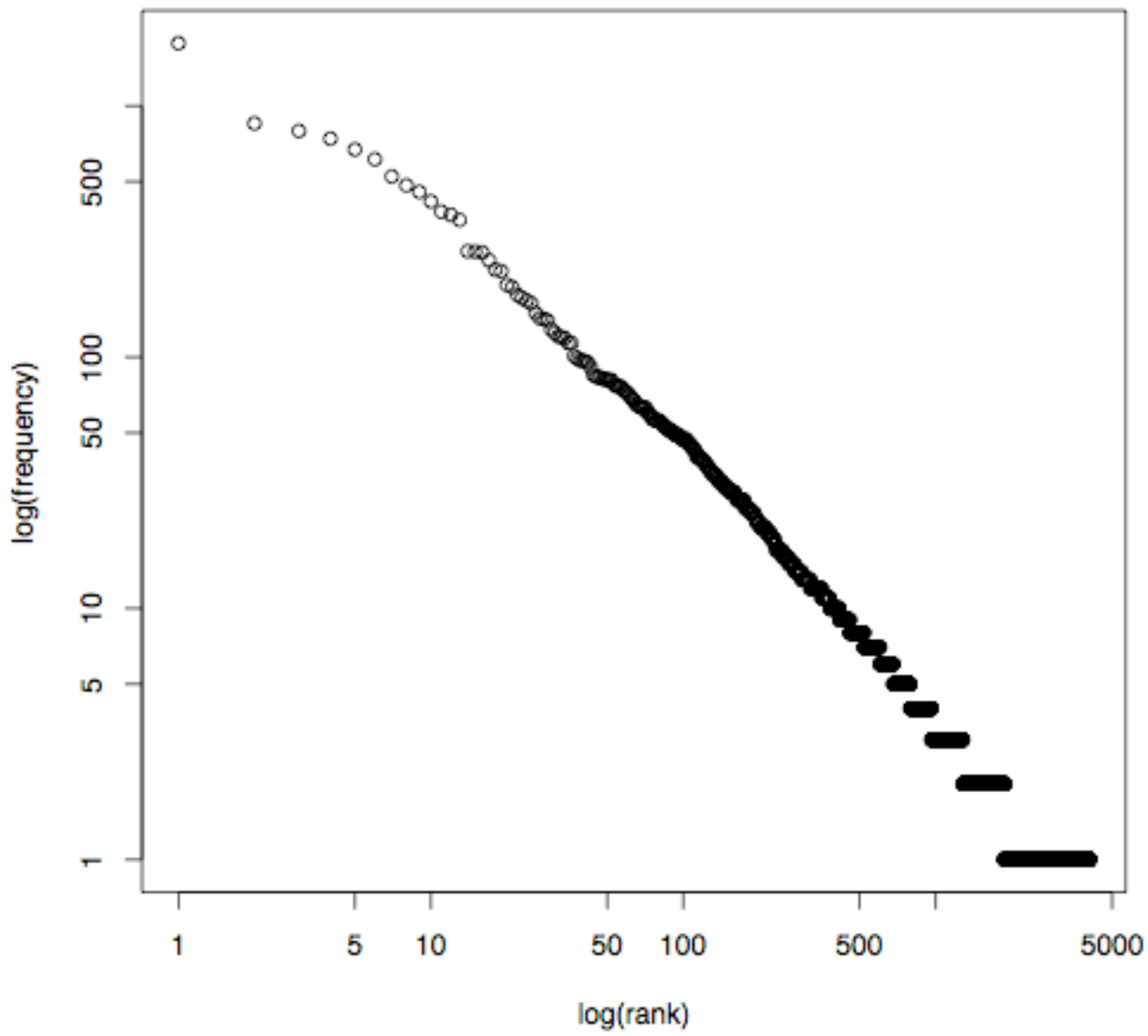
Hands-on Exercise Dataset



Does Zipf's law generalize across domains?

Zipf's Law

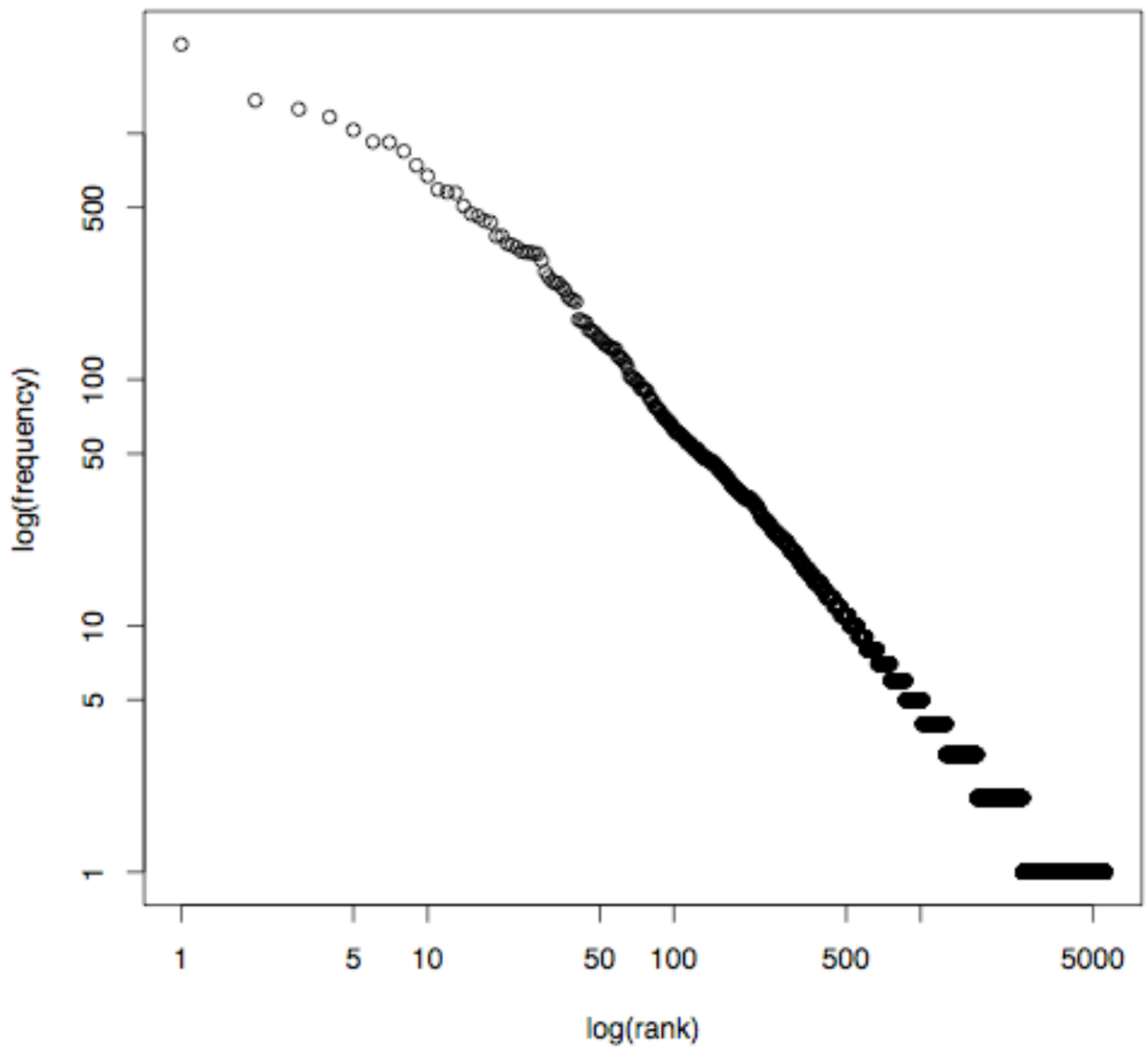
Alice in Wonderland



(text courtesy of Project Gutenberg)

Zipf's Law

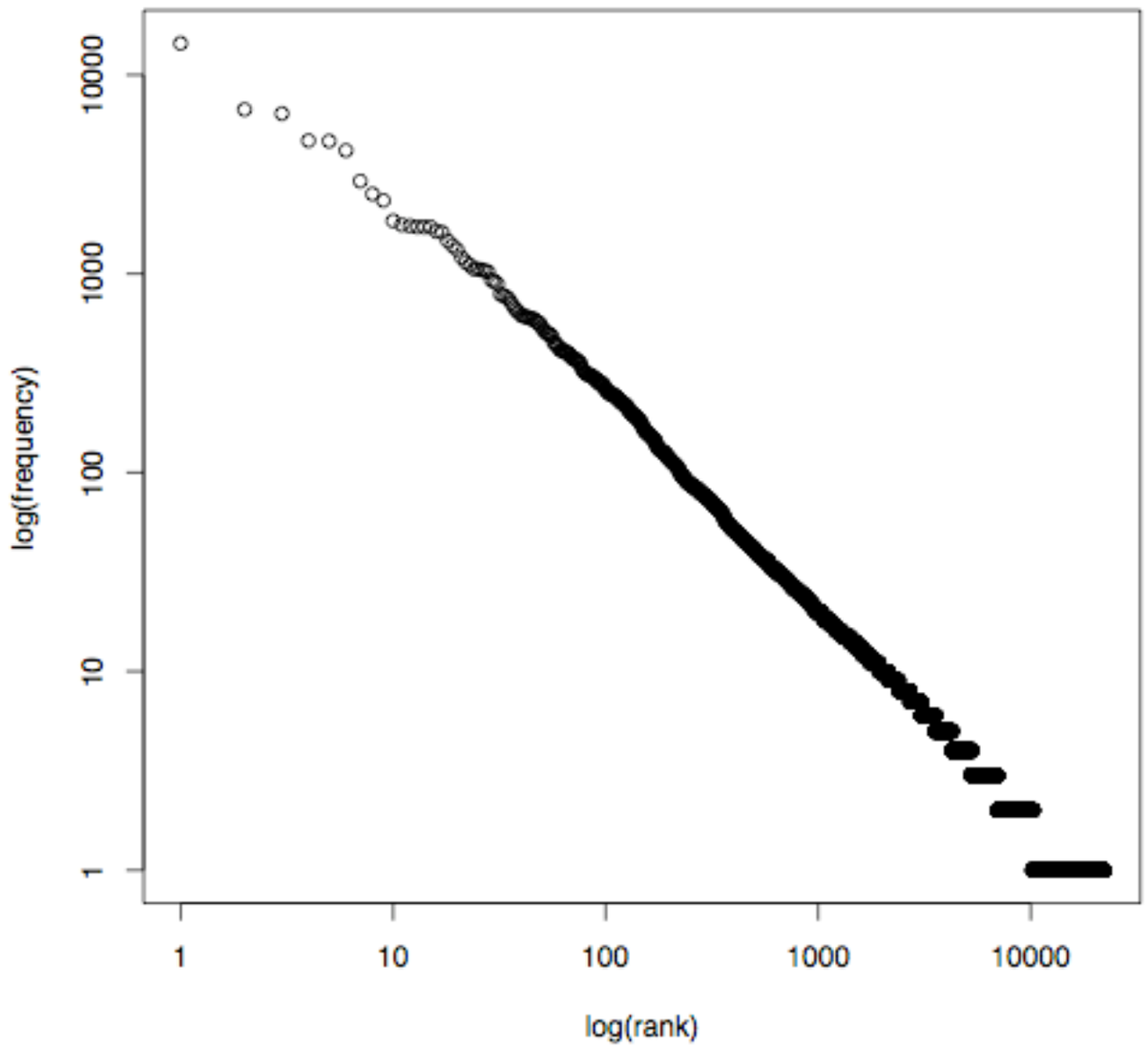
Peter Pan



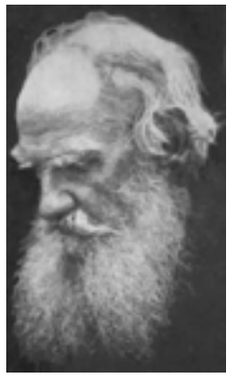
(text courtesy of Project Gutenberg)

Zipf's Law

Moby Dick

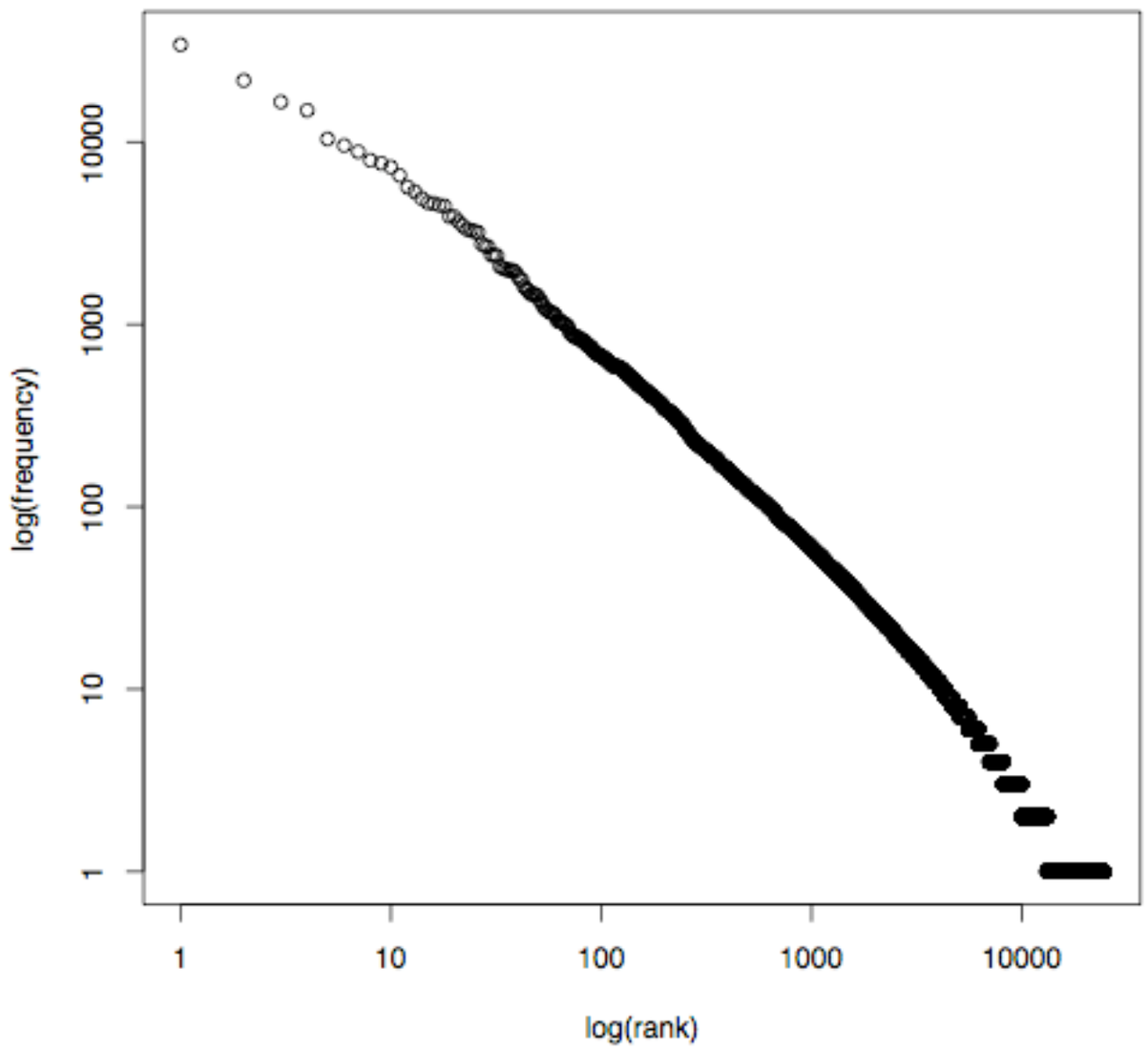


(text courtesy of Project Gutenberg)

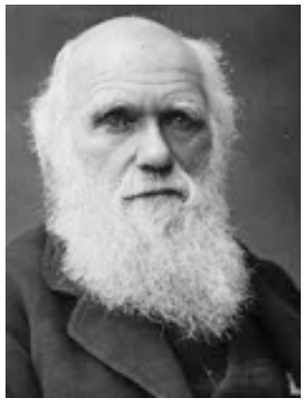


Zipf's Law

War and Peace

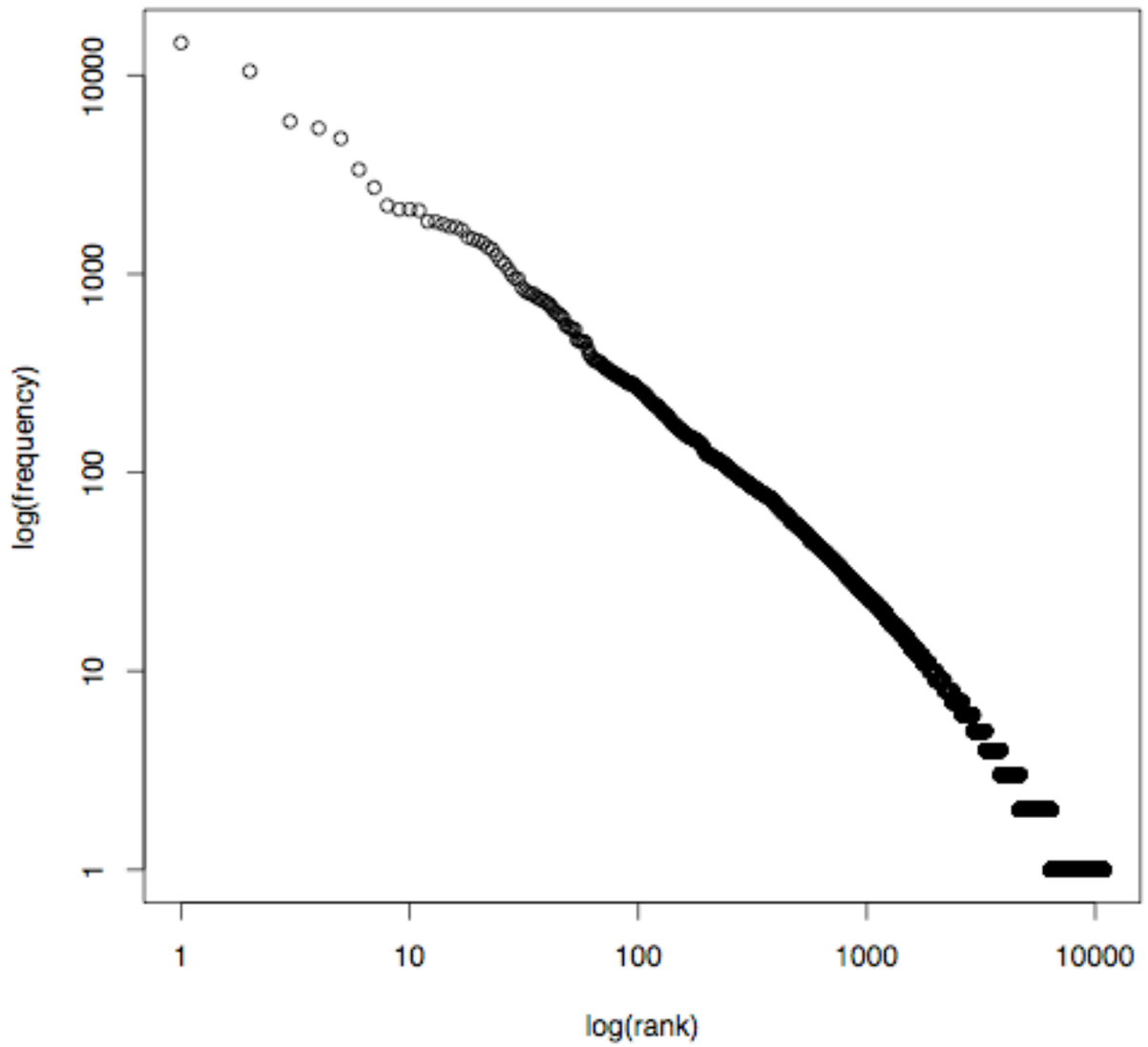


(text courtesy of Project Gutenberg)

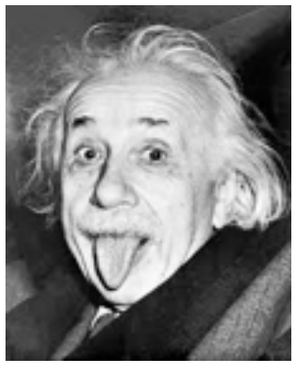


Zipf's Law

On the Origin of Species

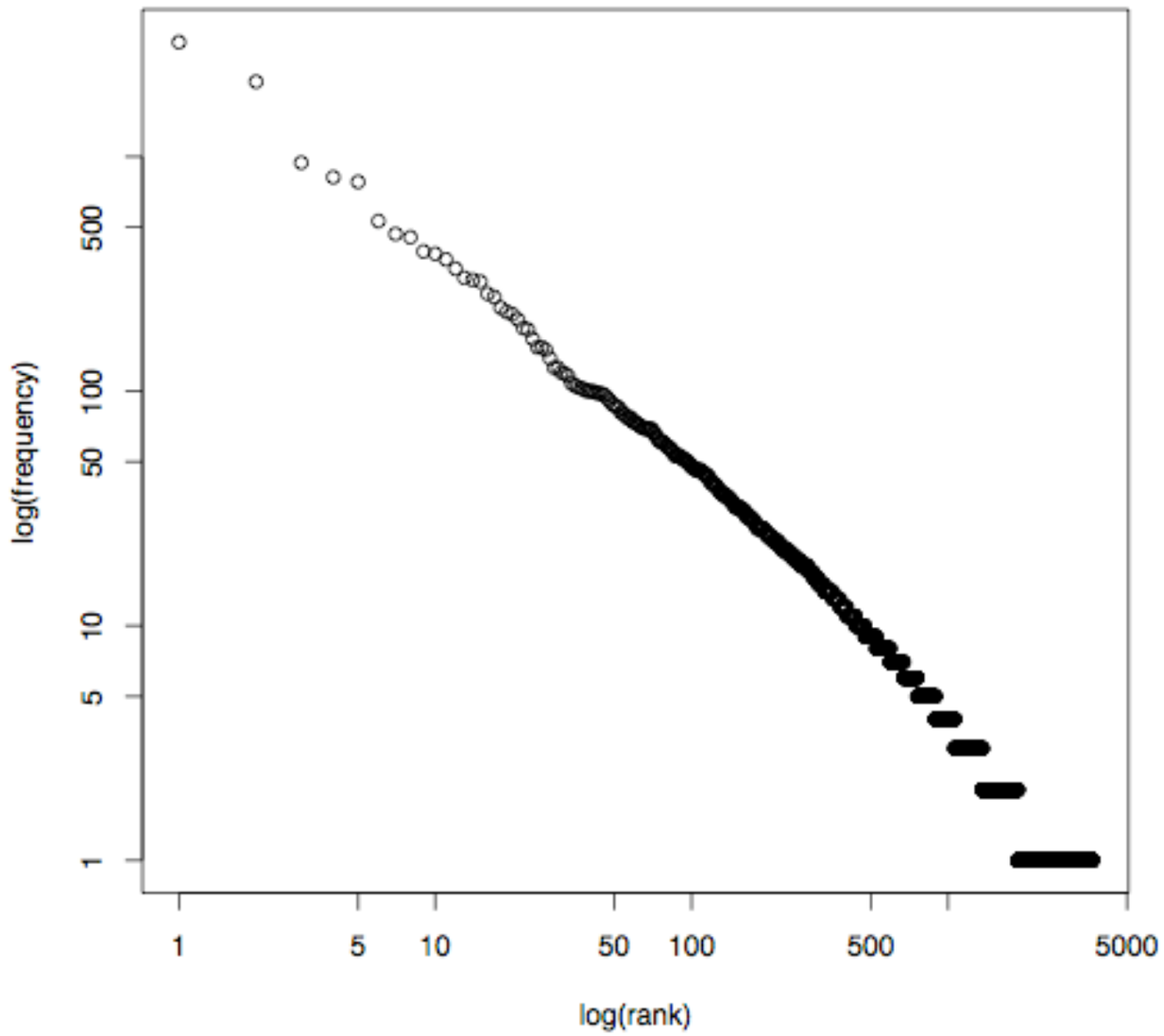


(text courtesy of Project Gutenberg)



Zipf's Law

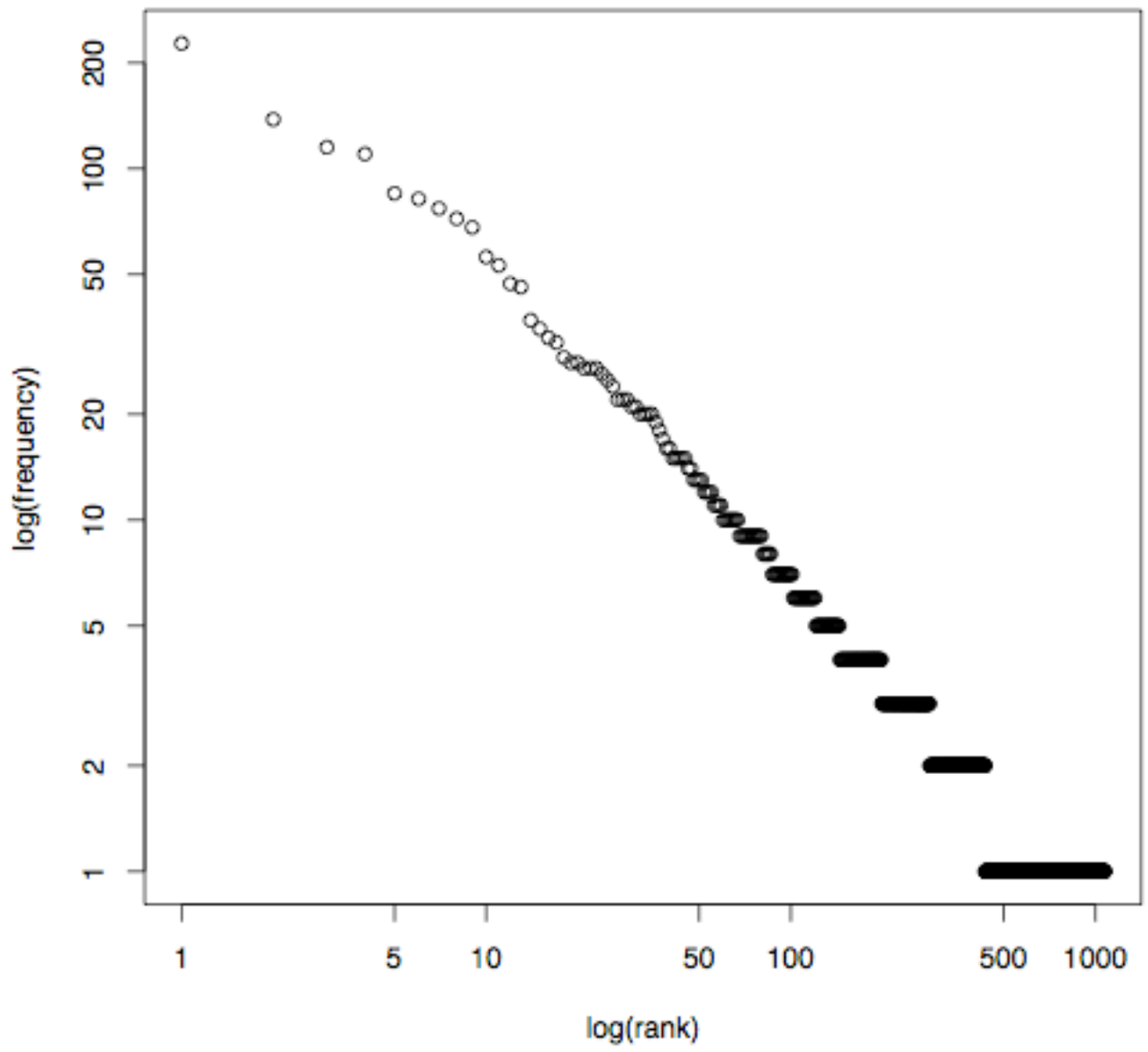
Relativity: The Special and General Theory



(text courtesy of Project Gutenberg)

Zipf's Law

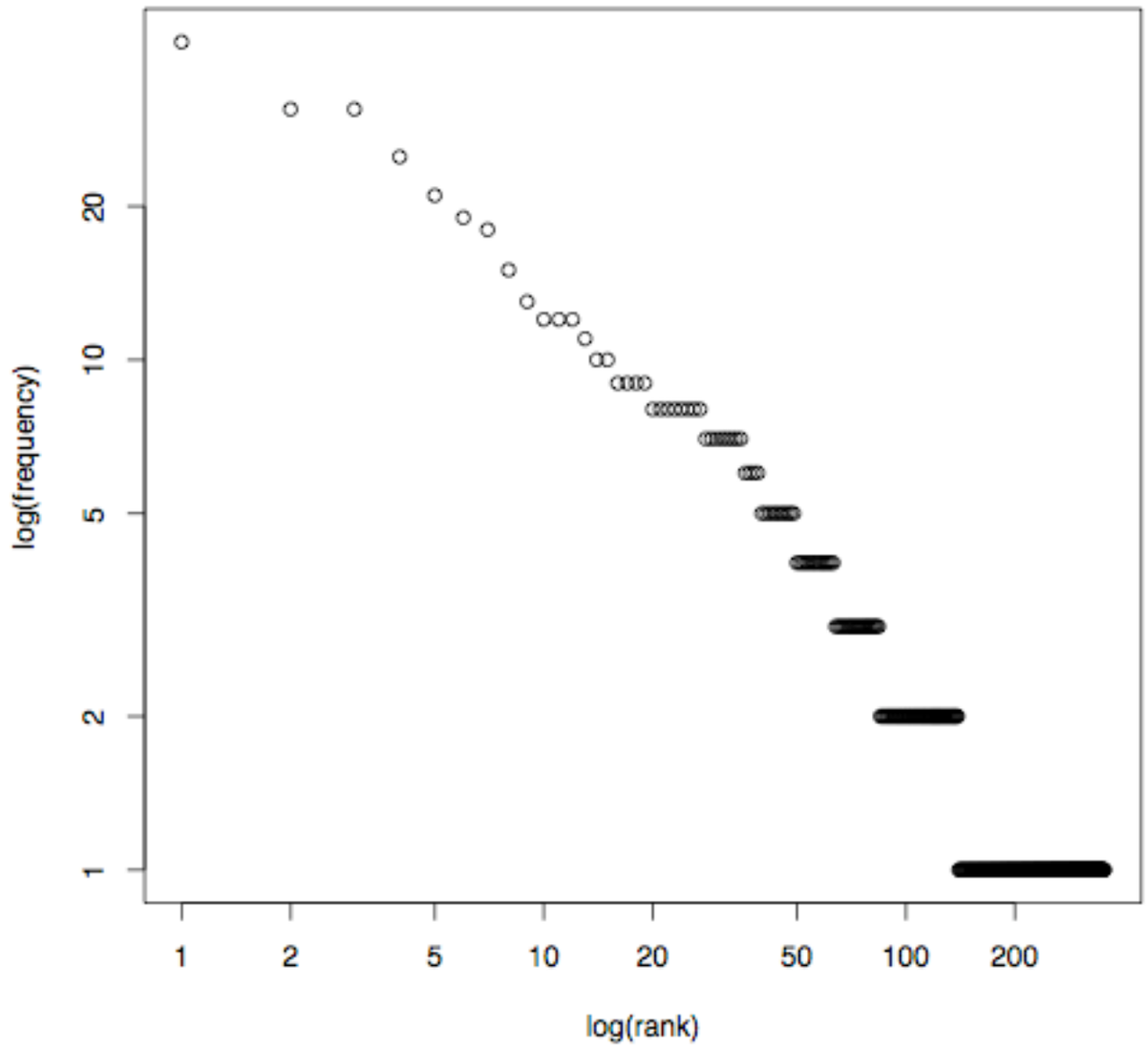
The Tale of Peter Rabbit



(text courtesy of Project Gutenberg)

Zipf's Law

The Three Bears



(text courtesy of Project Gutenberg)

Feature Selection

- Unsupervised Feature Selection
  - does not require training data
  - potentially useful features are selected using term and dataset statistics

- Supervised Feature Selection
  - requires training data (e.g., positive/negative labels)
  - potentially useful features are selected using co-occurrence statistics between terms and the target label

Supervised Feature Selection

- What are the terms that tend to co-occur with a particular class value (e.g., positive or negative)?

Mutual Information

$$\mathrm{MI}(w,c) = \log\left(\frac{P(w,c)}{P(w)P(c)}\right)$$

- **P(w,c)**: the probability that word **w** and class value **c** occur together
- **P(w)**: the probability that word **w** occurs (with or without class value **c**)
- **P(c)**: probability that class value **c** occurs (with or without word **w**)

Mutual Information

$$\mathrm{MI}(w,c) = \log\left(\frac{P(w,c)}{P(w)P(c)}\right)$$

- If **P(w,c) = P(w) P(c)**, it means that the word **w** is independent of class value **c**
- If **P(w,c) > P(w) P(c)**, it means that the word **w** is dependent of class value **c**

Mutual Information

- Every instance falls under one of these quadrants

	class value **c** occurs	class value **c** does not occur
word **w** occurs	a	b
word **w** does not occur	c	d

total # of instances N =
a + b + c + d

P(w, c) = ?

P(c) = ?

P(w) = ?

$$\mathrm{MI}(w,c) = \log\left(\frac{P(w,c)}{P(w)P(c)}\right)$$

Mutual Information

- Every instance falls under one of these quadrants

	class value **c** occurs	class value **c** does not occur
word **w** occurs	a	b
word **w** does not occur	c	d

total # of instances N = a + b + c + d

P(w, c) = a / N

P(c) = (a + c) / N

P(w) = (a + b) / N

$$\mathrm{MI}(w,c) = \log\left(\frac{P(w,c)}{P(w)P(c)}\right)$$

Hands-on Exercise Training Set

terms correlated with positive class

term	MI	term	MI	term	MI
captures	0.69315	urban	0.60614	fellow	0.58192
viewings	0.69315	overlooked	0.59784	masterpiece	0.57808
extraordinary	0.62415	breathtaking	0.59784	legend	0.57536
allows	0.62415	biography	0.59784	awards	0.55962
delight	0.61904	intensity	0.59784	donald	0.55962
wayne	0.61904	represent	0.59784	journey	0.55500
unforgettable	0.61904	elegant	0.59784	traditional	0.55005
sentimental	0.61904	emma	0.59784	seasons	0.55005
touching	0.61619	deliberate	0.59784	mass	0.53900
essence	0.61310	friendship	0.59784	court	0.53900
superb	0.61310	splendid	0.59784	princess	0.53900
underrated	0.61310	desires	0.59784	refreshing	0.53900
devoted	0.60614	terrific	0.59784	drunken	0.53900
frightening	0.60614	delightful	0.59306	adapted	0.53900
perfection	0.60614	gorgeous	0.59306	stewart	0.53900

Hands-on Exercise Training Set

terms correlated with negative class

term	MI	term	MI	term	MI
atrocious	0.693147181	gross	0.613104473	existent	0.575364145
blatant	0.693147181	appalling	0.606135804	dumb	0.572519193
miserably	0.693147181	unintentional	0.606135804	zero	0.571786324
unfunny	0.693147181	drivel	0.606135804	!@#$	0.568849464
unconvincing	0.693147181	pointless	0.60077386	amateurish	0.567984038
stupidity	0.693147181	unbelievably	0.597837001	garbage	0.559615788
blah	0.693147181	blockbuster	0.597837001	dreadful	0.559615788
suck	0.693147181	stinker	0.597837001	horribly	0.559615788
sounded	0.693147181	renting	0.597837001	tedious	0.550046337
redeeming	0.660357358	idiotic	0.597837001	uninteresting	0.550046337
laughable	0.652325186	awful	0.596154915	wasted	0.550046337
downright	0.624154309	lame	0.585516516	insult	0.550046337
irritating	0.619039208	worst	0.58129888	horrible	0.547193268
waste	0.613810438	brain	0.579818495	pretentious	0.546543706
horrid	0.613104473	sucks	0.575364145	offensive	0.546543706

Co-occurrence Statistics

- Mutual Information
- Chi-squared
- Term strength
- Information Gain
- For a nice review, see:
  - Yang and Pedersen. A Comparative Study of Feature Selection for Text Categorization. 1997

Chi Squared

- Every instance falls under one of these quadrants

	class value **c** occurs	class value **c** does not occur
word **w** occurs	a	b
word **w** does not occur	c	d

$$\chi^2(w,c) = \frac{N \times (ad - cb)^2}{(a+c) \times (b+d) \times (a+b) \times (c+d)}$$

Hands-on Exercise Training Set

chi-squared term statistics

term	chi-squared	term	chi-squared	term	chi-squared
bad	160.9971465	best	42.61226642	guy	30.21744225
worst	129.7245814	love	40.85783977	highly	30.18018867
great	114.4167082	even	39.61387169	very	29.04056204
waste	90.05925899	don	38.87461084	masterpiece	28.83716791
awful	84.06935342	superb	38.22460907	amazing	28.79058228
nothing	49.63235294	excellent	36.35817308	fantastic	28.42431877
boring	48.08302214	only	35.37872166	i	28.07171446
!@#$	47.01798462	minutes	34.16970651	redeeming	27.55615262
stupid	47.01038257	worse	33.43003177	dumb	26.86372932
terrible	46.87740534	no	33.13496711	ridiculous	26.73027231
t	46.72237358	poor	32.66596825	any	25.86206897
acting	46.36780576	lame	31.82041653	like	25.69031789
horrible	44.78927425	annoying	31.32494449	mess	25.58837466
supposed	44.48292448	brilliant	30.89314779	poorly	25.58837466
wonderful	43.24661832	make	30.61995968	not	25.47840442

Hands-on Exercise Training Set

chi-squared term statistics

term	chi-squared	term	chi-squared	term	chi-squared
avoid	24.64813529	cheap	22.26804124	gore	19.46385538
plot	24.32739264	favorite	22.21941826	this	19.3814528
loved	24.13368514	always	21.72980415	perfect	19.28060105
oh	24.10901468	laughable	21.4278481	so	19.26007925
lives	23.93399462	family	21.40903284	beautiful	19.25267715
m	23.85882353	better	21.35884719	role	19.14529915
pointless	23.45760278	zero	21.19956379	classic	19.13622759
garbage	22.95918367	unless	20.938872	anything	19.02801032
they	22.8954747	1	20.88669951	unfortunately	18.9261532
or	22.68259489	there	20.4478906	also	18.48036413
script	22.60364052	half	20.23467433	8	18.18641071
terrific	22.46152424	unfunny	20.2020202	suck	18.16347124
performance	22.42822967	low	19.89567408	brain	17.53115039
money	22.34443913	touching	19.86071221	guess	17.52876709
movie	22.34161803	attempt	19.75051975	were	17.49633958

Outline: Predictive and Exploratory Analysis

Concepts, Instances, and Features

Human Annotation

Text Representation

Learning Algorithms

Evaluation metrics

Experimentation

Clustering

Hands-on Exercise

Instance-based Classification

Instance-Based Classification

Motivation

training data

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10	sentiment
1	0	1	0	1	0	0	1	1	0	positive
0	1	0	1	1	0	1	1	0	0	negative
0	1	0	1	1	0	1	0	0	0	negative
0	0	1	0	1	1	0	1	1	1	positive
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	0	1	1	0	0	1	0	1	positive

test instance

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10	sentiment
1	0	1	0	1	0	0	1	1	0	?

Instance-Based Classification

Motivation

training data

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10	sentiment
1	0	1	0	1	0	0	1	1	0	positive
0	1	0	1	1	0	1	1	0	0	negative
0	1	0	1	1	0	1	0	0	0	negative
0	0	1	0	1	1	0	1	1	1	positive
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	0	1	1	0	0	1	0	1	positive

test instance

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10	sentiment
1	0	1	0	1	0	0	1	1	0	?

Instance-Based Classification

Motivation

training data

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10	sentiment
1	0	1	0	1	0	0	1	1	0	positive
0	1	0	1	1	0	1	1	0	0	negative
0	1	0	1	1	0	1	0	0	0	negative
0	0	1	0	1	1	0	1	1	1	positive
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	0	1	1	0	0	1	0	1	positive

test instance

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10	sentiment
1	0	1	0	1	0	0	1	1	0	positive

|||

Instance-Based Classification

Motivation

training data

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10	sentiment
1	0	1	0	1	0	0	1	1	0	positive
0	1	0	1	1	0	1	1	0	0	negative
0	1	0	1	1	0	1	0	0	0	negative
0	0	1	0	1	1	0	1	1	1	positive
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	0	1	1	0	0	1	0	1	positive

test instance

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10	sentiment
1	0	1	0	1	0	0	1	0	0	?

Instance-Based Classification

Motivation

training data

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10	sentiment
1	0	1	0	1	0	0	1	1	0	positive
0	1	0	1	1	0	1	1	0	0	negative
0	1	0	1	1	0	1	0	0	0	negative
0	0	1	0	1	1	0	1	1	1	positive
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	0	1	1	0	0	1	0	1	positive

test instance

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10	sentiment
1	0	1	0	1	0	0	1	1	0	positive

Typical Supervised Classification

training

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10	sentiment
1	0	1	0	1	0	0	1	1	0	positive
0	1	0	1	1	0	1	1	0	0	negative
0	1	0	1	1	0	1	0	0	0	negative
0	0	1	0	1	1	0	1	1	1	positive
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	0	1	1	0	0	1	0	1	positive

labeled examples

machine learning algorithm

model

testing

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10	sentiment
1	0	1	0	1	0	0	1	1	0	???

new, unlabeled example

model

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10	sentiment
1	0	1	0	1	0	0	1	1	0	positive

prediction

Instance-based Classification

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10	sentiment
1	0	1	0	1	0	0	1	1	0	positive
0	1	0	1	1	0	1	1	0	0	negative
0	1	0	1	1	0	1	0	0	0	negative
0	0	1	0	1	1	0	1	1	1	positive
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	0	1	1	0	0	1	0	1	positive

labeled examples

testing

instance-based algorithm

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10	sentiment
1	0	1	0	1	0	0	1	1	0	???

new, unlabeled example

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10	sentiment
1	0	1	0	1	0	0	1	1	0	positive

prediction

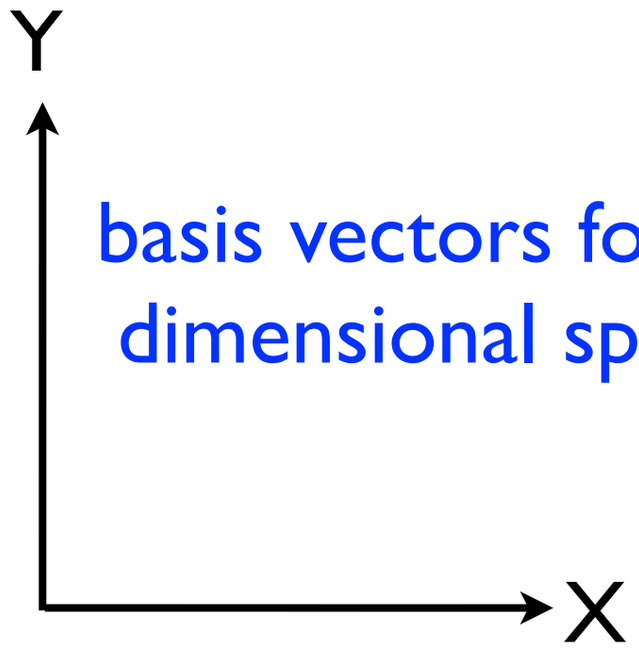
Instance-based Classification

- Assumption: instances with similar feature values should have the same target label
- Necessary Ingredients:
  - feature representation: a set of measures used to characterize each instance
  - distance metric: a measure of similarity between pairs of instances based on their feature values
  - an averaging technique: a way of combining the labels from the most similar training instances

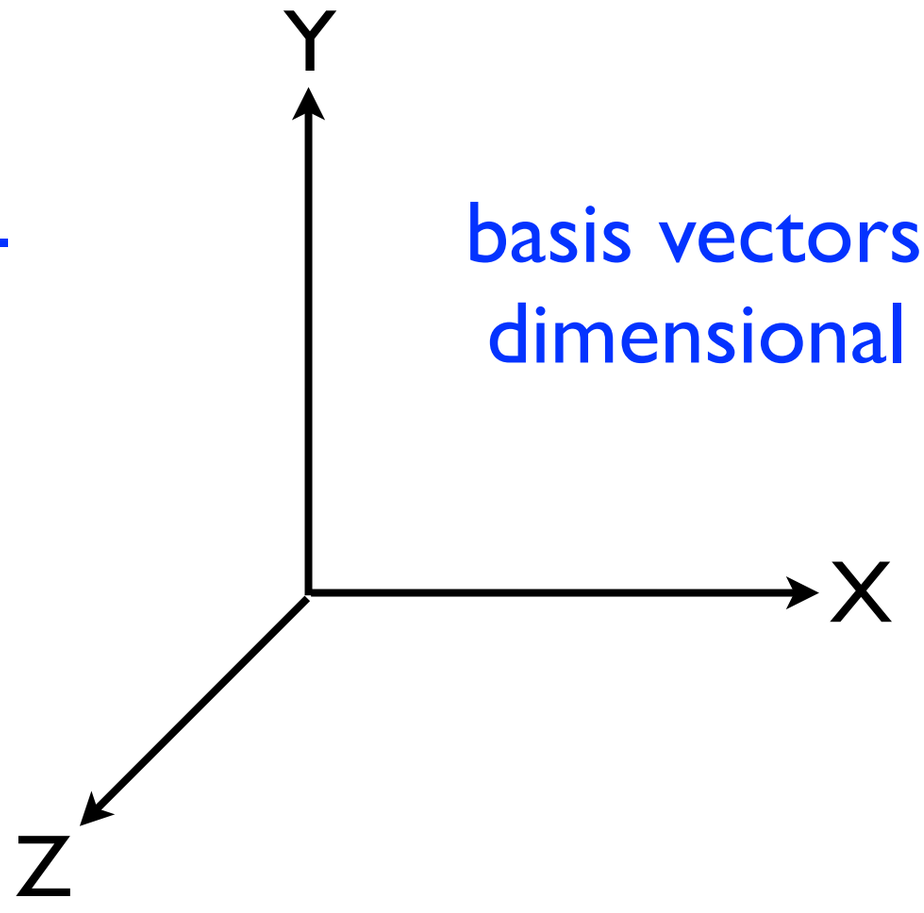
Vector Space

What is a Vector Space?

- Formally, a vector space is defined by a set of linearly independent basis vectors
- The basis vectors correspond to the dimensions or directions of the vector space



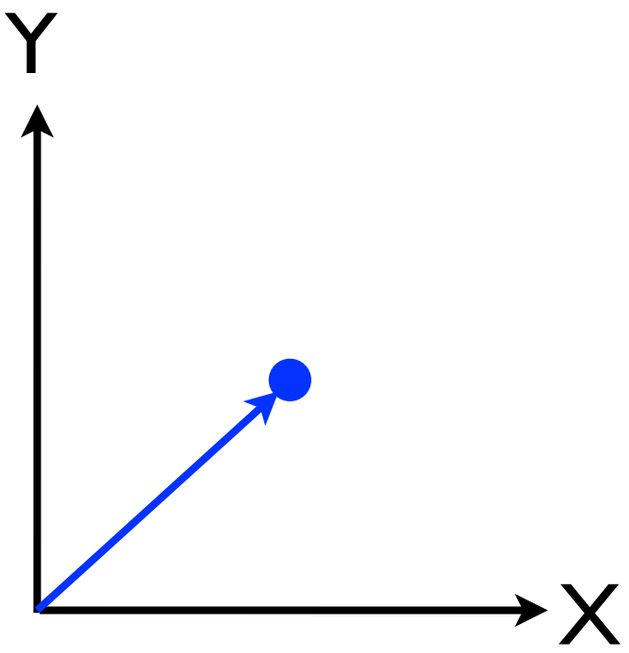
basis vectors for 2-dimensional space

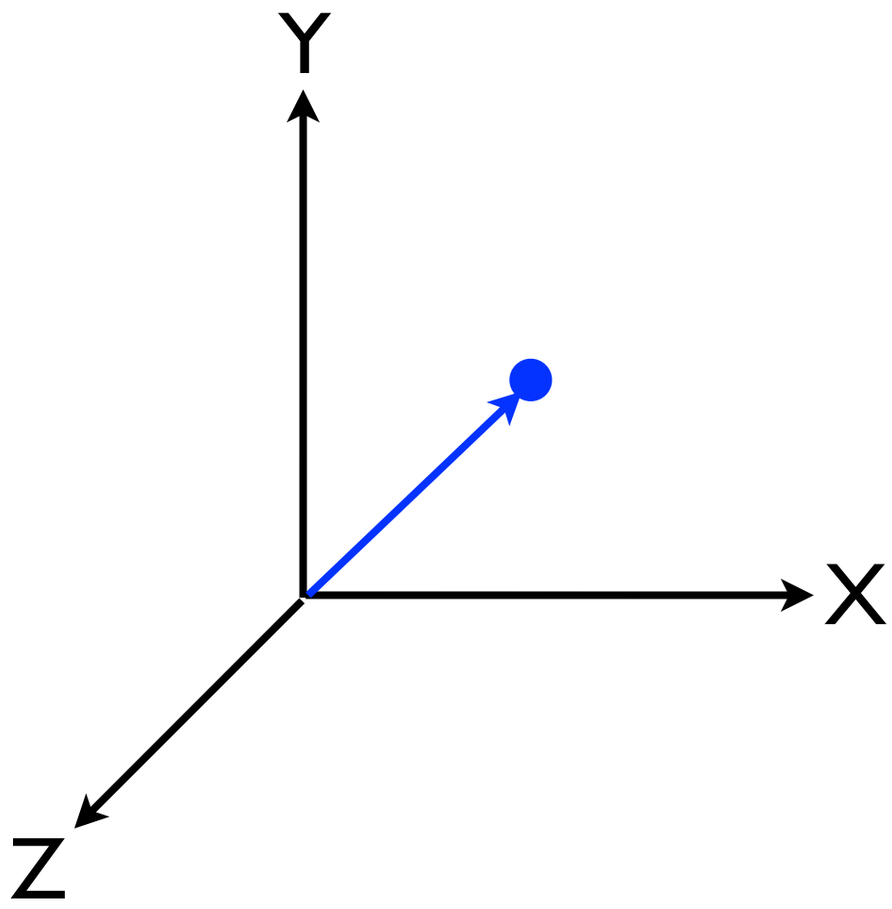


basis vectors for 3-dimensional space

What is a Vector?

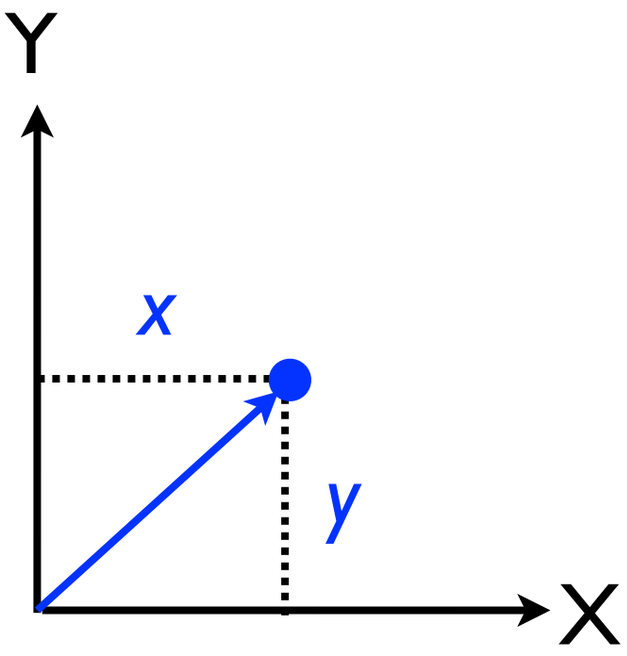
- A vector is a point in a vector space and has length (from the origin to the point) and direction

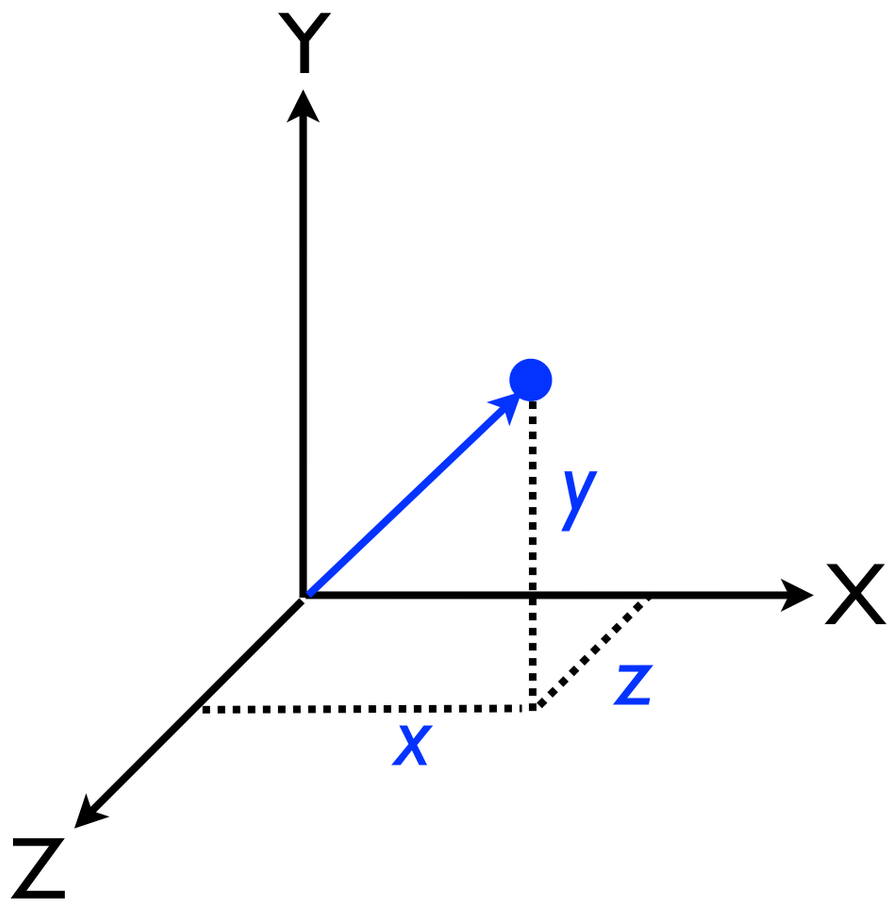




What is a Vector?

- A 2-dimensional vector can be written as [*x*,*y*]
- A 3-dimensional vector can be written as [*x*,*y*,*z*]





Binary Text Representation

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10	sentiment
1	0	1	0	1	0	0	1	1	0	positive
0	1	0	1	1	0	1	1	0	0	negative
0	1	0	1	1	0	1	0	0	0	negative
0	0	1	0	1	1	0	1	1	1	positive
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	0	1	1	0	0	1	0	1	positive

- Terms as features
- Bag of words representation: no word order
- 1 = the term appears in the text and 0 = the term does not appear in the text

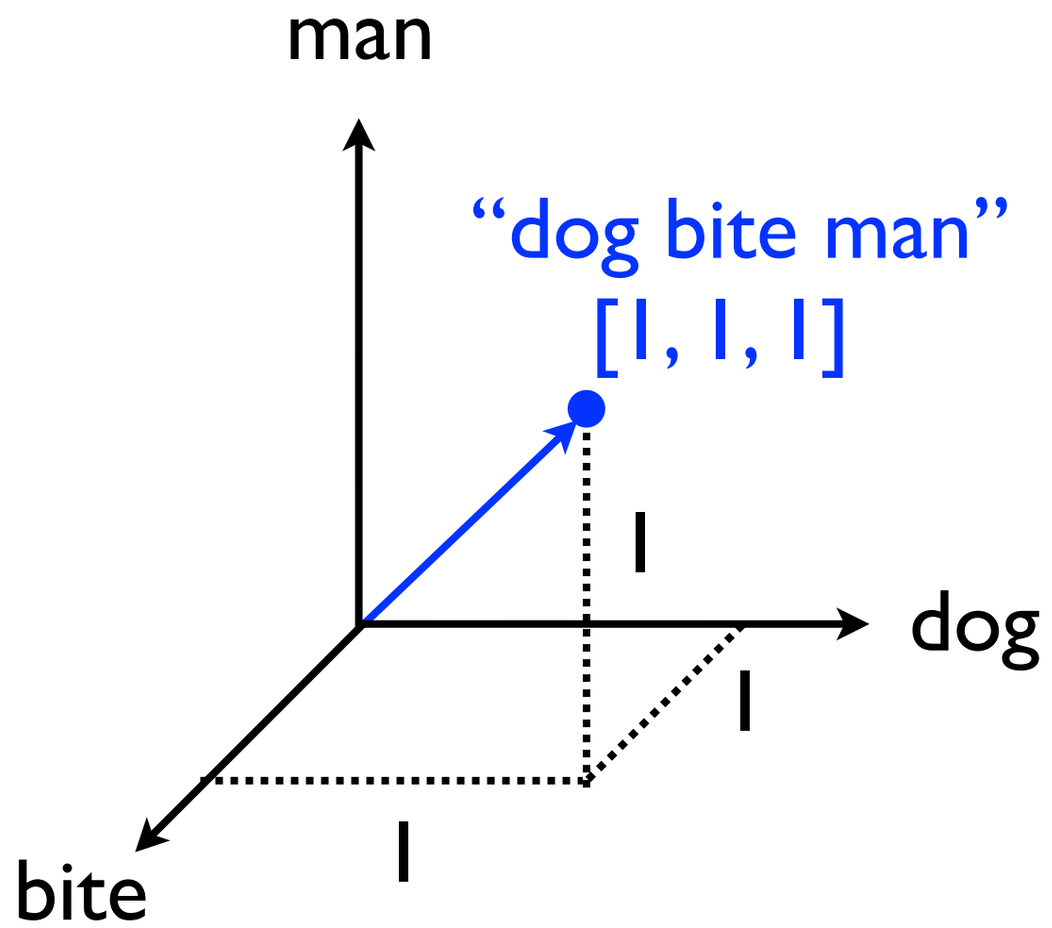
Vector Space Representation

- Let V denote the set of features in our feature representation
- Any arbitrary instance can be represented as a vector in |V|-dimensional space
- For simplicity, let's assume three features: dog, bite, man (i.e., |V| = 3)
- Why? Because it's easy to visualize 3-D space

Vector Space Representation
with binary weights

- 1 = the term appears at least once
- 0 = the term does not appear

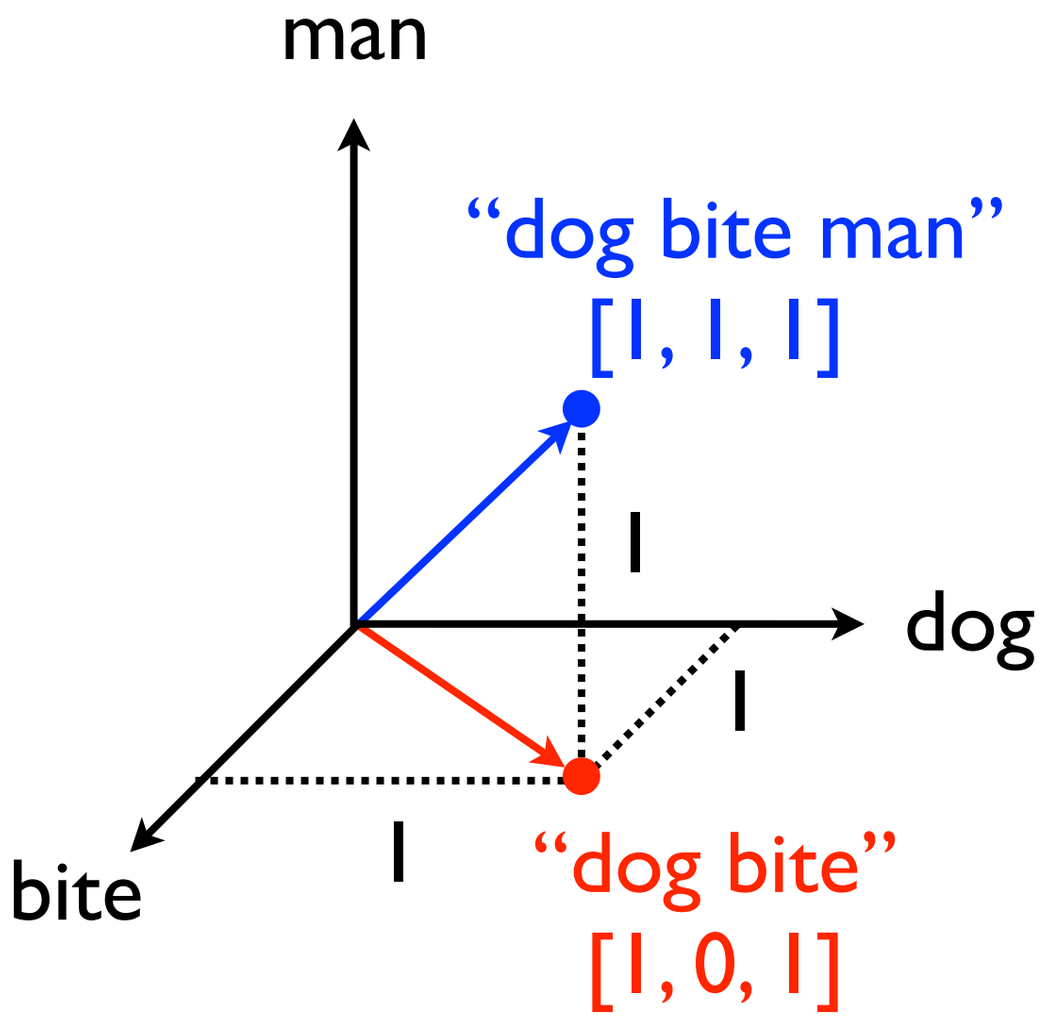
	dog	*man*	*bite*
i_1	1	1	1



Vector Space Representation
with binary weights

- 1 = the term appears at least once
- 0 = the term does not appear

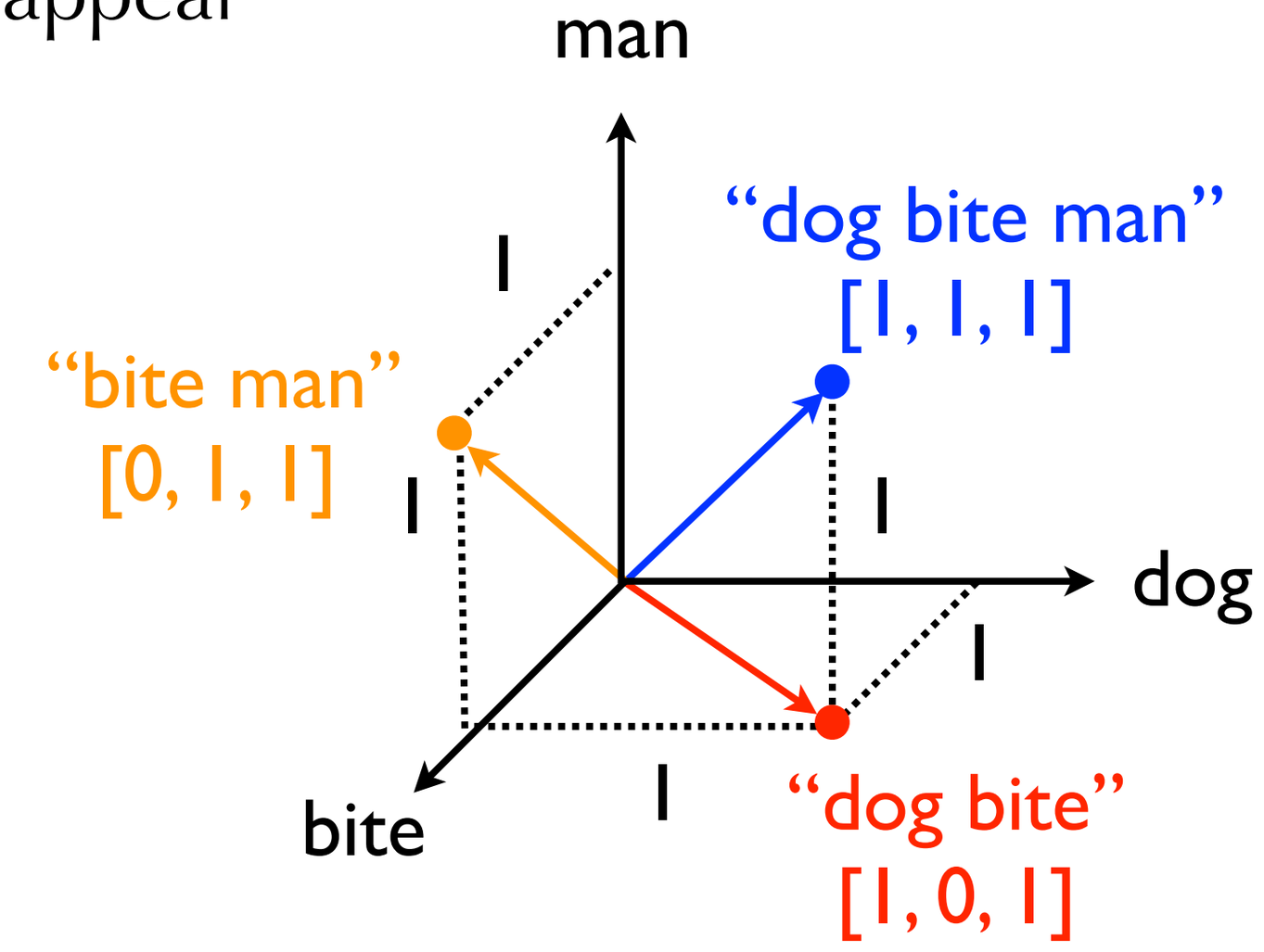
	dog	*man*	*bite*
i_1	1	1	1
i_2	1	0	1



Vector Space Representation
with binary weights

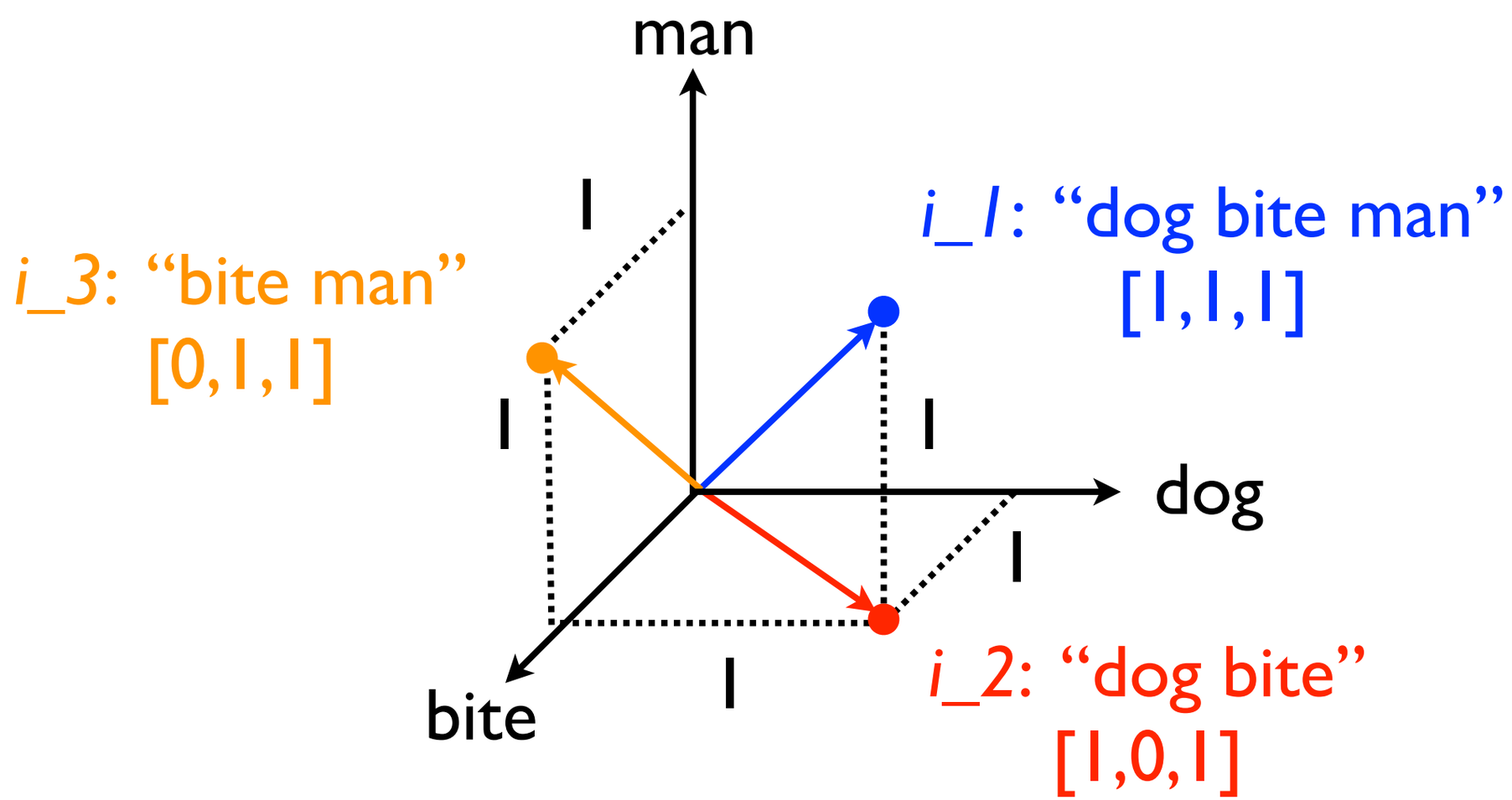
- 1 = the term appears at least once
- 0 = the term does not appear

	dog	*man*	*bite*
i_1	1	1	1
i_2	1	0	1
i_3	0	1	1



Vector Space Representation
with binary weights

- How can we use a vector-space representation to compute similarity or distance?



Vector Space Representation
with binary weights

- How can we use a vector-space representation to compute similarity or distance?
- Euclidean distance:

$$D(x,y) = \sqrt{\left(\sum_{i=1}^{|\mathcal{V}|}(x_i - y_i)^2\right)}$$

Euclidean Distance

	x	y	$(x_i - y_i)^2$
dog	1	1	0
bite	1	1	0
man	1	1	0
$D(x,y) = \sqrt{\left(\sum_{i=1}^{\lvert\mathcal{V}\rvert}(x_i - y_i)^2\right)}$			0

"dog bite man" vs. "dog bite man"

Euclidean Distance

	x	y	$(x_i - y_i)^2$
dog	1	1	0
bite	1	1	0
man	1	0	1
$D(x,y) = \sqrt{\left(\sum_{i=1}^{\lvert V \rvert}(x_i - y_i)^2\right)}$			1

"dog bite man" vs. "dog bite"

Euclidean Distance

	x	y	$(x_i - y_i)^2$
dog	1	0	1
bite	1	1	0
man	1	0	1
$D(x,y) = \sqrt{\left(\sum_{i=1}^{\lvert V \rvert}(x_i - y_i)^2\right)}$			**1.41**

"dog bite man" vs. "bite"

Binary Text Representation

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10	sentiment
1	0	1	0	1	0	0	1	1	0	positive
0	1	0	1	1	0	1	1	0	0	negative
0	1	0	1	1	0	1	0	0	0	negative
0	0	1	0	1	1	0	1	1	1	positive
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	0	1	1	0	0	1	0	1	positive

- Is this a good (bag of words) representation?
- Can we do better?

Term-Weighting

what are the most important terms?

- Movie: Rocky (1976)
- Plot:

Rocky Balboa is a struggling boxer trying to make the big time. Working in a meat factory in Philadelphia for a pittance, he also earns extra cash as a debt collector. When heavyweight champion Apollo Creed visits Philadelphia, his managers want to set up an exhibition match between Creed and a struggling boxer, touting the fight as a chance for a "nobody" to become a "somebody". The match is supposed to be easily won by Creed, but someone forgot to tell Rocky, who sees this as his only shot at the big time. Rocky Balboa is a small-time boxer who lives in an apartment in Philadelphia, Pennsylvania, and his career has so far not gotten off the canvas. Rocky earns a living by collecting debts for a loan shark named Gazzo, but Gazzo doesn't think Rocky has the viciousness it takes to beat up deadbeats. Rocky still boxes every once in a while to keep his boxing skills sharp, and his ex-trainer, Mickey, believes he could've made it to the top if he was willing to work for it. Rocky, goes to a pet store that sells pet supplies, and this is where he meets a young woman named Adrian, who is extremely shy, with no ability to talk to men. Rocky befriends her. Adrain later surprised Rocky with a dog from the pet shop that Rocky had befriended. Adrian's brother Paulie, who works for a meat packing company, is thrilled that someone has become interested in Adrian, and Adrian spends Thanksgiving with Rocky. Later, they go to Rocky's apartment, where Adrian explains that she has never been in a man's apartment before. Rocky sets her mind at ease, and they become lovers. Current world heavyweight boxing champion Apollo Creed comes up with the idea of giving an unknown a shot at the title. Apollo checks out the Philadelphia boxing scene, and chooses Rocky. Fight promoter Jergens gets things in gear, and Rocky starts training with Mickey. After a lot of training, Rocky is ready for the match, and he wants to prove that he can go the distance with Apollo. The 'Italian Stallion', Rocky Balboa, is an aspiring boxer in downtown Philadelphia. His one chance to make a better life for himself is through his boxing and Adrian, a girl who works in the local pet store. Through a publicity stunt, Rocky is set up to fight Apollo Creed, the current heavyweight champion who is already set to win. But Rocky really needs to triumph, against all the odds...

Term-Frequency

how important is a term?

rank	term	freq.	rank	term	freq.
1	a	22	16	creed	5
2	rocky	19	17	philadelphia	5
3	to	18	18	has	4
4	the	17	19	pet	4
5	is	11	20	boxing	4
6	and	10	21	up	4
7	in	10	22	an	4
8	for	7	23	boxer	4
9	his	7	24	s	3
10	he	6	25	balboa	3
11	adrian	6	26	it	3
12	with	6	27	heavyweigh	3
13	who	6	28	champion	3
14	that	5	29	fight	3
15	apollo	5	30	become	3

Term-Frequency

how important is a term?

rank	term	freq.	rank	term	freq.
1	a	22	16	creed	5
2	rocky	19	17	philadelphia	5
3	to	18	18	has	4
4	the	17	19	pet	4
5	is	11	20	boxing	4
6	and	10	21	up	4
7	in	10	22	an	4
8	for	7	23	boxer	4
9	his	7	24	s	3
10	he	6	25	balboa	3
11	adrian	6	26	it	3
12	with	6	27	heavyweigh	3
13	who	6	28	champion	3
14	that	5	29	fight	3
15	apollo	5	30	become	3

Inverse Document Frequency (IDF)

how important is a term?

$$idf_t = \log(\frac{N}{df_t})$$

- N = number of training set instances
- df_t = number of training set instances where term t appears

Inverse Document Frequency (IDF)

how important is a term?

rank	term	idf	rank	term	idf
1	doesn	11.66	16	creed	6.84
2	adrain	10.96	17	paulie	6.82
3	viciousness	9.95	18	packing	6.81
4	deadbeats	9.86	19	boxes	6.75
5	touting	9.64	20	forgot	6.72
6	jergens	9.35	21	ease	6.53
7	gazzo	9.21	22	thanksgivin	6.52
8	pittance	9.05	23	earns	6.51
9	balboa	8.61	24	pennsylvani	6.50
10	heavyweigh	7.18	25	promoter	6.43
11	stallion	7.17	26	befriended	6.38
12	canvas	7.10	27	exhibition	6.31
13	ve	6.96	28	collecting	6.23
14	managers	6.88	29	philadelphia	6.19
15	apollo	6.84	30	gear	6.18

TF.IDF

how important is a term?

$$tf_t \times idf_t$$

greater when the term is frequent in the instance

greater when the term is rare in the training set

TF.IDF

how important is a term?

rank	term	tf.idf	rank	term	tf.idf
1	rocky	96.72	16	meat	11.76
2	apollo	34.20	17	doesn	11.66
3	creed	34.18	18	adrain	10.96
4	philadelphia	30.95	19	fight	10.02
5	adrian	26.44	20	viciousness	9.95
6	balboa	25.83	21	deadbeats	9.86
7	boxing	22.37	22	touting	9.64
8	boxer	22.19	23	current	9.57
9	heavyweigh	21.54	24	jergens	9.35
10	pet	21.17	25	s	9.29
11	gazzo	18.43	26	struggling	9.21
12	champion	15.08	27	training	9.17
13	match	13.96	28	pittance	9.05
14	earns	13.01	29	become	8.96
15	apartment	11.82	30	mickey	8.96

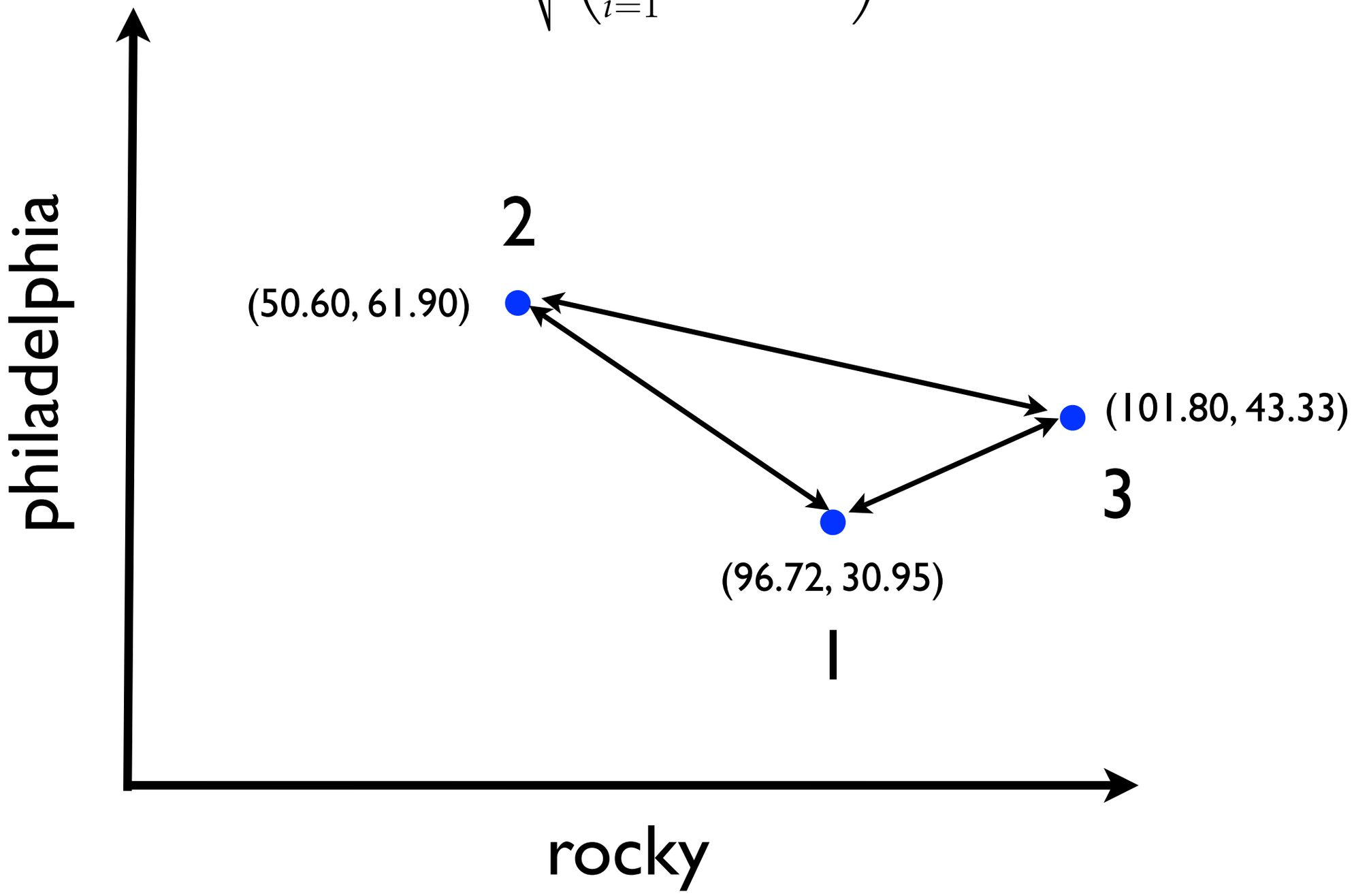
Calculating TF.IDF Weights

$$tf_t \times log\left(\frac{N}{df_t}\right)$$

term	tf	N	df	idf	tf.idf
rocky	19	230721	1420	5.09	96.72
philadelphia	5	230721	473	6.19	30.95
boxer	4	230721	900	5.55	22.19
fight	3	230721	8170	3.34	10.02
mickey	2	230721	2621	4.48	8.96
for	7	230721	117137	0.68	4.75

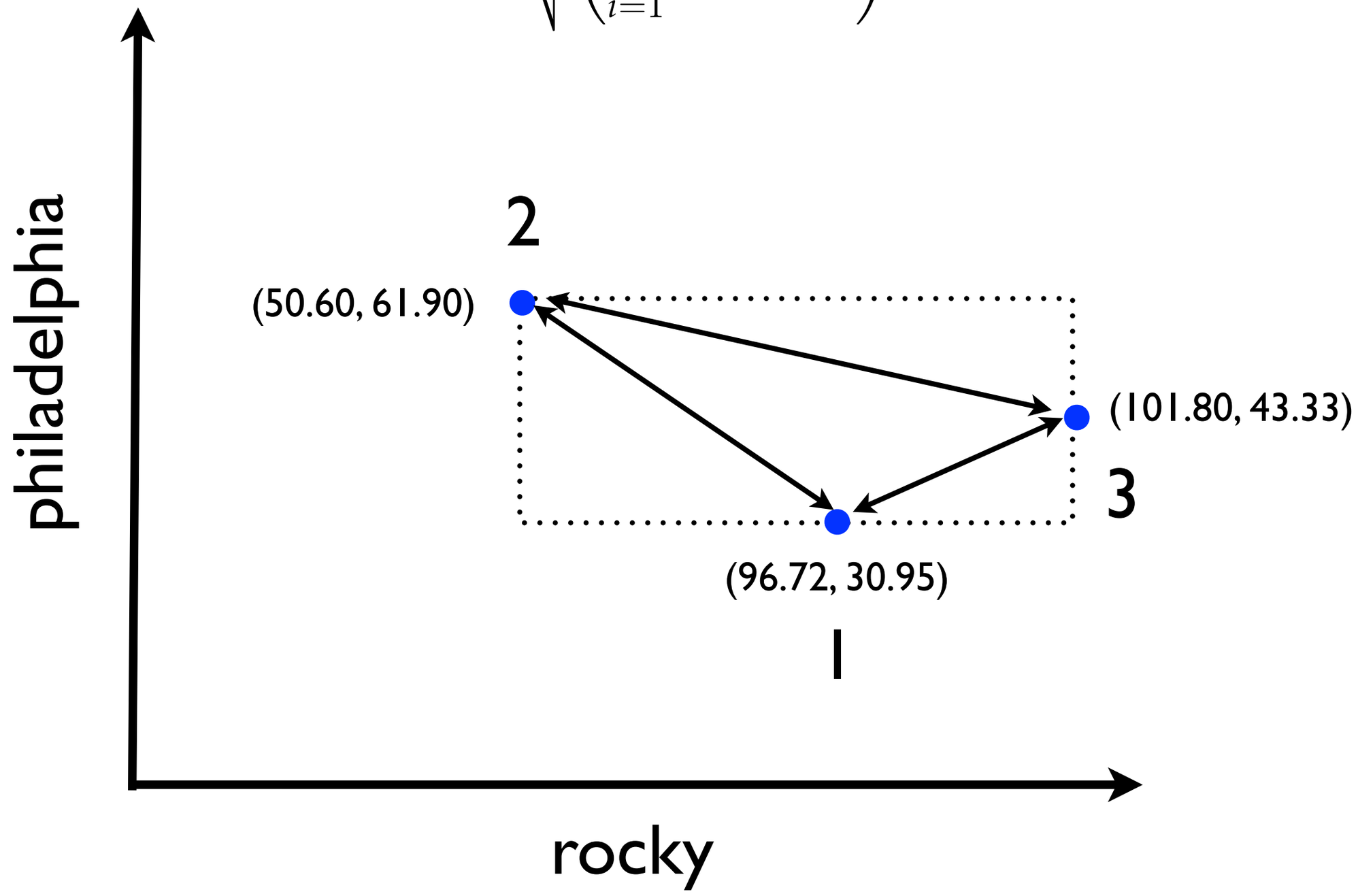
Putting Everything Together

$$D(x,y) = \sqrt{\left(\sum_{i=1}^{|\mathcal{V}|}(x_i - y_i)^2\right)}$$

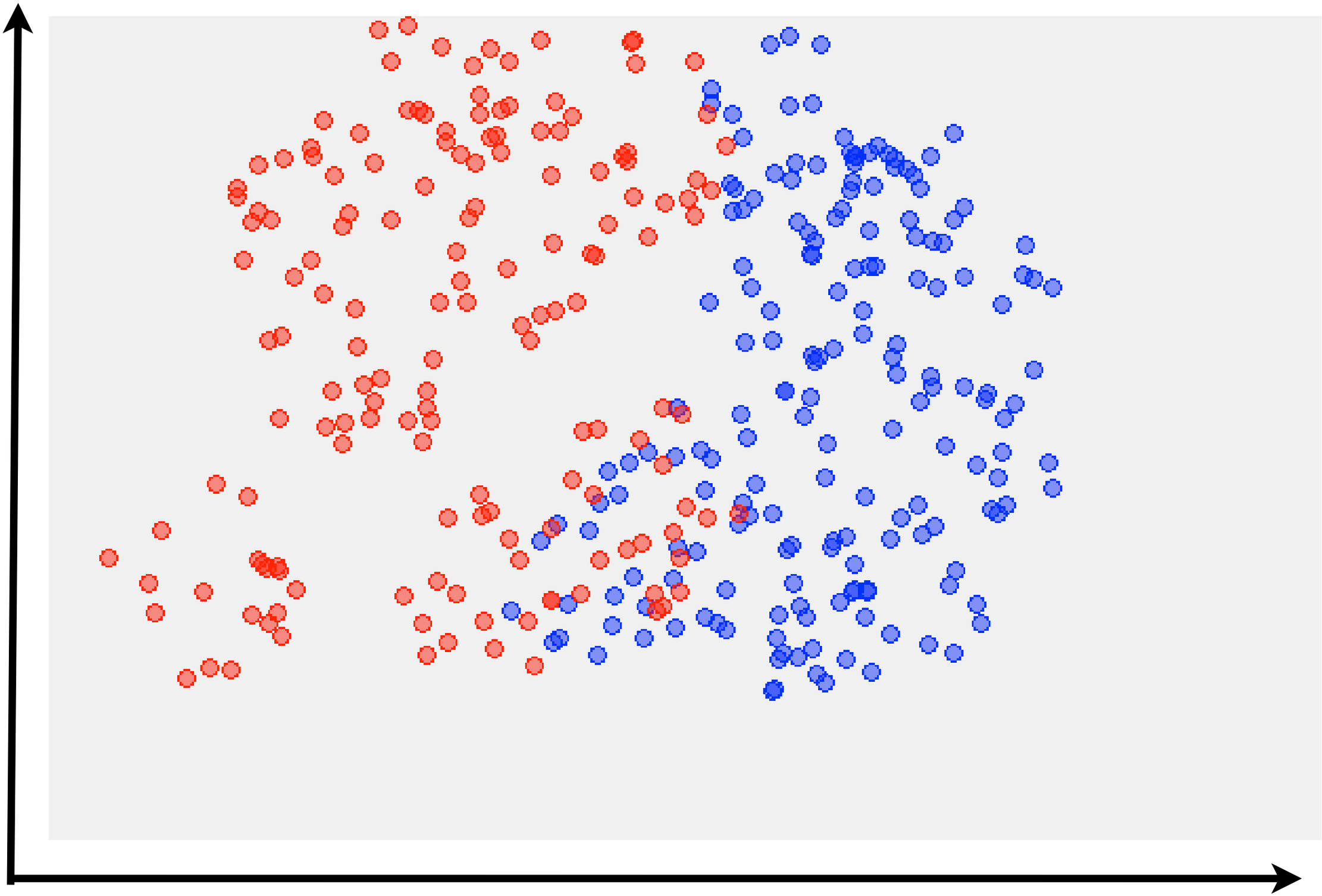


Putting Everything Together

$$D(x,y) = \sqrt{\left(\sum_{i=1}^{|\mathcal{V}|}(x_i - y_i)^2\right)}$$

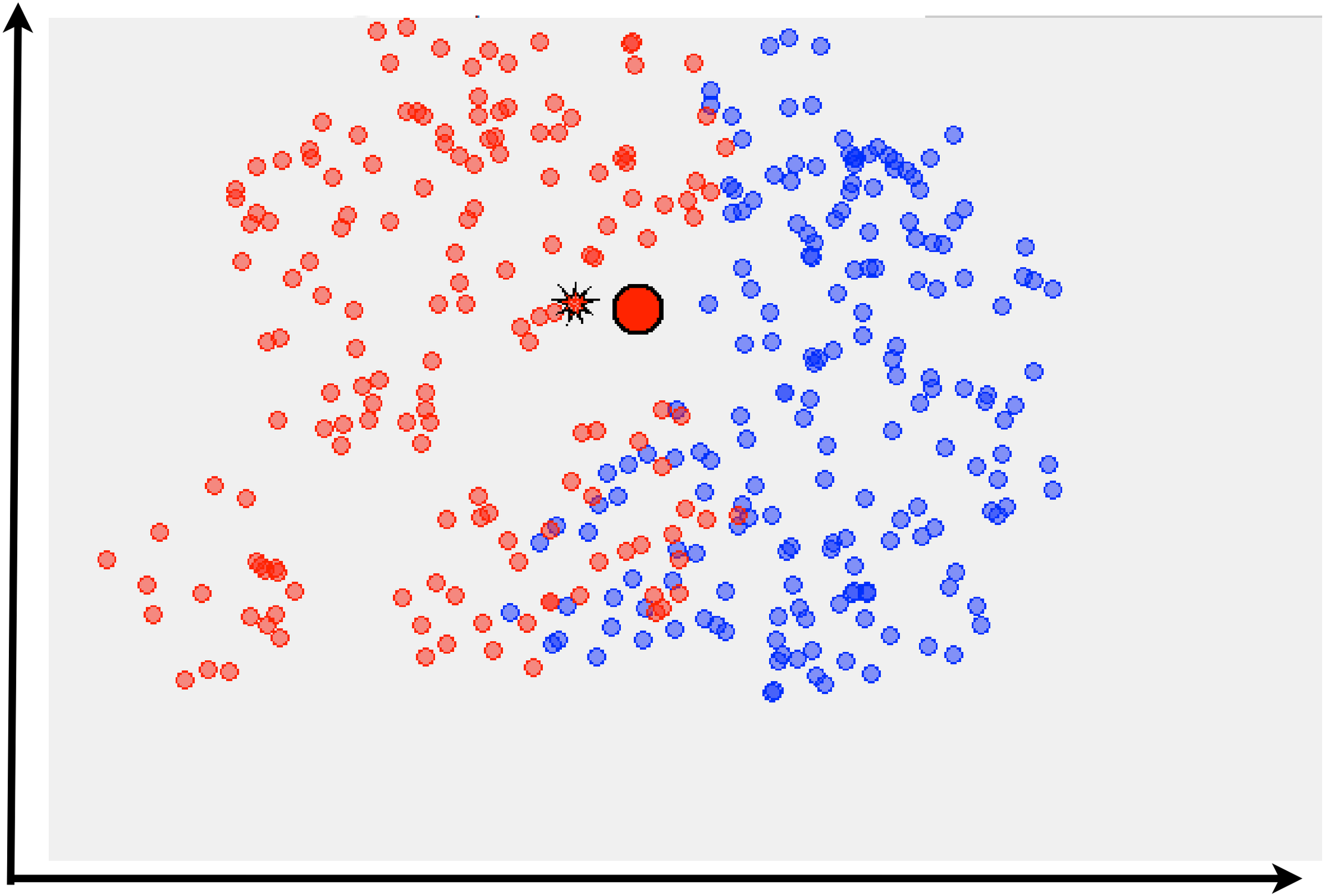


Nearest-Neighbor Classification

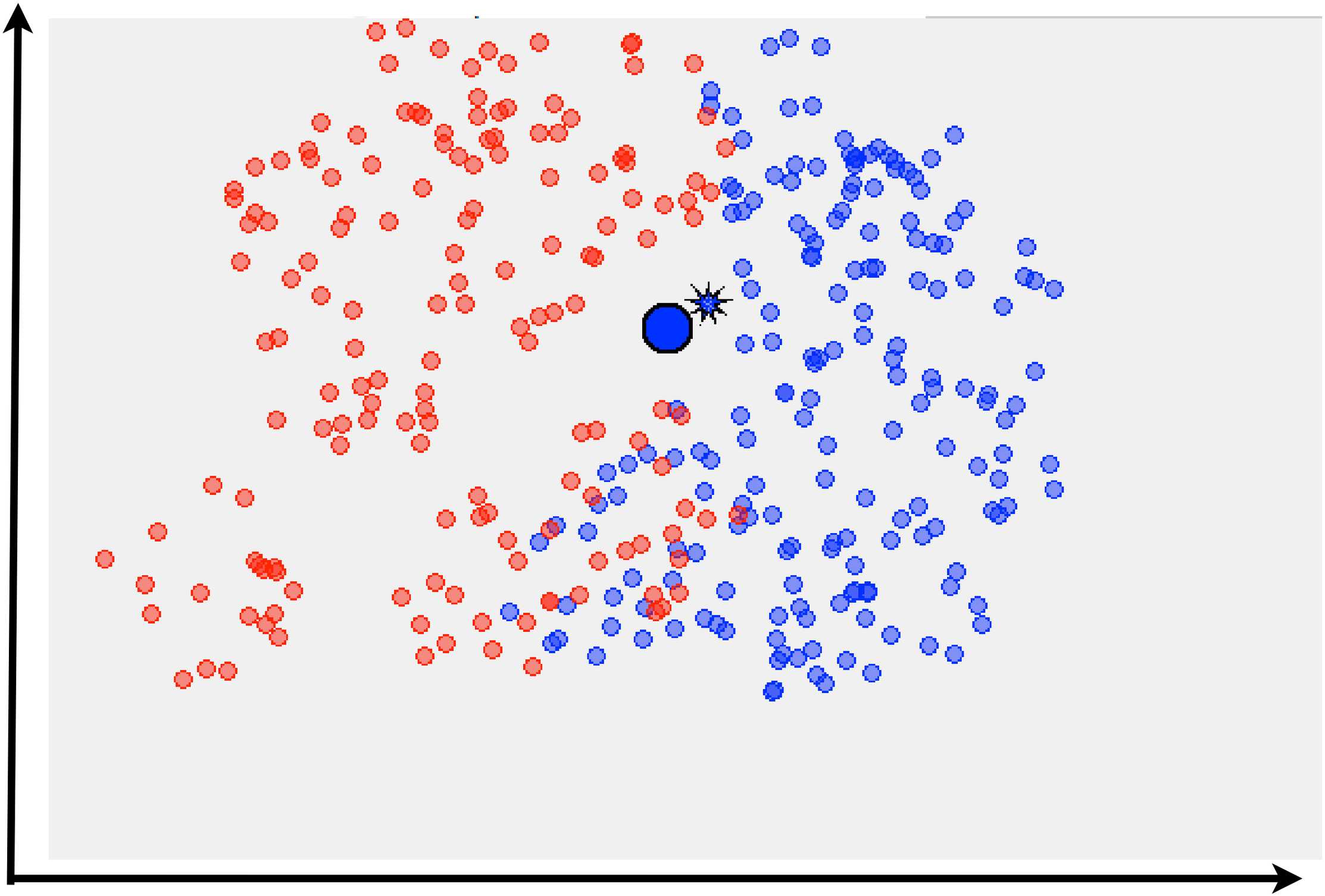


source: http://www.math.le.ac.uk/people/ag153/homepage/KNN/

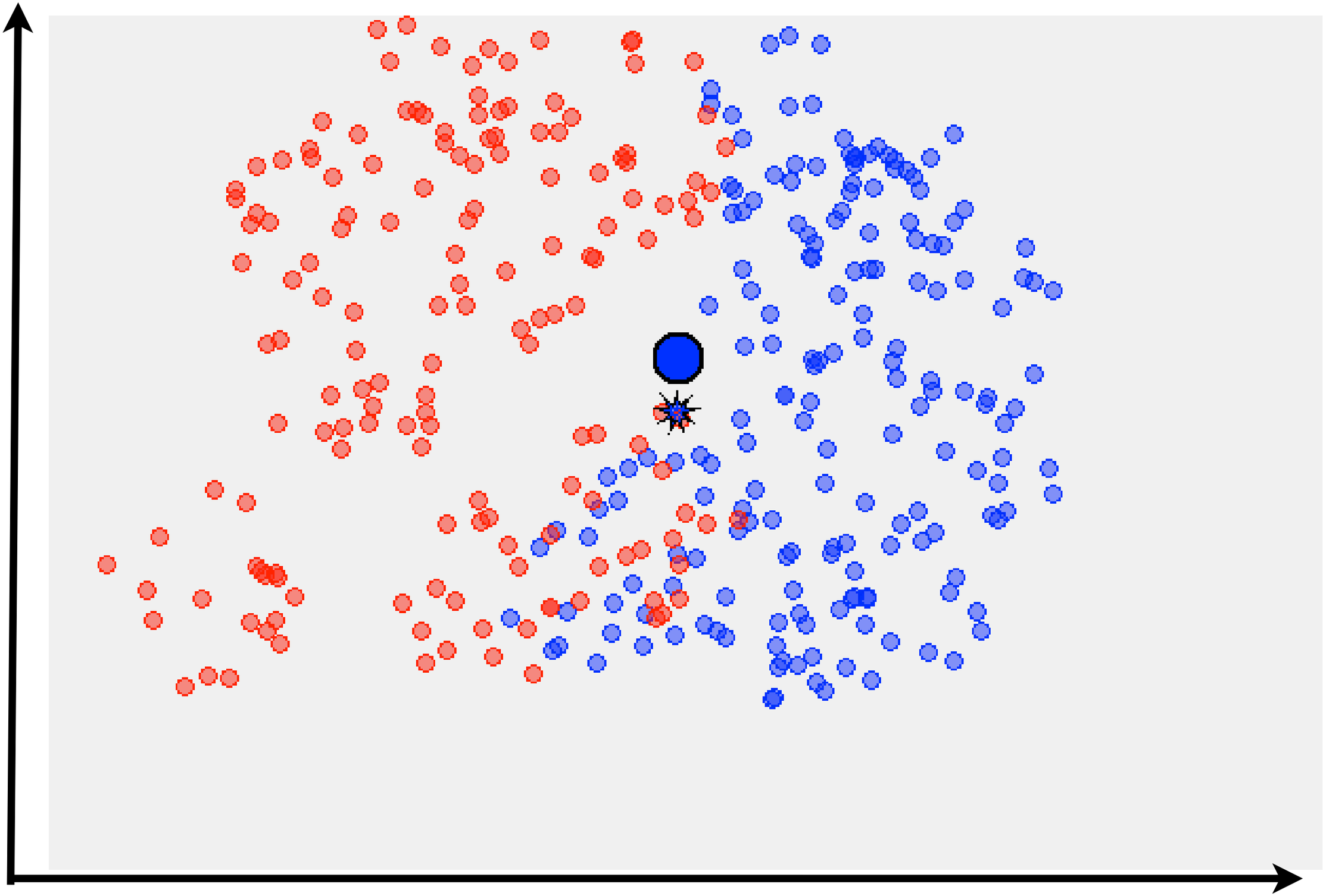
Nearest-Neighbor Classification



Nearest-Neighbor Classification



Nearest-Neighbor Classification



Nearest-Neighbor Classification

- Given a test instance, assign the label associated with the nearest training set instance
- What are a potential limitation of this approach?

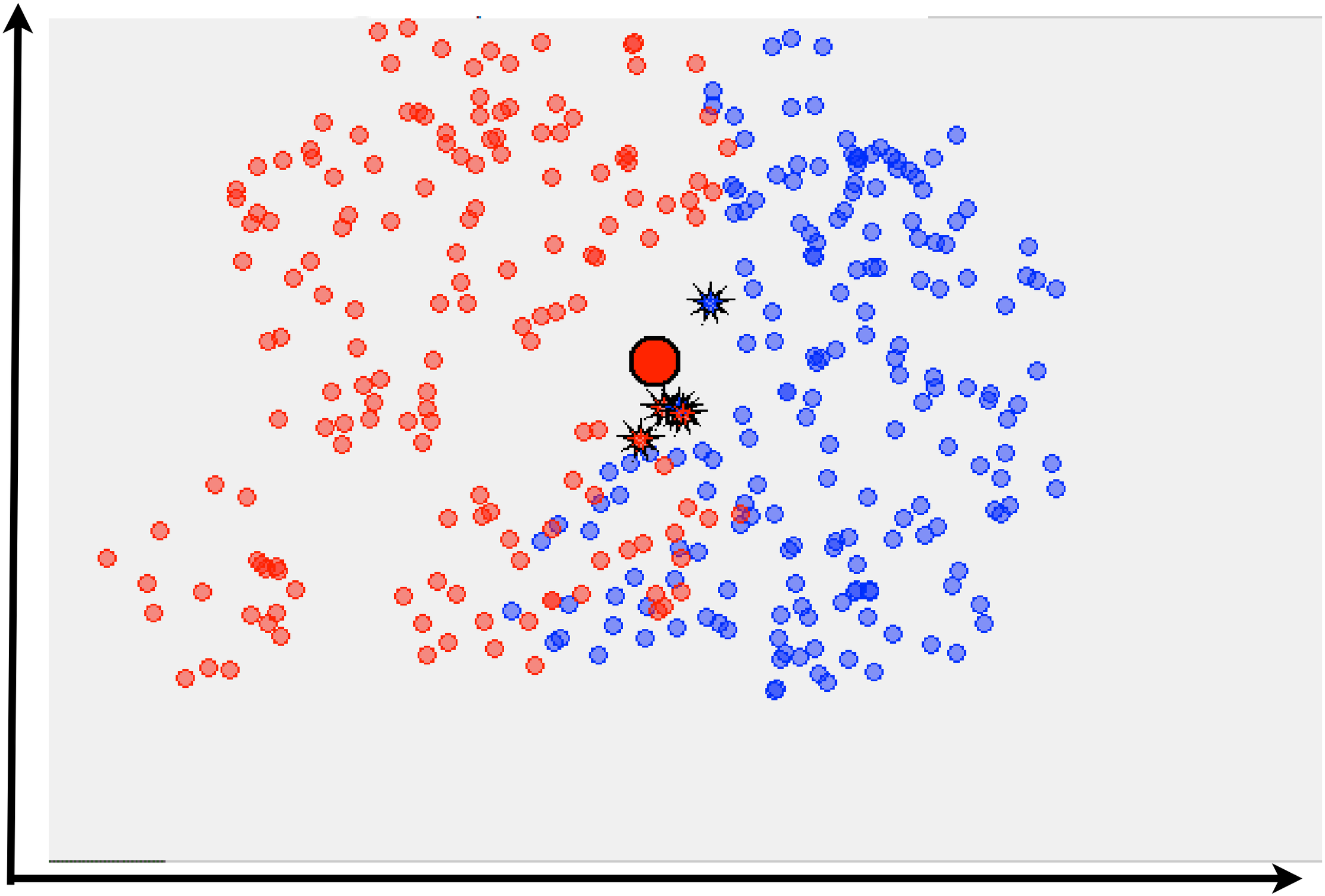
Nearest-Neighbor Classification

- Given a test instance, assign the label associated with the nearest training set instance
- What are a potential limitation of this approach?
- The nearest neighbor may be an outlier

- For example: a positive movie review with lots of negative words
- Solution: use the majority class associated with the K nearest neighbors

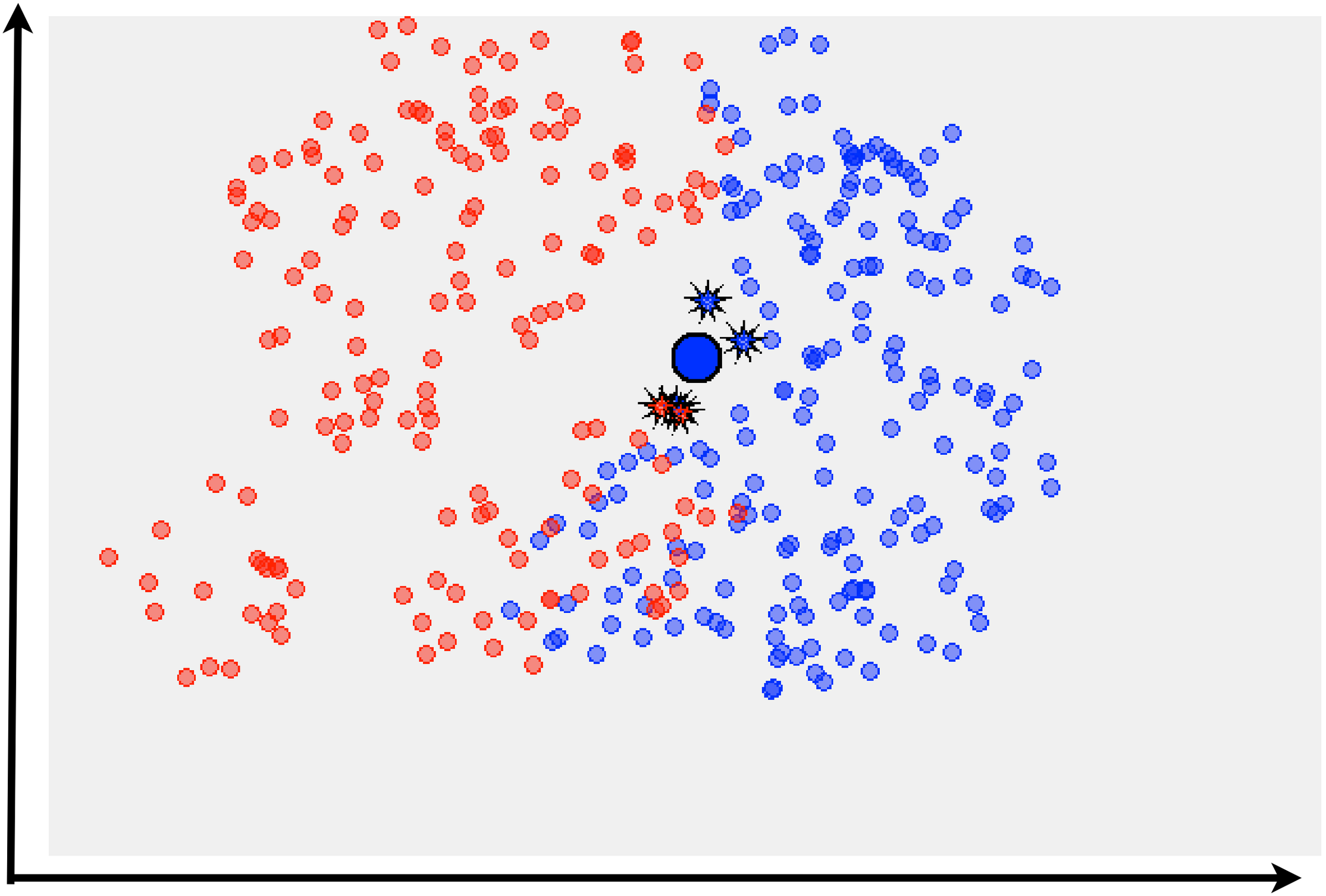
K Nearest-Neighbor (KNN)

(K = 5)



K Nearest-Neighbor (KNN)

(K = 5)



K Nearest-Neighbor Classification

- Given a test instance, assign the majority label associated with the K nearest training set instances
- What are a potential limitation of this approach?

K Nearest-Neighbor Classification

- Given a test instance, assign the majority label associated with the K nearest training set instances
- What are a potential limitation of this approach?
- Nearest-neighbors that are far away have the same influence as nearest-neighbors that are close
- Solution: use some kind of weighted voting
- There are many, many variants
- Including one that does weighted voting using the entire training set

Topic Detection And Tracking (TDT)

- Objective: monitor stream of text and identify documents about a news story of interest

Obama to Public Schools: Allow Transgender Students Access to ...
ABC News - 12 hours ago
Editor's note: An earlier version of this story characterized the letter sent to school by the **Obama** administration as an order; this has been ...

The **Obama** Administration Is Warning Schools Over Transgender ...
TIME - 48 minutes ago

Obama Decrees ALL Public Schools Must Allow Transgender ...
Daily Caller - 13 hours ago

Obama administration to issue guidance on transgender access to ...
In-Depth - CNN - 6 hours ago

Obama's gender warning: The fundamental transform of our country ...
Opinion - Fox News - 1 hour ago

US Gives Sweeping Guidance to Schools on Transgender Students
In-Depth - NBCNews.com - 42 minutes ago

TIME


Daily Caller

Fox News

Wall Street Jo...


FOX 4 News


New York Post

Explore in depth (431 more articles)

K Nearest-Neighbor TDT (1)

$$P(yes|d) = \sum_{d' \in P(d,k)} sim(d,d') - \sum_{d' \in N(d,k)} sim(d,d')$$

k-NN from positive set

k-NN from negative set

K Nearest-Neighbor TDT (2)

- Treat positive and negative instances differently

$$P(yes|d) = \frac{1}{k1} \sum_{d' \in P(d,k1)} sim(d,d') - \frac{1}{k2} \sum_{d' \in N(d,k2)} sim(d,d')$$

k-NN from positive set

k-NN from negative set

Outline: Predictive and Exploratory Analysis

Concepts, Instances, and Features

Human Annotation

Text Representation

Learning Algorithms

Evaluation metrics

Experimentation

Clustering

Hands-on Exercise

Evaluation

- Predictive analysis: training a model to make predictions on previously unseen data
- Evaluation: using previously unseen labeled data to estimate the quality of a model's predictions on new data
- Evaluation Metric: a measure that summarizes the quality of a model's predictions

Predictive Analysis

training

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10	sentiment
1	0	1	0	1	0	0	1	1	0	positive
0	1	0	1	1	0	1	1	0	0	negative
0	1	0	1	1	0	1	0	0	0	negative
0	0	1	0	1	1	0	1	1	1	positive
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	0	1	1	0	0	1	0	1	positive

labeled examples

machine learning algorithm

model

testing

model

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10	sentiment
1	0	1	0	1	0	0	1	1	0	positive
0	1	0	1	1	0	1	1	0	0	negative
0	1	0	1	1	0	1	0	0	0	negative
0	0	1	0	1	1	0	1	1	1	positive
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	0	1	1	0	0	1	0	1	positive

new, labeled examples

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10	sentiment
1	0	1	0	1	0	0	1	1	0	positive
0	1	0	1	1	0	1	1	0	0	negative
0	1	0	1	1	0	1	0	0	0	negative
0	0	1	0	1	1	0	1	1	1	positive
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	0	1	1	0	0	1	0	1	positive

predictions

Evaluation Metrics

- There are many different metrics
- Different metrics make different assumptions about what end users care about
- Choosing the most appropriate metric is important!

Evaluation Metrics

(1) accuracy

- Accuracy: percentage of correct predictions

		true: pos	true: neg
predicted	pos	a	b
	neg	c	d

$$\mathcal{A} = \frac{(a+d)}{(a+b+c+d)}$$

Evaluation Metrics

(1) accuracy

- Accuracy: percentage of correct predictions

predicted \ true	pos	neut.	neg
pos	a	b	c
neut.	d	e	f
neg	g	h	i

$$\mathcal{A} = \frac{(a+e+i)}{(a+b+c+d+e+f+g+h+i)}$$

Evaluation Metrics

(1) accuracy

- What assumption(s) does accuracy make?

		true		
		pos	neut.	neg
predicted	pos	a	b	c
	neut.	d	e	f
	neg	g	h	i

$$\mathcal{A} = \frac{(a+e+i)}{(a+b+c+d+e+f+g+h+i)}$$

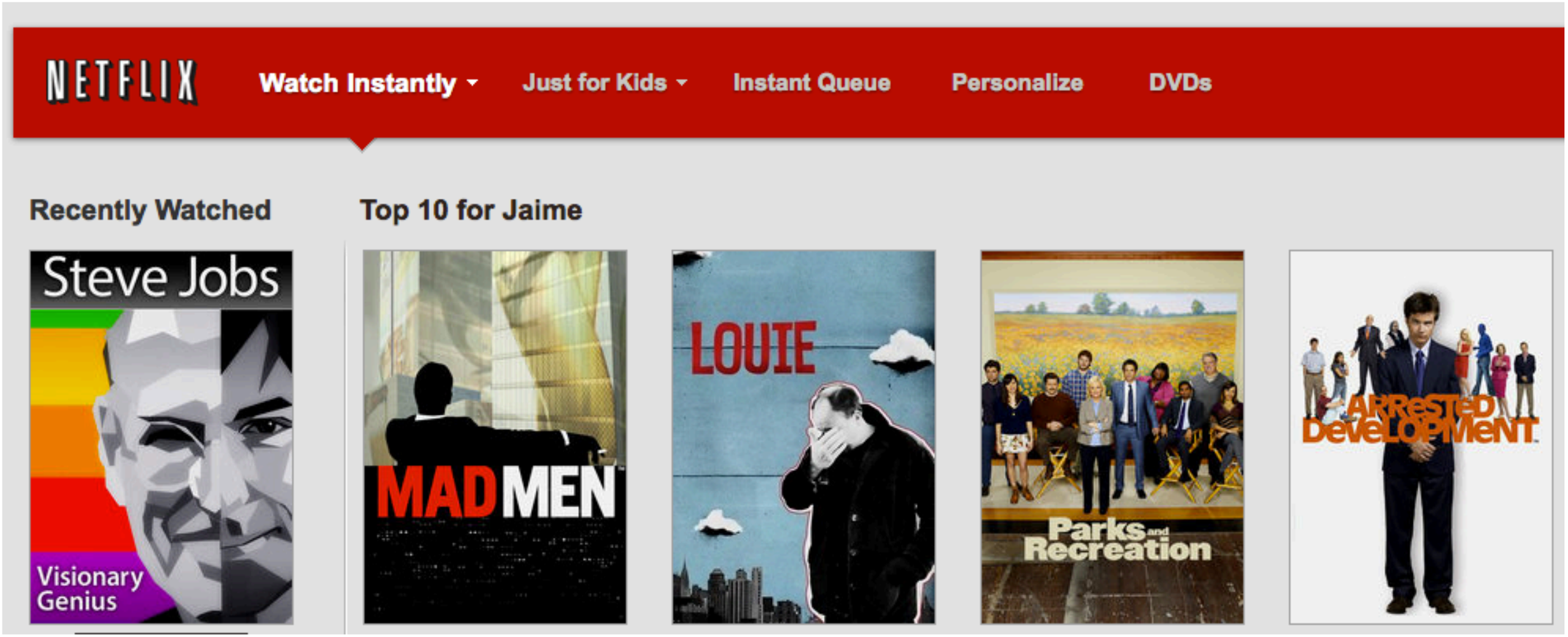
Evaluation Metrics

(1) accuracy

- What assumption(s) does accuracy make?
- It assumes that all prediction errors are equally bad
- Oftentimes, we care more about one class than the others
- If so, the class of interest is usually the minority class
- We are looking for the “needles in the haystack”
- In this case, accuracy is not a good evaluation metric
- There are metrics that provide more insight into per-class performance

Evaluation Metrics

- Content recommendation: relevant vs. non-relevant



Evaluation Metrics

- Email spam filtering: spam vs. ham

From	Subject	Date Received	Categories
SUNDAY			
audio@DesktopTrainingOnline.com	Adobe Acrobat Pro: Instructor-Led Training t...	Sun 9/30/12 5:19 PM	Junk
THURSDAY			
ei-sci@ei-sci.org	SCI-EI期刊检索、收录 （ICIEEE 2013）邀请函	Thu 9/27/12 2:50 AM	Junk
WEDNESDAY			
The New York Times	Act now to receive FREE digital access PLUS 5...	Wed 9/26/12 3:49 PM	Junk
Citrix Systems	Give people the freedom to work anyplace	Wed 9/26/12 1:20 PM	Junk
LAST WEEK			
audio@DesktopTrainingOnline.com	Excel 2007/2010 Formatting & Customizing...	Mon 9/24/12 8:24 PM	Junk
Vonage	Last Chance: Unlimited calls with Vonage Basi...	Mon 9/24/12 2:56 PM	Junk
conference EDM	World's Tallest Tower in Tokyo - Join 2013 E...	Thu 9/20/12 10:48 PM	Junk
2 WEEKS AGO			
Jim Davidson & Strategic Investment	Washington Insider Comes out of the Shadow...	Tue 9/18/12 12:02 PM	Junk
audio@supertrainme.com	Student Record Retention: Secure Data, Maint...	Tue 9/18/12 6:56 AM	Junk
audio@DesktopTrainingOnline.com	Mastering Excel 2007/2010 Charts: Tips & Tri...	Thu 9/13/12 8:31 PM	Junk
3 WEEKS AGO			
Vonage	Get Unlimited Calling with Vonage Basic Talk...	Fri 9/7/12 2:41 PM	Junk
prof_qian	[EI SCOPUS ISI Journal, Beijing, China]Internati...	Fri 9/7/12 1:32 PM	Junk

Evaluation Metrics

- Product reviews: positive vs. negative vs. neutral

Evaluation Metrics

- Text-based Forecasting: buy vs. sell vs. hold

Evaluation Metrics

- Health monitoring system: alarm vs. no alarm

Evaluation Metrics

(1) accuracy

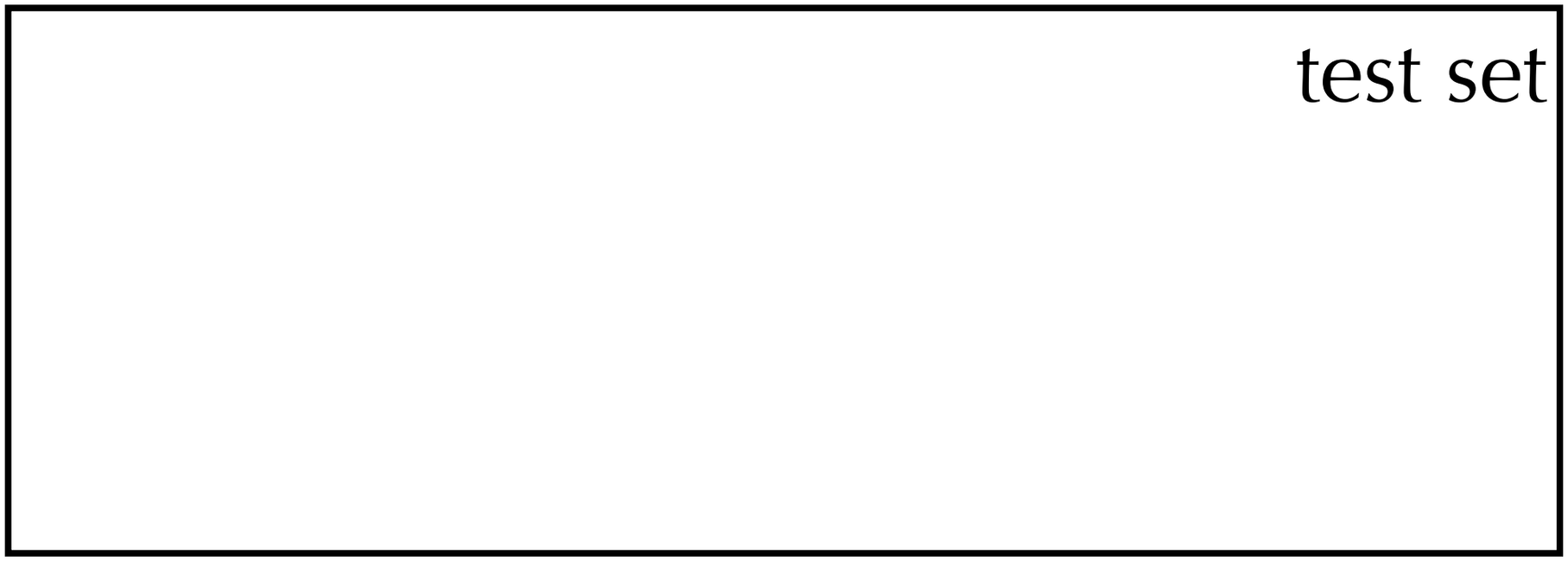
- What assumption(s) does accuracy make?
- It assumes that all prediction errors are equally bad
- Oftentimes, we care more about one class than the others
- If so, the class of interest is usually the minority class
- We are looking for the “needles in the haystack”
- In this case, accuracy is not a good evaluation metric
- There are metrics that provide more insight into per-class performance

Evaluation Metrics
(2) precision and (3) recall

- For a given class **C**:
  - precision: the percentage of positive predictions that are truly positive
  - recall: the percentage of true positives that are correctly predicted positive

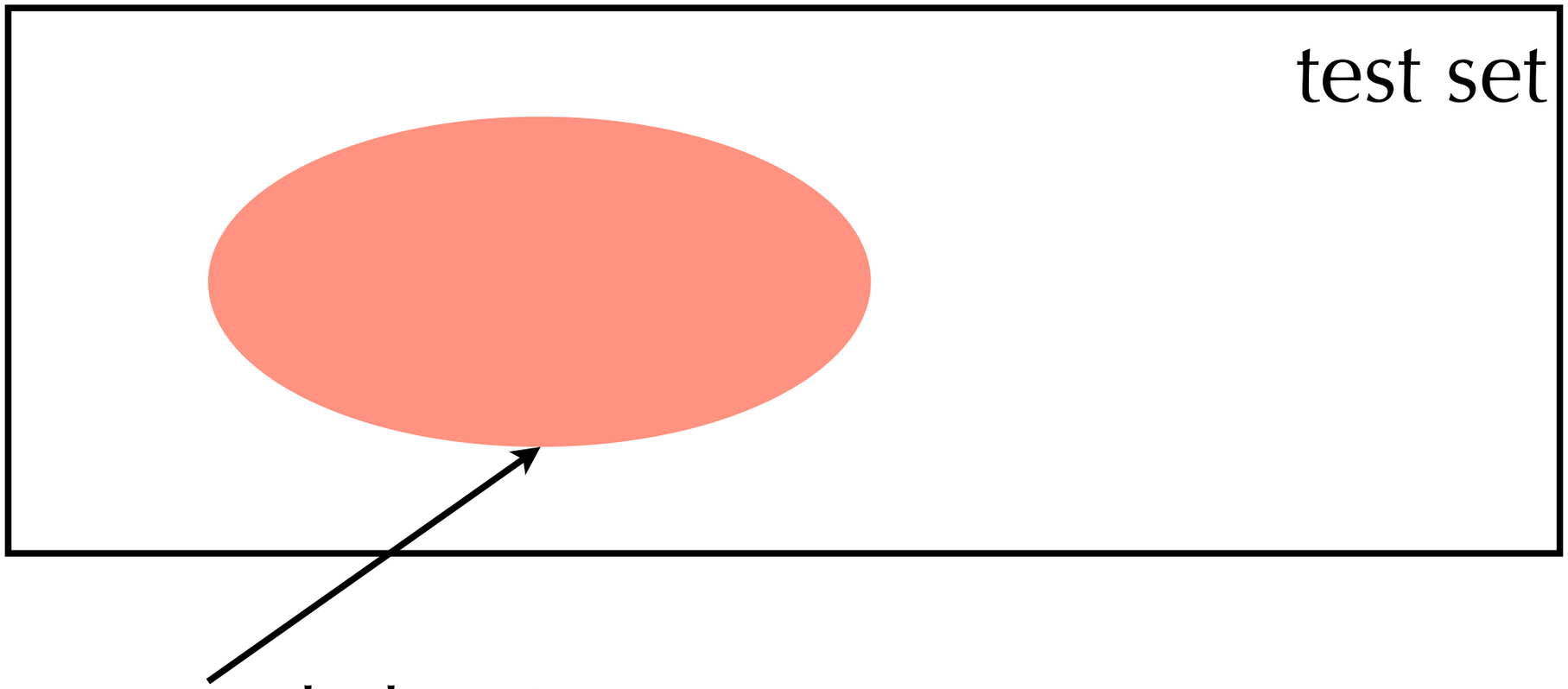
Evaluation Metrics

(2) precision and (3) recall



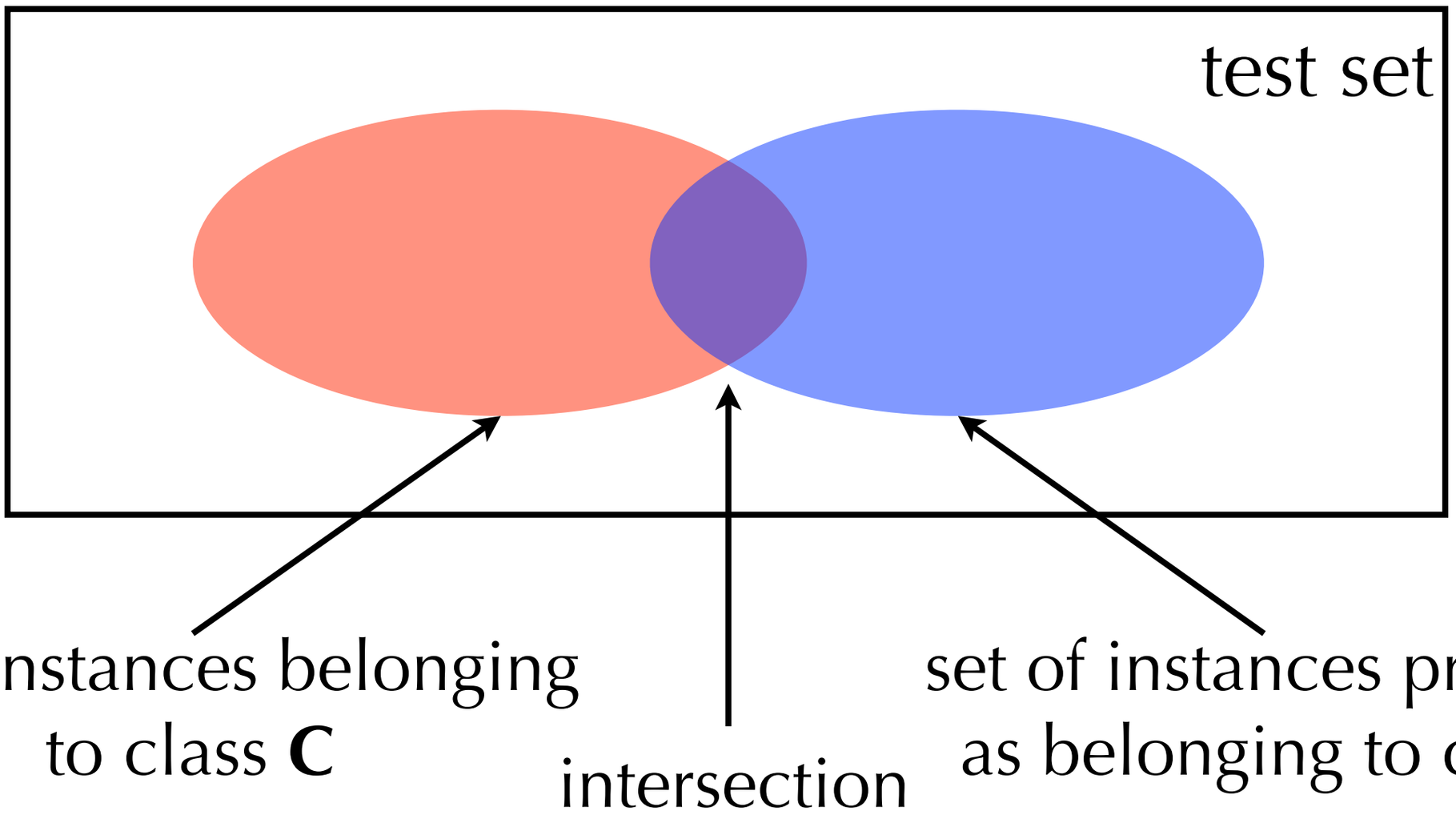
Evaluation Metrics

(2) precision and (3) recall



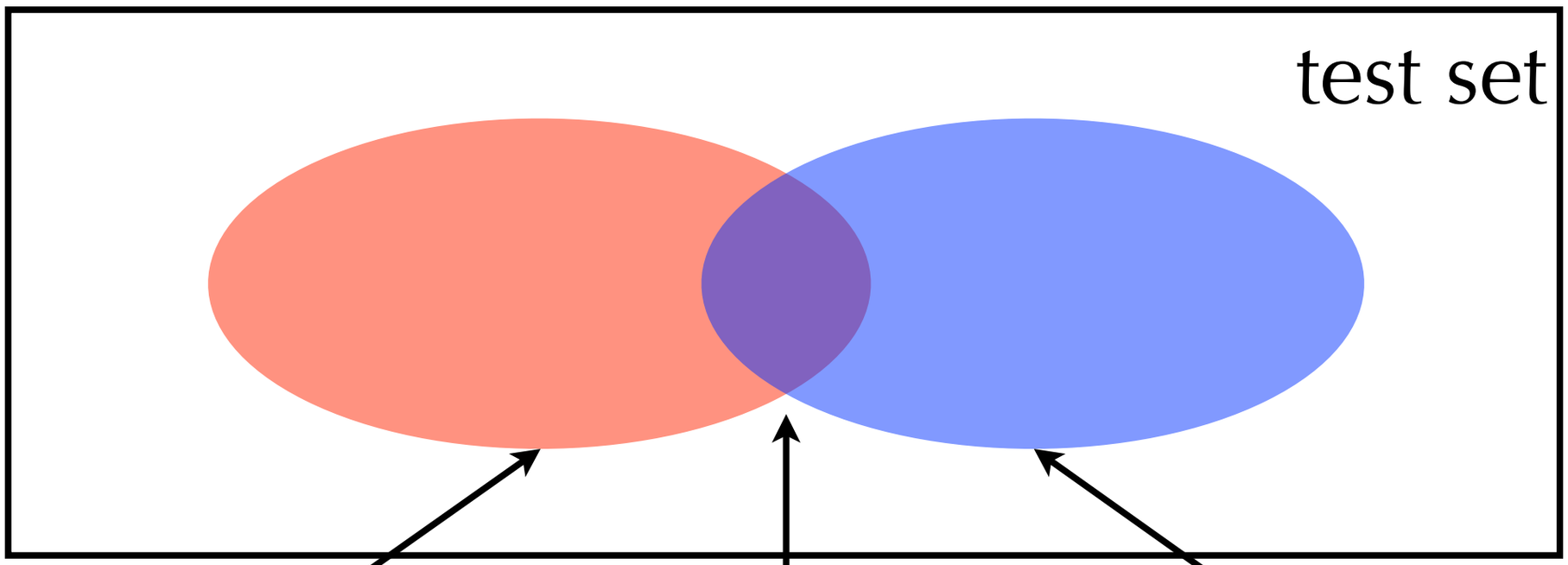
Evaluation Metrics

(2) precision and (3) recall



Evaluation Metrics
(2) precision and (3) recall

- Precision = ?



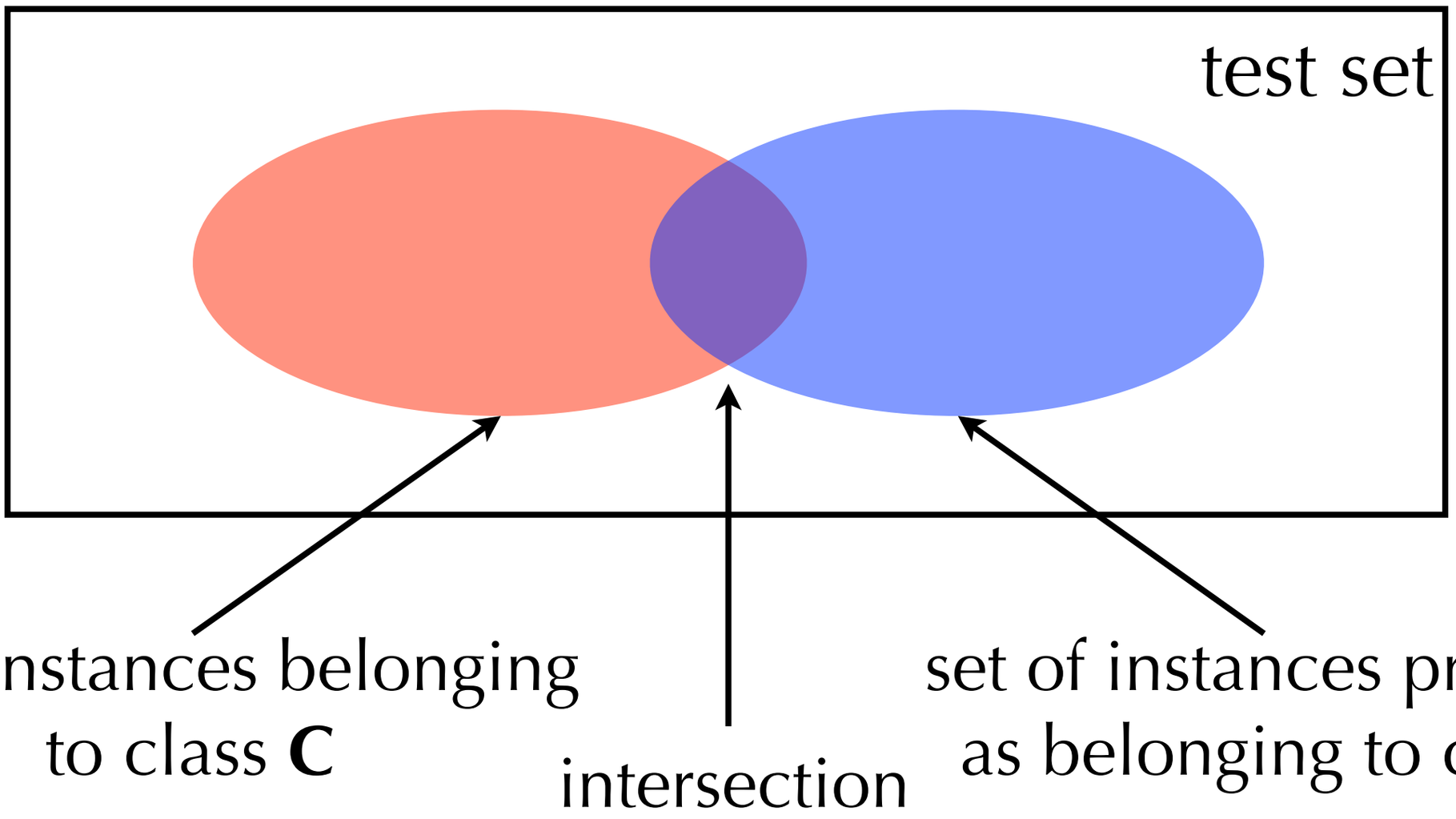
set of instances belonging to class **C**

intersection

set of instances predicted as belonging to class **C**

Evaluation Metrics
(2) precision and (3) recall

- Recall = ?



Evaluation Metrics

(2) precision

predicted \ true	pos	neut.	neg
pos	a	b	c
neut.	d	e	f
neg	g	h	i

$$\mathcal{P}_{\text{positive}} = \frac{a}{a+b+c}$$

Evaluation Metrics

(3) recall

		true		
		pos	neut.	neg
predicted	pos	a	b	c
	neut.	d	e	f
	neg	g	h	i

$$\mathcal{R}_{\text{positive}} = \frac{a}{a + d + g}$$

Evaluation Metrics

prevision vs. recall

		true		
		pos	neut.	neg
predicted	pos	a	b	c
	neut.	d	e	f
	neg	g	h	i

Evaluation Metrics
(4) precision-recall curves

- F-measure: assumes that the "end users" care equally about precision and recall

Evaluation Metrics
(4) precision-recall curves

- Most machine-learning algorithms provide a prediction confidence value

- The prediction confidence value can be used as a threshold in order to trade-off precision and recall

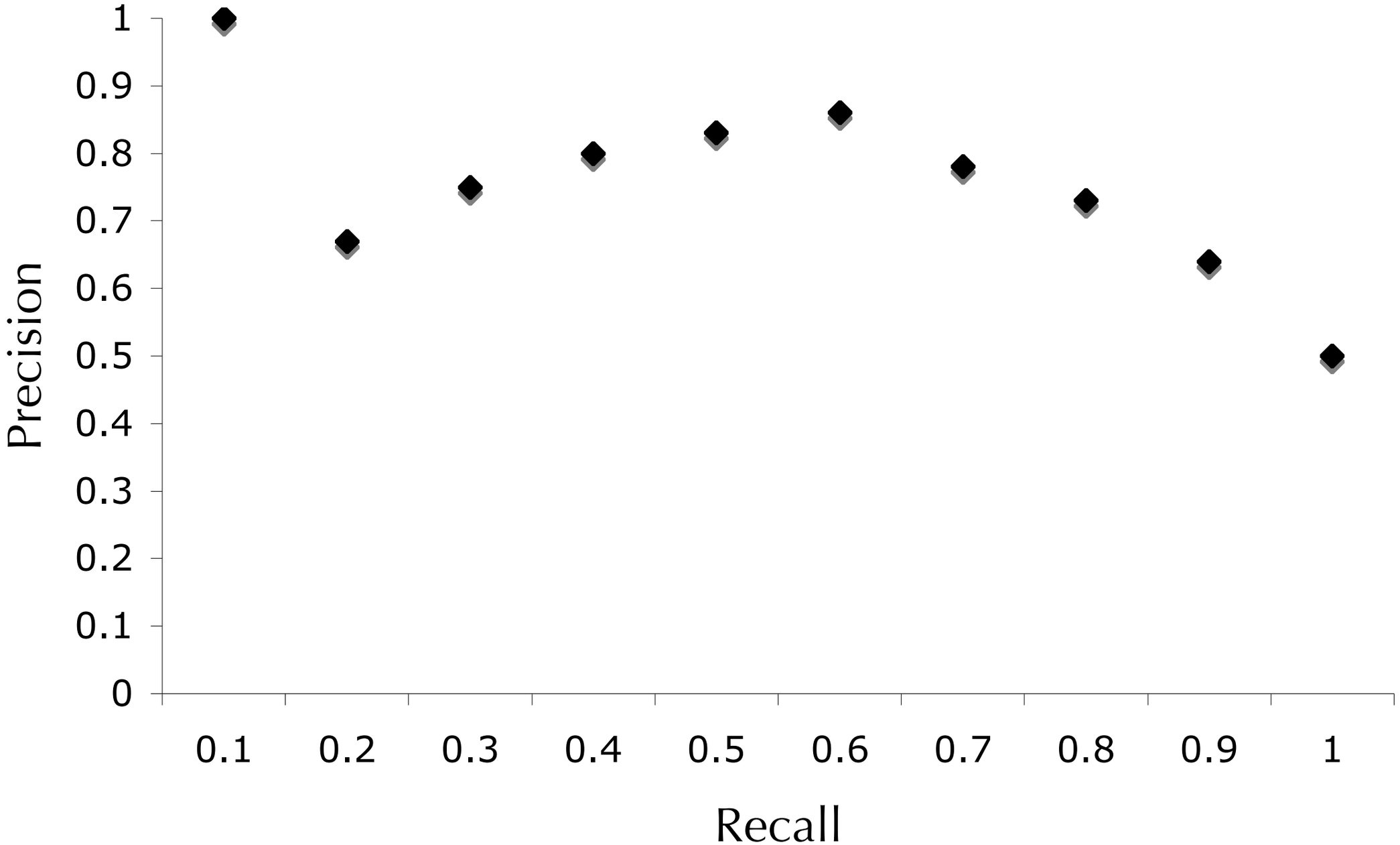
Evaluation Metrics

(4) precision-recall curves

rank (K)	ranking	P(POS|D)	P@K	R@K
1		0.99	1.00	0.10
2		0.87	0.50	0.10
3		0.84	0.67	0.20
4		0.83	0.75	0.30
5		0.77	0.80	0.40
6		0.63	0.83	0.50
7		0.58	0.86	0.60
8		0.57	0.75	0.60
9		0.56	0.78	0.70
10		0.34	0.70	0.70
11		0.33	0.73	0.80
12		0.25	0.67	0.80
13		0.21	0.62	0.80
14		0.15	0.64	0.90
15		0.14	0.60	0.90
16		0.14	0.56	0.90
17		0.12	0.53	0.90
18		0.08	0.50	0.90
19		0.01	0.47	0.90
20		0.01	0.50	1.00

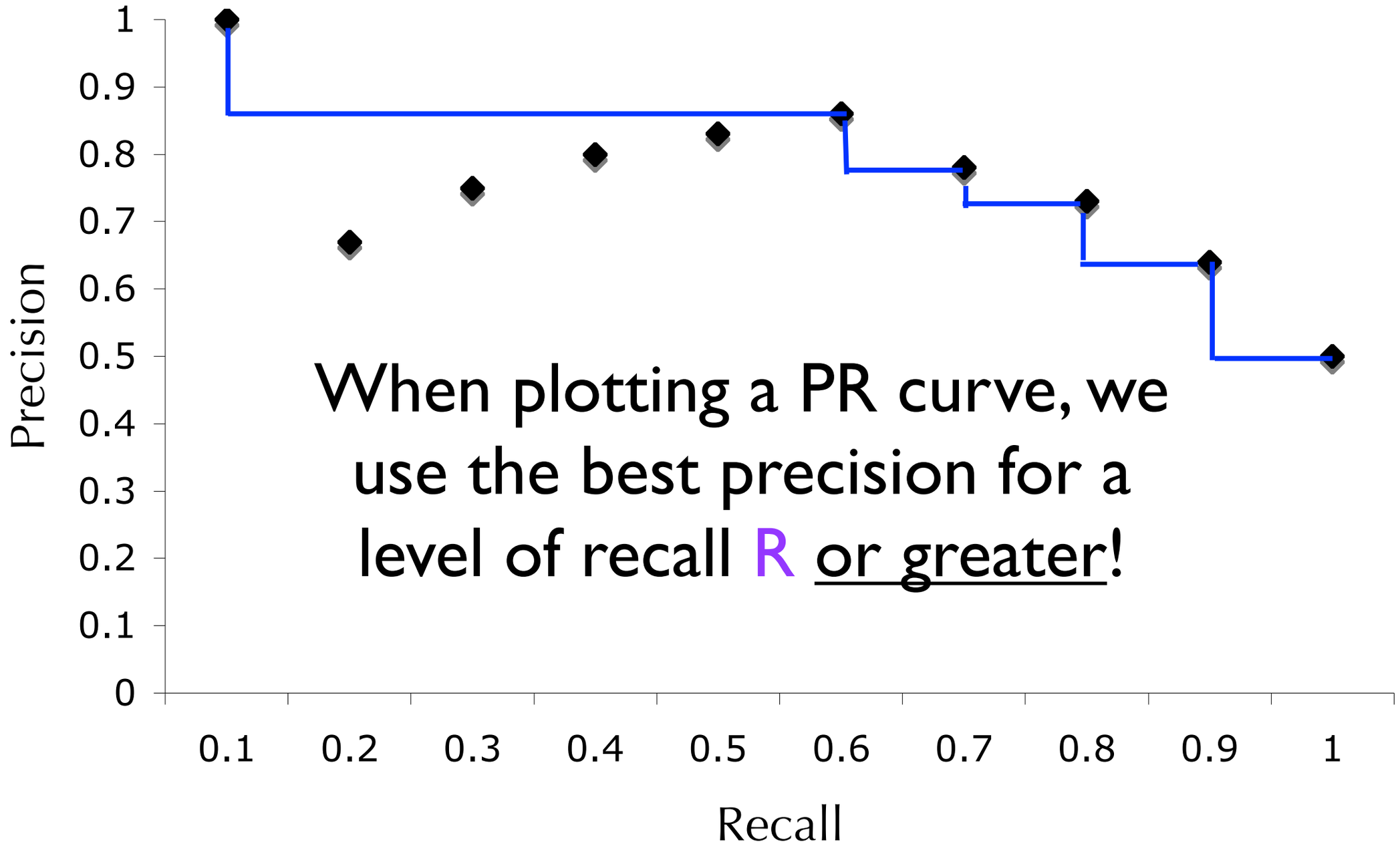
Evaluation Metrics

(4) precision-recall curves



Evaluation Metrics

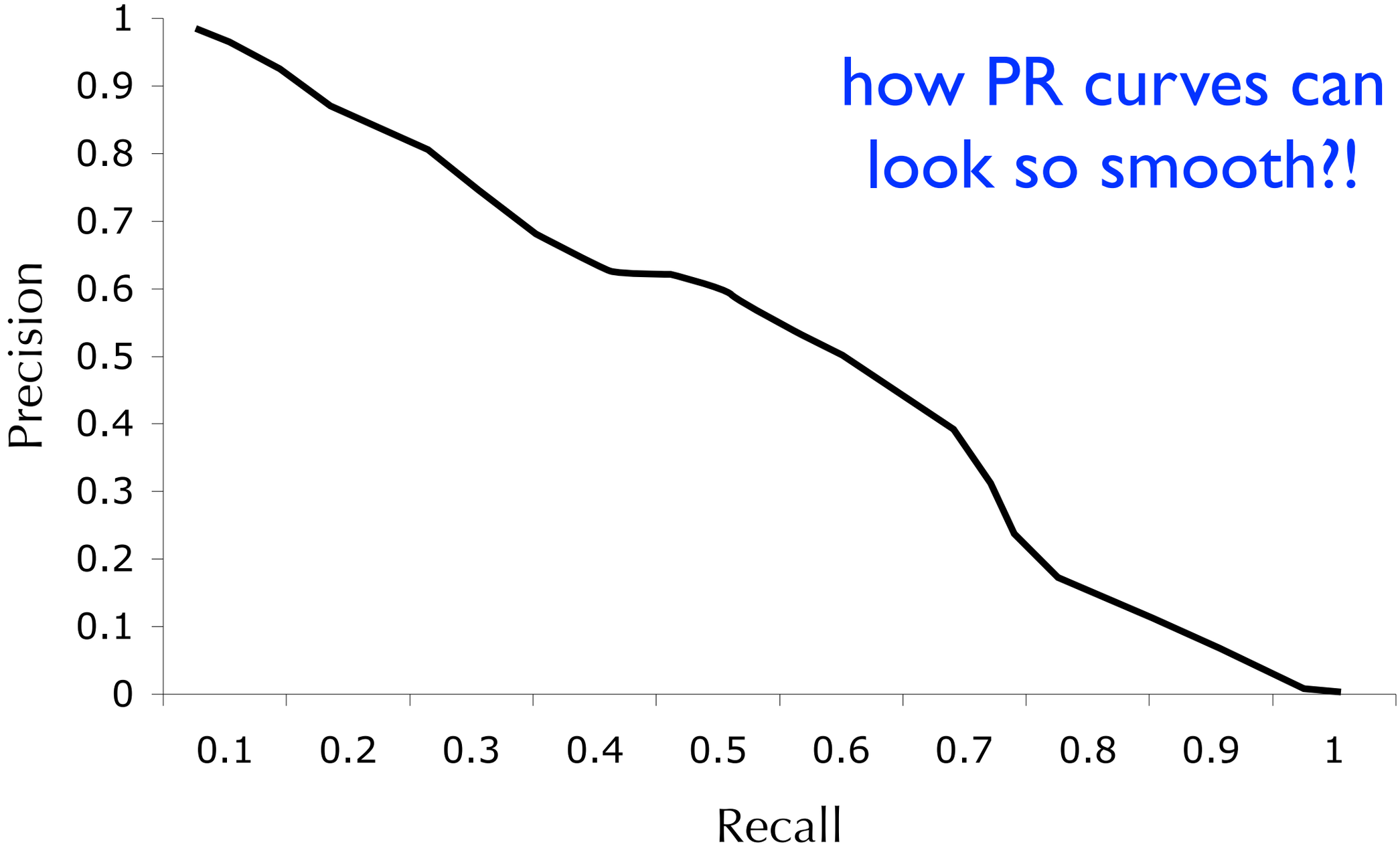
(4) precision-recall curves



When plotting a PR curve, we use the best precision for a level of recall R or greater!

Evaluation Metrics

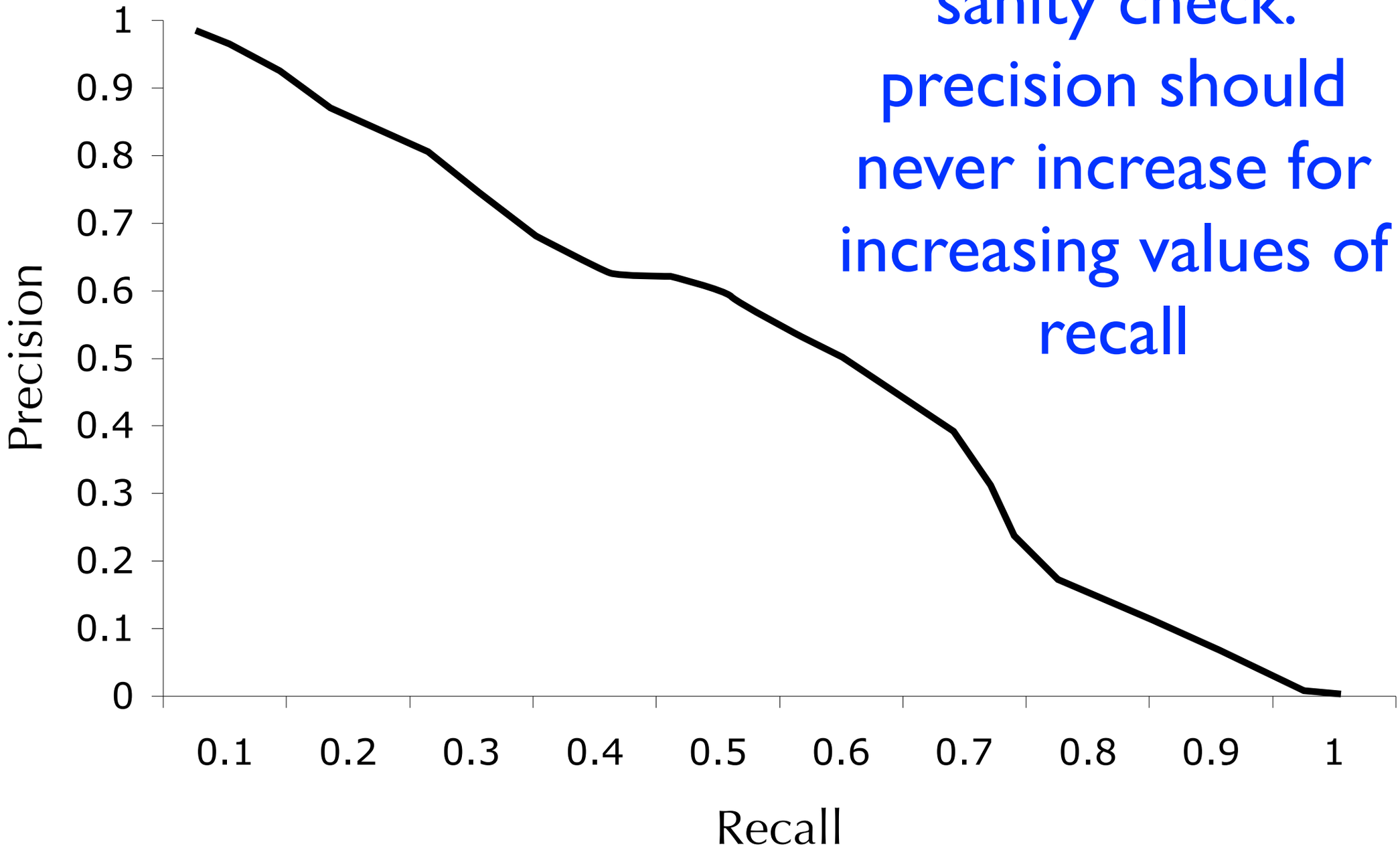
(4) precision-recall curves



how PR curves can look so smooth?!

Evaluation Metrics

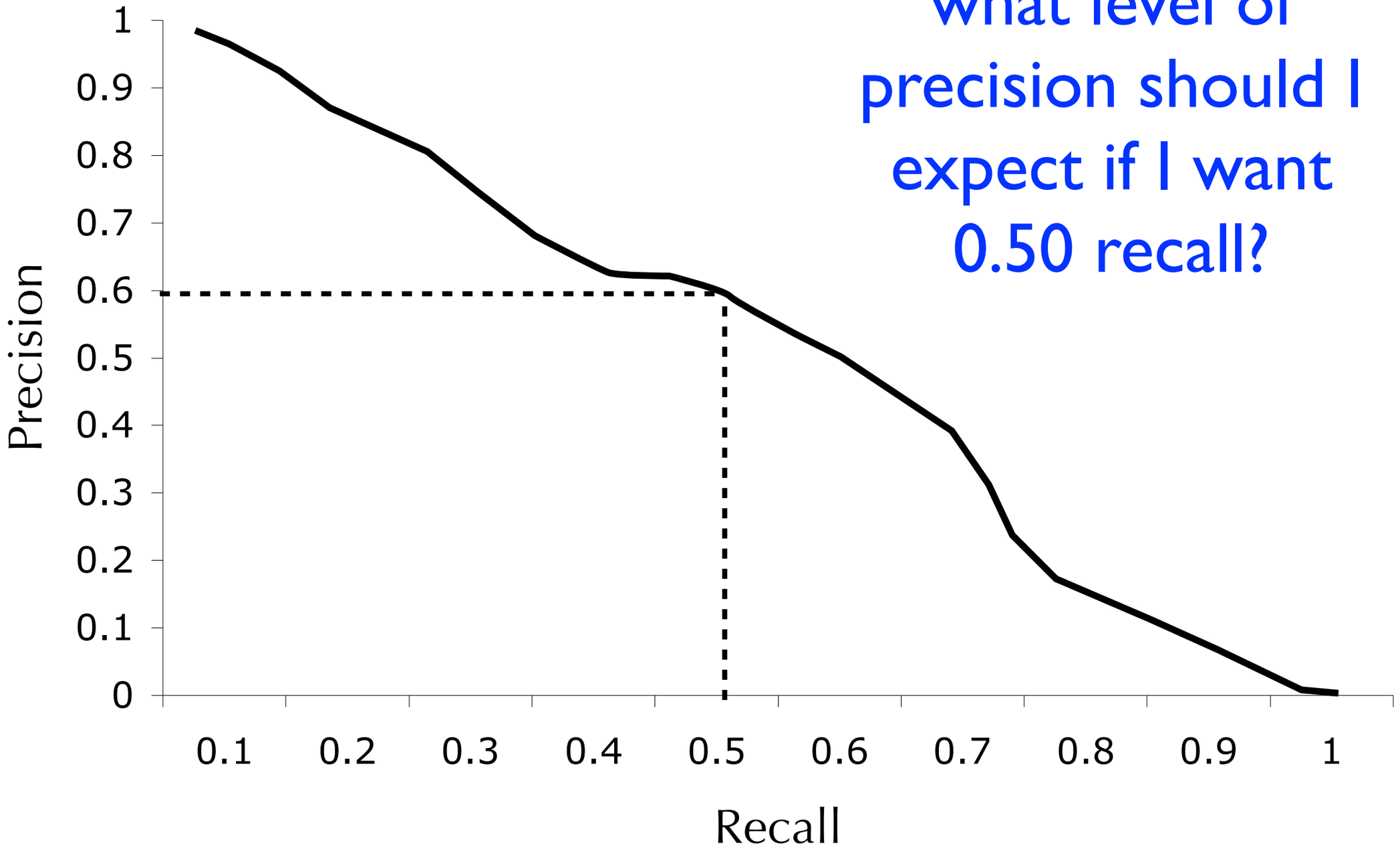
(4) precision-recall curves



sanity check: precision should never increase for increasing values of recall

Evaluation Metrics

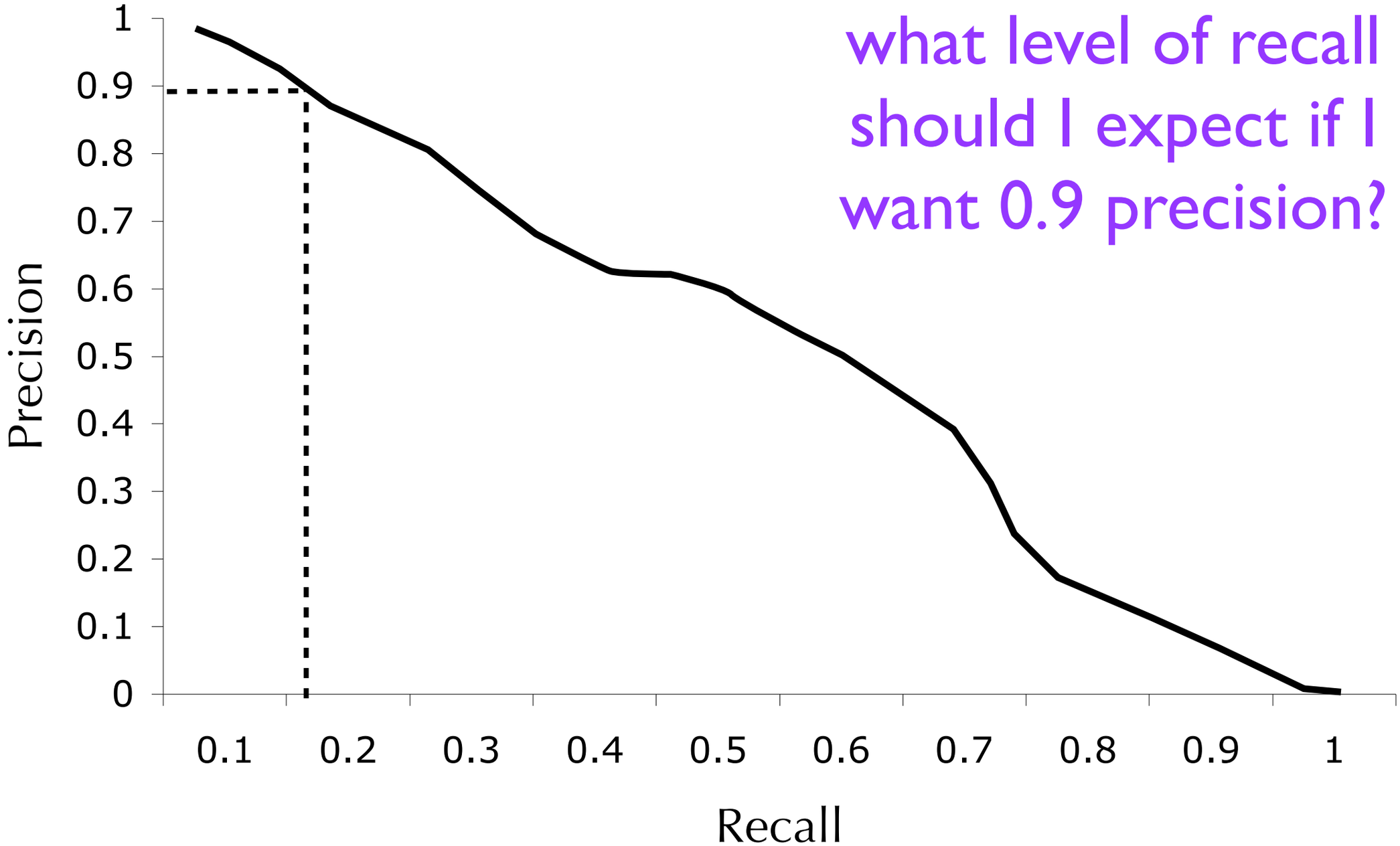
(4) precision-recall curves



what level of precision should I expect if I want 0.50 recall?

Evaluation Metrics

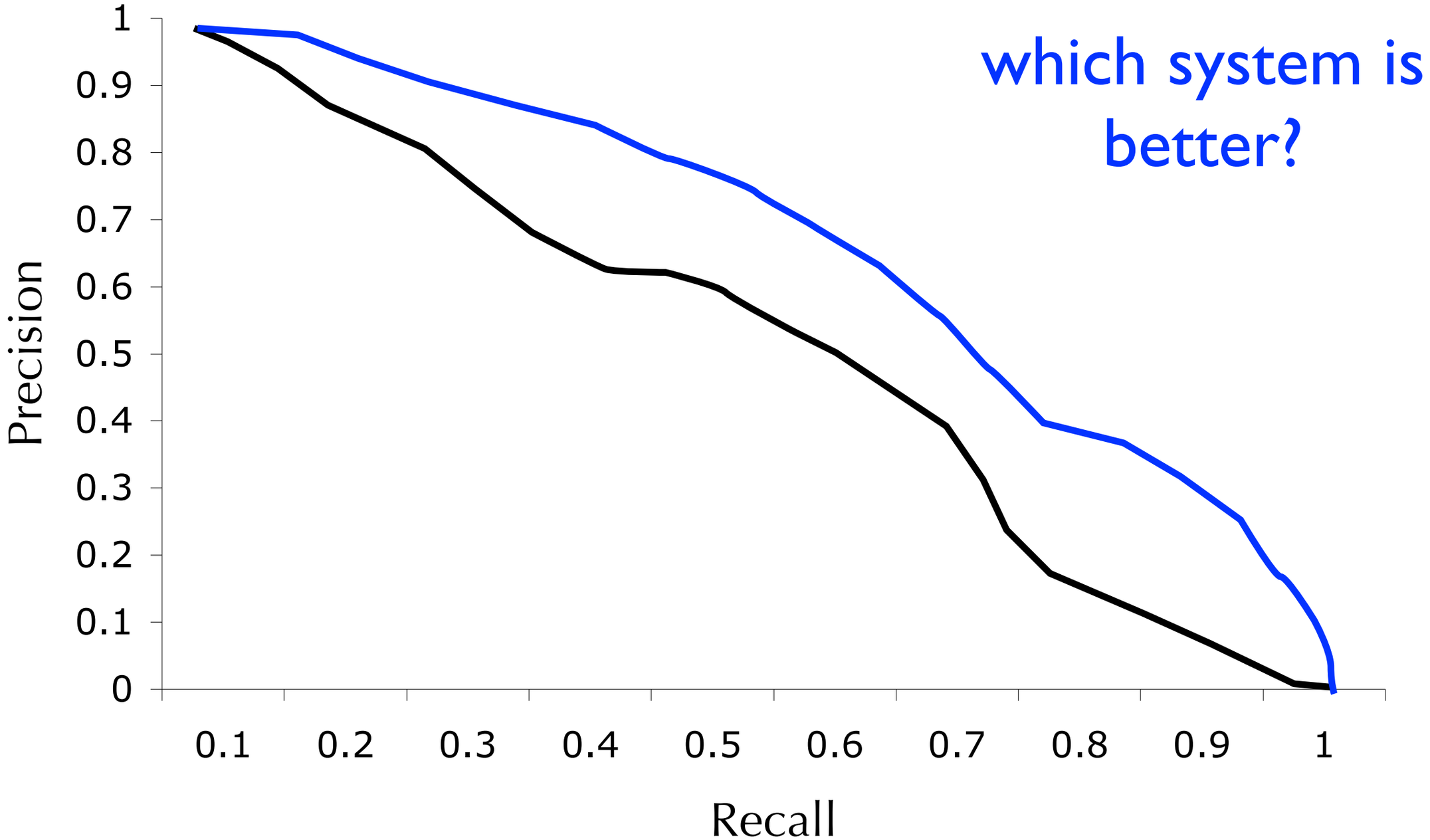
(4) precision-recall curves



what level of recall should I expect if I want 0.9 precision?

Evaluation Metrics

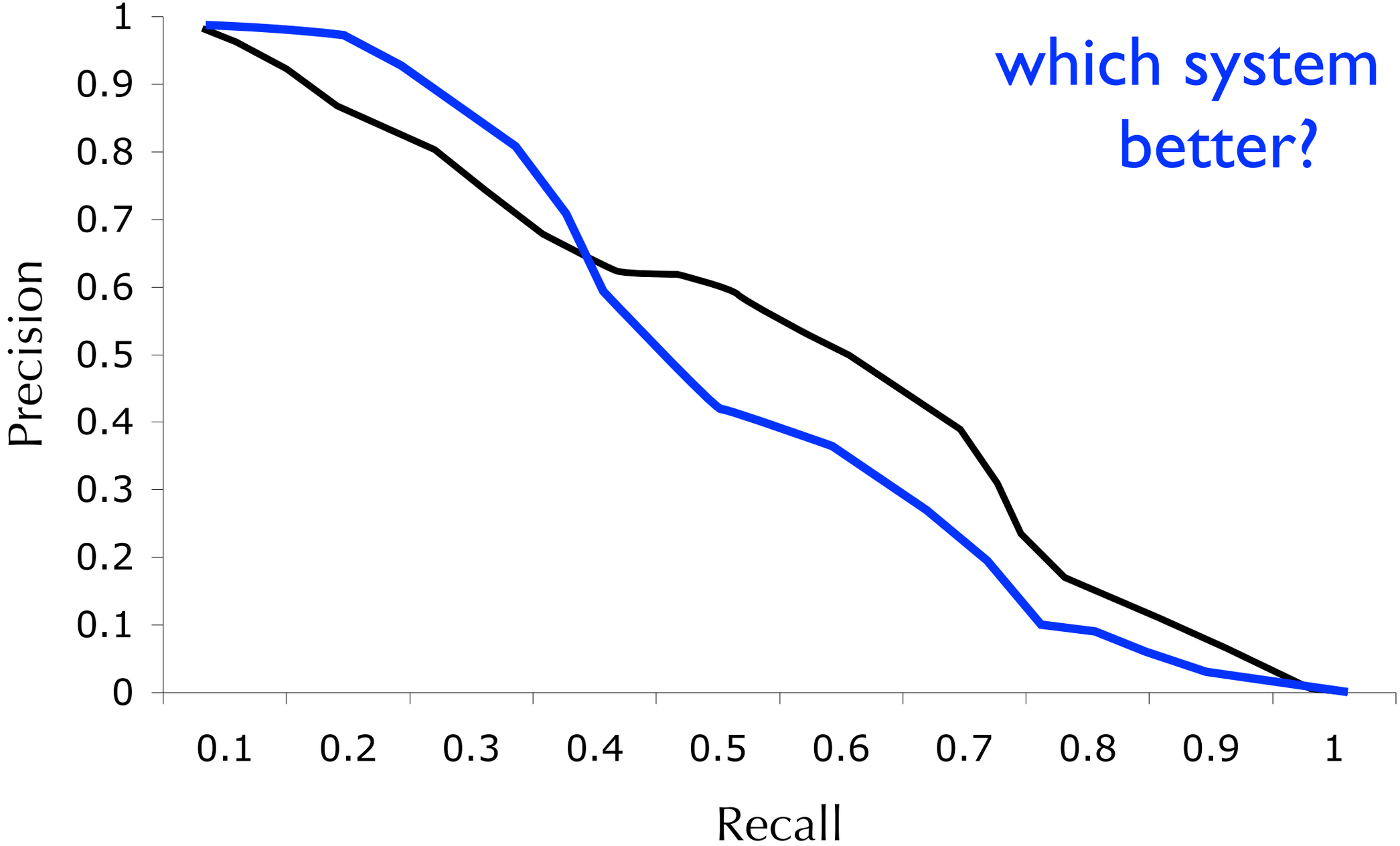
(4) precision-recall curves



which system is better?

Evaluation Metrics

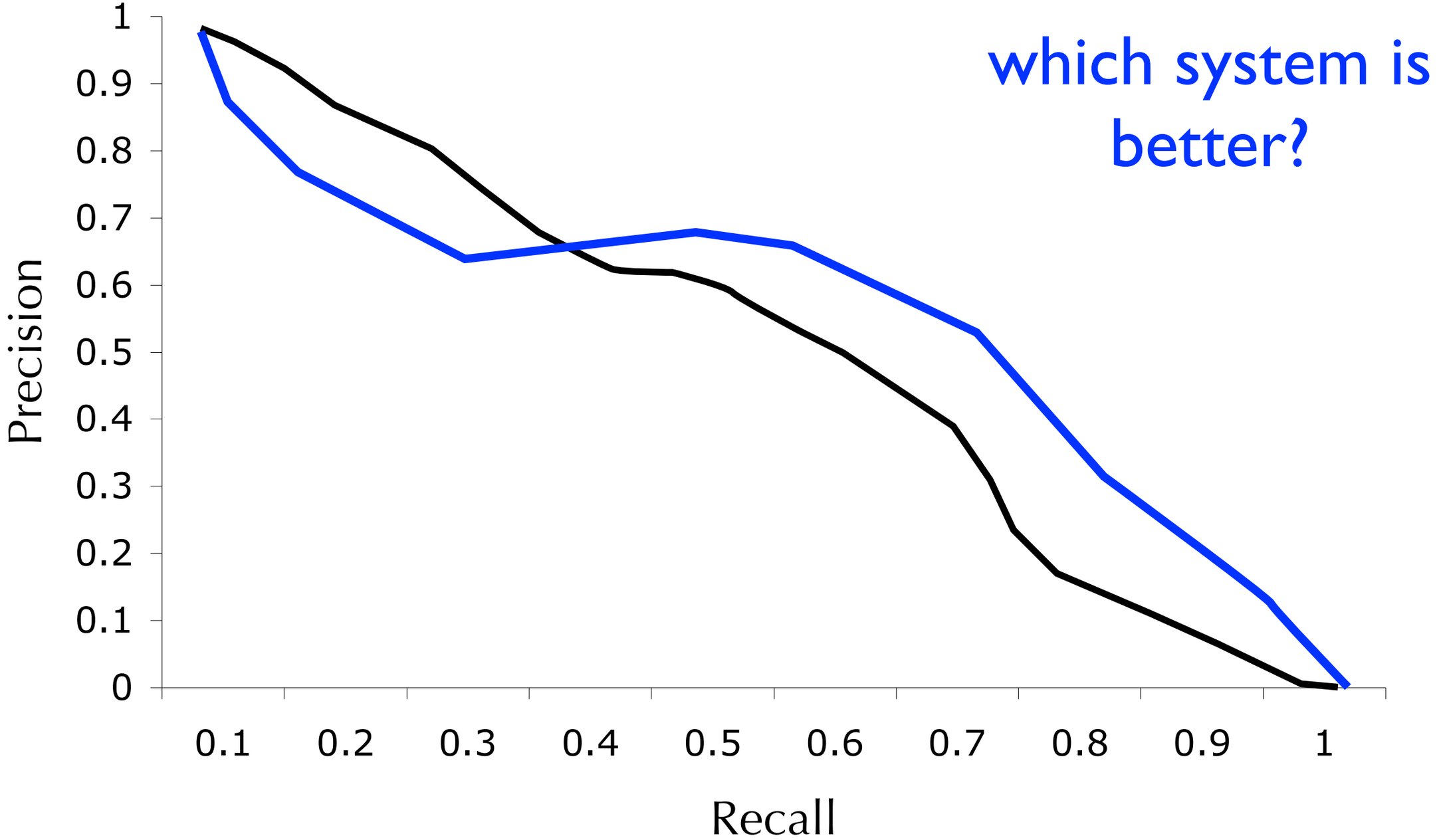
(4) precision-recall curves



which system is better?

Evaluation Metrics

(4) precision-recall curves



which system is better?

Evaluation Metrics

(4) precision-recall curves

content recommendation

PR curves for 'relevant'

Evaluation Metrics

(4) precision-recall curves

health monitoring

PR curves for ‘alarm’

Evaluation Metrics
(4) precision-recall curves

- PR curves show different precision-recall operating points (or trade-off points)
- How many false positives will I have to sift through for a desired level of recall?
- How many true positives will I have to miss for a desired level of precision?

Evaluation Metrics

- Accuracy
- Precision
- Recall
- PR curves (not a metric, but rather a way to show different PR operating points)

Outline: Predictive and Exploratory Analysis

Concepts, Instances, and Features

Human Annotation

Text Representation

Learning Algorithms

Evaluation metrics

Experimentation

Clustering

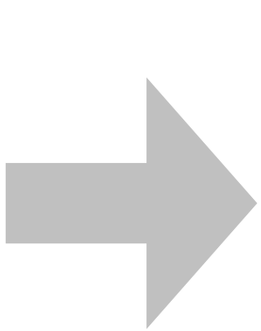
Hands-on Exercise

Training and Testing

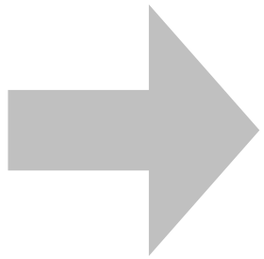
training

color	size	sides	equal sides	...	label
red	big	3	no	...	yes
green	big	3	yes	...	yes
blue	small	inf	yes	...	no
blue	small	4	yes	...	no
⋮	⋮	⋮	⋮	⋮	⋮
red	big	3	yes	...	yes

labeled examples



machine learning algorithm

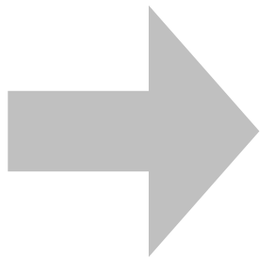


model

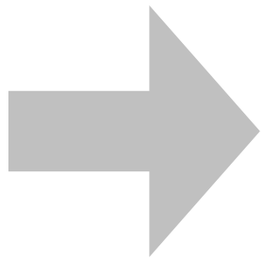
testing

color	size	sides	equal sides	...	label
red	big	3	no	...	???
green	big	3	yes	...	???
blue	small	inf	yes	...	???
blue	small	4	yes	...	???
⋮	⋮	⋮	⋮	⋮	???
red	big	3	yes	...	???

new, unlabeled examples



model



color	size	sides	equal sides	...	label
red	big	3	no	...	yes
green	big	3	yes	...	yes
blue	small	inf	yes	...	no
blue	small	4	yes	...	no
⋮	⋮	⋮	⋮	⋮	⋮
red	big	3	yes	...	yes

predictions

Training and Testing

- Suppose we have a set of labeled data
- Ultimately, we'll train a model using all this data and send the model out into the world to make predictions
- However, before we do this, we want to estimate its performance

What's wrong with this picture?

training

color	size	sides	equal sides	...	label
red	big	3	no	...	yes
green	big	3	yes	...	yes
blue	small	inf	yes	...	no
blue	small	4	yes	...	no
⋮	⋮	⋮	⋮	⋮	⋮
red	big	3	yes	...	yes

labeled set

machine learning algorithm

model

testing

color	size	sides	equal sides	...	label
red	big	3	no	...	???
green	big	3	yes	...	???
blue	small	inf	yes	...	???
blue	small	4	yes	...	???
⋮	⋮	⋮	⋮	⋮	???
red	big	3	yes	...	???

labeled set

model

color	size	sides	equal sides	...	label
red	big	3	no	...	yes
green	big	3	yes	...	yes
blue	small	inf	yes	...	no
blue	small	4	yes	...	no
⋮	⋮	⋮	⋮	⋮	⋮
red	big	3	yes	...	yes

predictions

Training and Testing

- Split the data into two sets.
- Learn a model using the training set
- Evaluate the model on the test set.

F = 0.50

Advantages and Disadvantages?

Single Train/Test Split

- Advantage
  - we are testing generalization performance.
- Disadvantage
  - we are putting all our eggs in one basket!
  - out of pure coincidence, the training set may have regularities that don't generalize to the test set

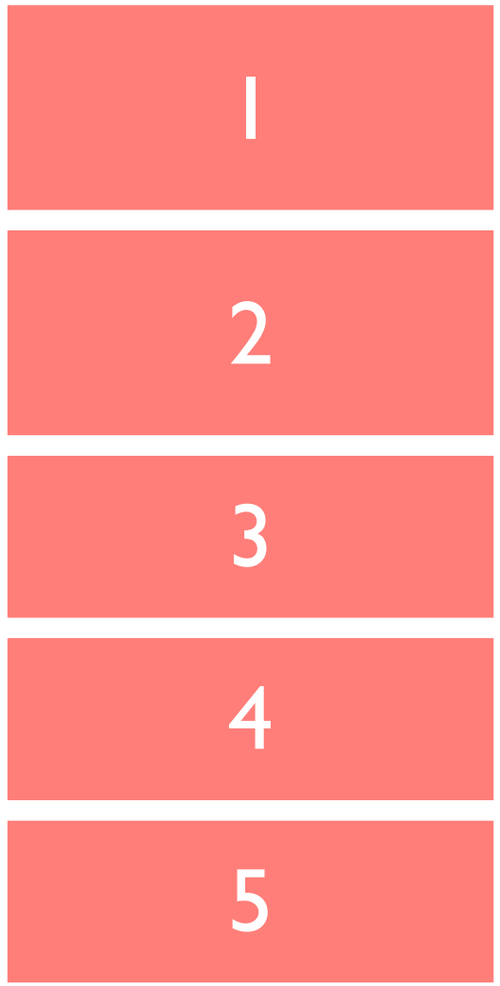
Cross-Validation

- N-fold Cross-validation
  1. divide the data into N sets of instances
  2. use the union of N-1 sets to find the best parameter values
  3. measure performance (using the best parameters) on the held-out set
  4. do steps 2-3 N times
  5. average performance across the N held-out sets
- This is called N-fold cross-validation (usually, N=10)

Cross-Validation

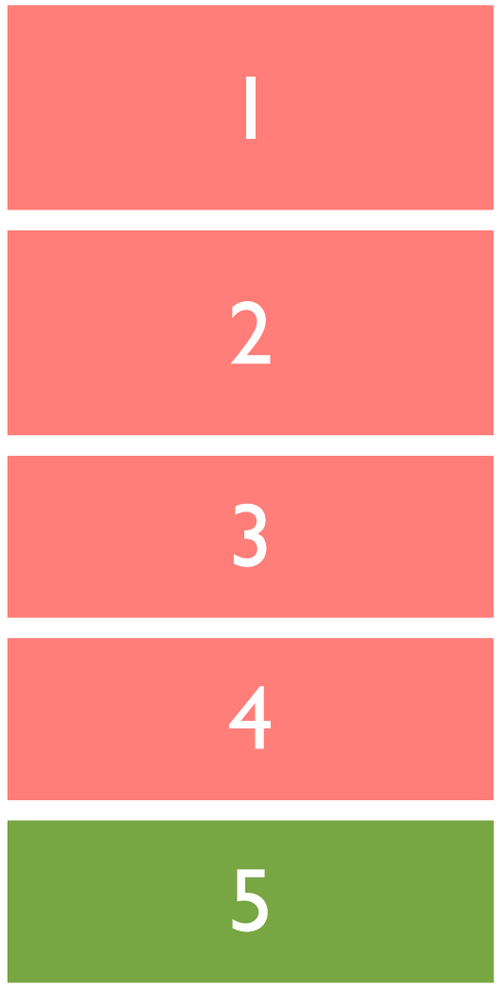
Cross-Validation

- Split the data into N = 5 folds



Cross-Validation

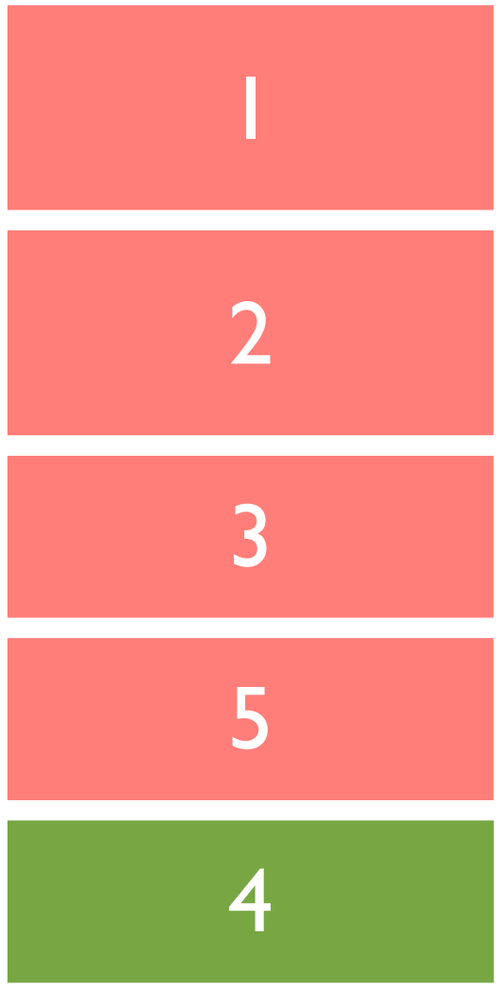
- For each fold, learn a model using the union of N - 1 folds as the training set and test the performance of this model on the held out fold



F = 0.50

Cross-Validation

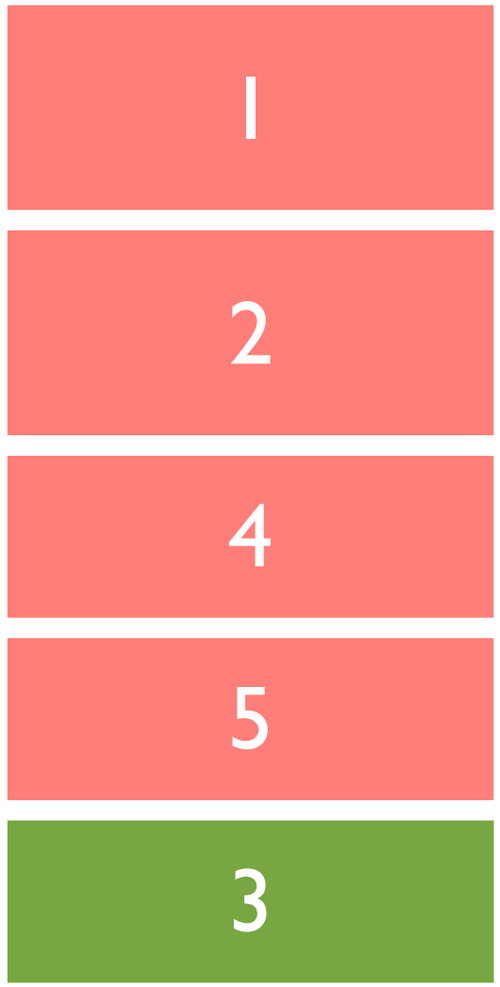
- For each fold, learn a model using the union of N - 1 folds as the training set and test the performance of this model on the held out fold



F = 0.60

Cross-Validation

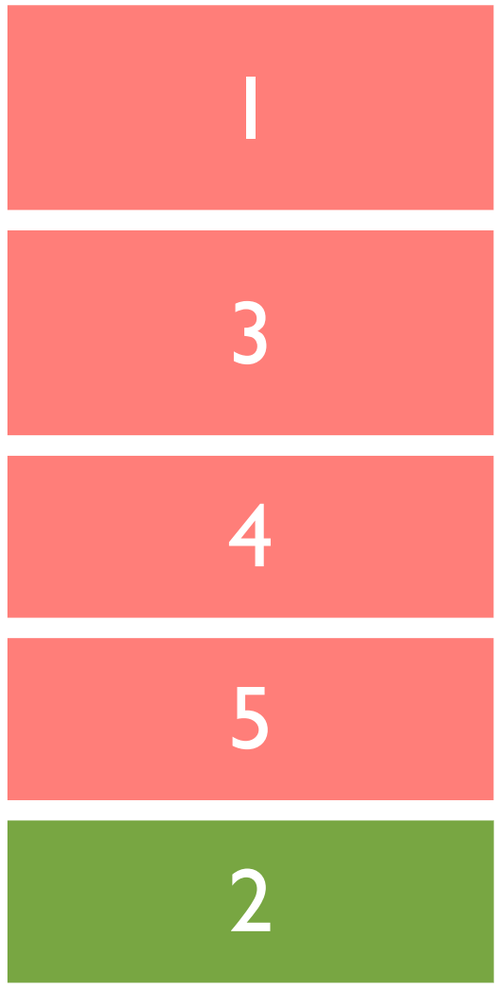
- For each fold, learn a model using the union of N - 1 folds as the training set and test the performance of this model on the held out fold



F = 0.70

Cross-Validation

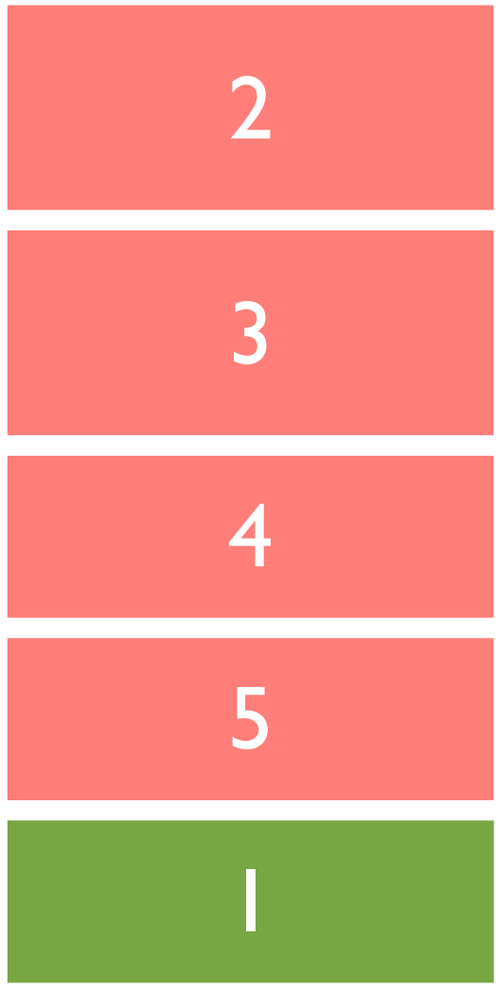
- For each fold, learn a model using the union of N - 1 folds as the training set and test the performance of this model on the held out fold



F = 0.60

Cross-Validation

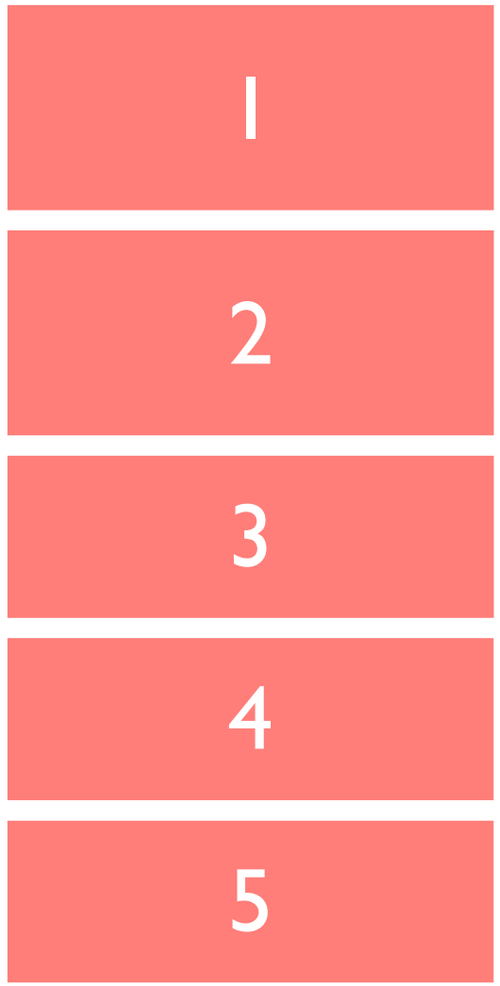
- For each fold, learn a model using the union of N - 1 folds as the training set and test the performance of this model on the held out fold



F = 0.50

Cross-Validation

- Average the performance across held-out folds



F = 0.50

F = 0.60

F = 0.70

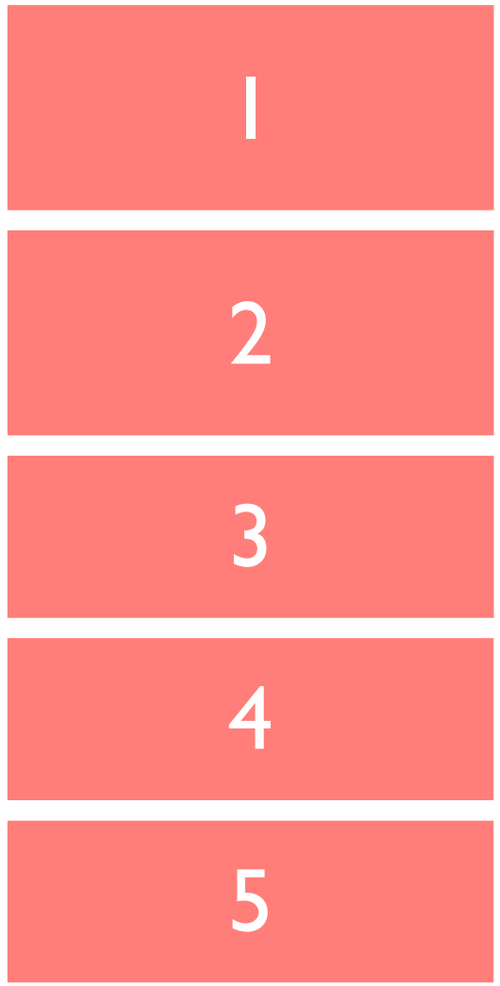
F = 0.60

F = 0.50

Average **F = 0.58**

Cross-Validation

- Average the performance across held-out folds



F = 0.50

F = 0.60

F = 0.70

F = 0.60

F = 0.50

Average **F = 0.58**

Advantages and Disadvantages?

N-Fold Cross-Validation

- Advantage
  - multiple rounds of generalization performance.
- Disadvantage
  - ultimately, we'll tune parameters on the whole dataset and send our system into the world.
  - a model trained on 100% of the data should perform better than one trained on 80%.
  - thus, we may be underestimating the model's performance!

Comparing Systems

- Train and test both systems using 10-fold cross validation
- Use the same folds for both systems
- Compare the difference in average performance across held-out folds

Fold	System A	System B
1	0.20	0.50
2	0.30	0.30
3	0.10	0.10
4	0.40	0.40
5	1.00	1.00
6	0.80	0.90
7	0.30	0.10
8	0.10	0.20
9	0.00	0.50
10	0.90	0.80
Average	0.41	0.48
	Difference	0.07

Significance Tests
motivation

- Why would it be risky to conclude that System B is better System A?
- Put differently, what is it that we're trying to achieve?

Significance Tests
motivation

- In theory: that the average performance of System B is greater than the average performance of System A for all possible test sets.

- However, we don't have all test sets. We have a sample
- And, this sample may favor one system vs. the other!

Significance Tests
definition

- A significance test is a statistical tool that allows us to determine whether a difference in performance reflects a true pattern or just random chance

Significance Tests
ingredients

- Test statistic: a measure used to judge the two systems (e.g., the difference between their average F-measure)
- Null hypothesis: no "true" difference between the two systems
- P-value: take the value of the observed test statistic and compute the probability of observing a value that large (or larger) under the null hypothesis

Significance Tests
ingredients

- If the p-value is large, we cannot reject the null hypothesis
- That is, we cannot claim that one system is better than the other

- If the p-value is small ($p<0.05$), we can reject the null hypothesis
- That is, the observed test-statistic is not due to random chance

Fisher's Randomization Test

procedure

- **Inputs:** counter = 0, N = 100,000
- Repeat N times:

  Step 1: for each fold, flip a coin and if it lands 'heads', flip the result between System A and B

  Step 2: see whether the test statistic is equal to or greater than the one observed and, if so, increment counter

- **Output:** counter / N

Fisher's Randomization Test

Fold	System A	System B
1	0.20	0.50
2	0.30	0.30
3	0.10	0.10
4	0.40	0.40
5	1.00	1.00
6	0.80	0.90
7	0.30	0.10
8	0.10	0.20
9	0.00	0.50
10	0.90	0.80
Average	0.41	0.48
	Difference	0.07

Fisher's Randomization Test

Fold	System A	System B
1	**0.50**	**0.20**
2	0.30	0.30
3	0.10	0.10
4	0.40	0.40
5	1.00	1.00
6	**0.90**	**0.80**
7	0.30	0.10
8	0.10	0.20
9	**0.50**	**0.00**
10	0.90	0.80
Average	0.5	0.39
	Difference	-0.11

at least 0.07?

iteration = 1 counter = 0

Fisher's Randomization Test

Fold	System A	System B
1	0.20	0.50
2	0.30	0.30
3	**0.10**	**0.10**
4	0.40	0.40
5	**1.00**	**1.00**
6	0.80	0.90
7	**0.10**	**0.30**
8	**0.20**	**0.10**
9	0.00	0.50
10	**0.08**	**0.90**
Average	0.318	0.5
	Difference	0.182

at least 0.07?

iteration = 2 counter = 1

Fisher's Randomization Test

Fold	System A	System B
1	**0.50**	**0.20**
2	0.30	0.30
3	**0.10**	**0.10**
4	**0.40**	**0.40**
5	1.00	1.00
6	**0.90**	**0.80**
7	0.30	0.10
8	0.10	0.20
9	**0.50**	**0.00**
10	0.90	0.80
Average	0.5	0.39
	Difference	-0.11

at least 0.07?

iteration = 100,000 counter = 25,678

Fisher's Randomization Test

procedure

- **Inputs:** counter = 0, N = 100,000
- Repeat N times:

  Step 1: for each query, flip a coin and if it lands 'heads', flip the result between System A and B

  Step 2: see whether the test statistic is equal to or greater than the one observed and, if so, increment counter

- **Output:** counter / N = (25,678/100,00) = 0.25678

Fisher's Randomization Test

- Under the null hypothesis, the probability of observing a value of the test statistic of 0.07 or greater is about 0.26.
- Because $p > 0.05$, we cannot confidently say that the value of the test statistic is not due to random chance.
- A difference between the average F-measure values of 0.07 is not significant

Outline: Predictive and Exploratory Analysis

Concepts, Instances, and Features

Human Annotation

Text Representation

Learning Algorithms

Evaluation metrics

Experimentation

Clustering

Hands-on Exercise

Clustering
objective

- Grouping documents or instances into subsets or clusters
- Documents in the same cluster should be similar
- Documents in different clusters should be dissimilar

- A common form of unsupervised learning
- Unsupervised = no human-produced labels
- The goal is to discover structure from the data

Clustering vs. Classification

- Classification:
  - the input to the system is a set of labeled data
  - the algorithm learns a model for predicting the label on new examples
- Clustering:
  - the input the system is a set of unlabeled data
  - the algorithm infers the labels from the data and assigns a label to each input instance

Clustering
objective

- Grouping documents or instances into subsets or clusters
- Documents within a the same cluster should be similar
- Documents from different clusters should be dissimilar

Clustering
basics

- What does it mean for documents to be “similar” or “dissimilar”?

Clustering
basics

- What does it mean for documents to be similar or dissimilar?
- We need a computational way of modeling similarity
- One solution: model similarity using distance in a vector space representation of the collection or dataset
  - small distance = high similarity
  - long distance = low similarity

Vector Space Representation

review

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10
1	0	1	0	1	0	0	1	1	0
0	1	0	1	1	0	1	1	0	0
0	1	0	1	1	0	1	0	0	0
0	0	1	0	1	1	0	1	1	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	0	1	1	0	0	1	0	1

Vector Space Representation

review

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10
1	0	1	0	1	0	0	1	1	0
0	1	0	1	1	0	1	1	0	0
0	1	0	1	1	0	1	0	0	0
0	0	1	0	1	1	0	1	1	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	0	1	1	0	0	1	0	1

- We can represent this document as a vector in a 10-dimensional vector space

Vector Space Representation

review

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10
1	0	1	0	1	0	0	1	1	0
0	1	0	1	1	0	1	1	0	0
0	1	0	1	1	0	1	0	0	0
0	0	1	0	1	1	0	1	1	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	0	1	1	0	0	1	0	1

- This representation assumes binary term-weights.
- Are there other term-weighting schemes?

Vector Space Representation

review

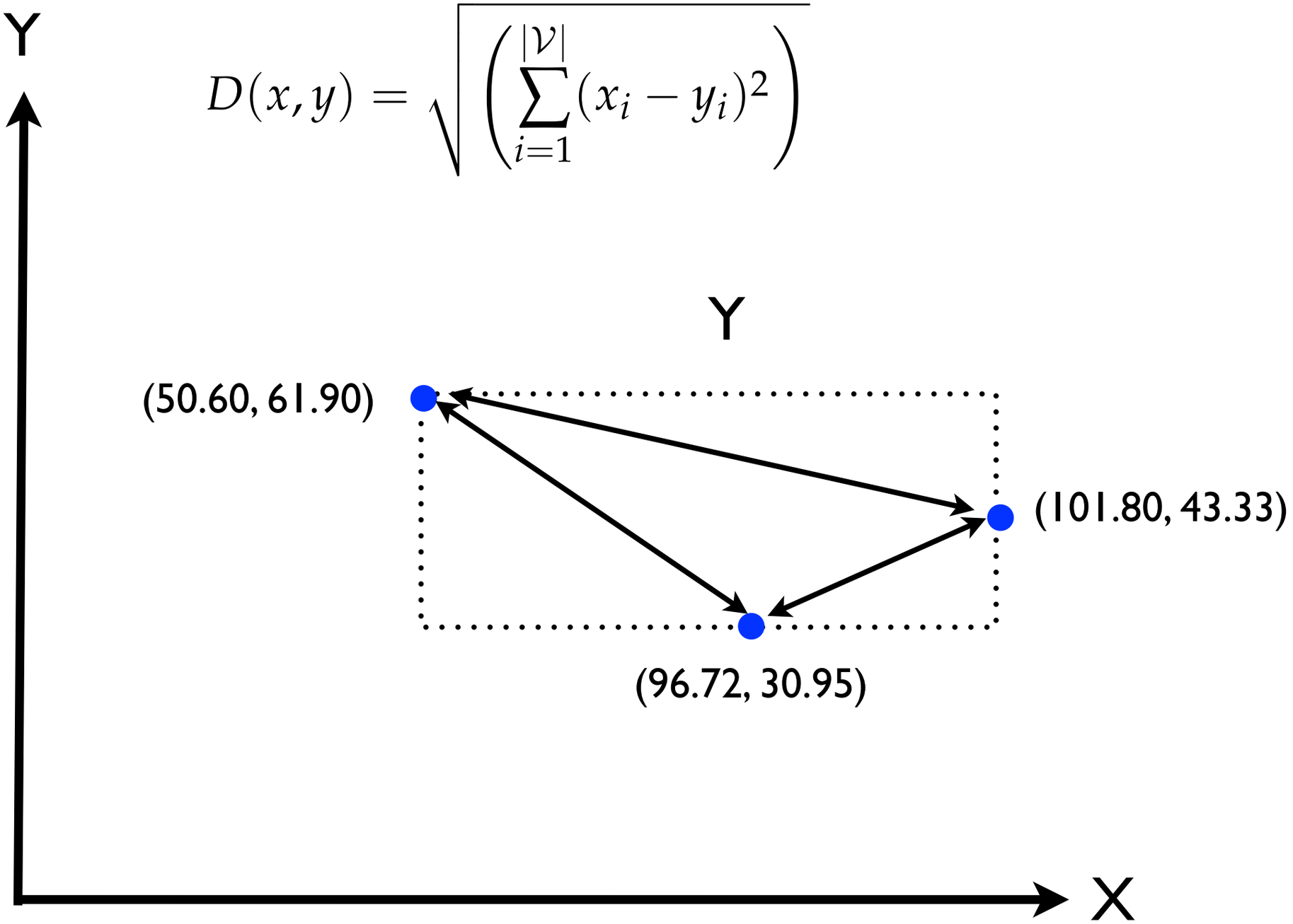
- Similarity = Euclidean Distance:

$$D(x,y) = \sqrt{\left(\sum_{i=1}^{|\mathcal{V}|}(x_i - y_i)^2\right)}$$

Vector Space Representation

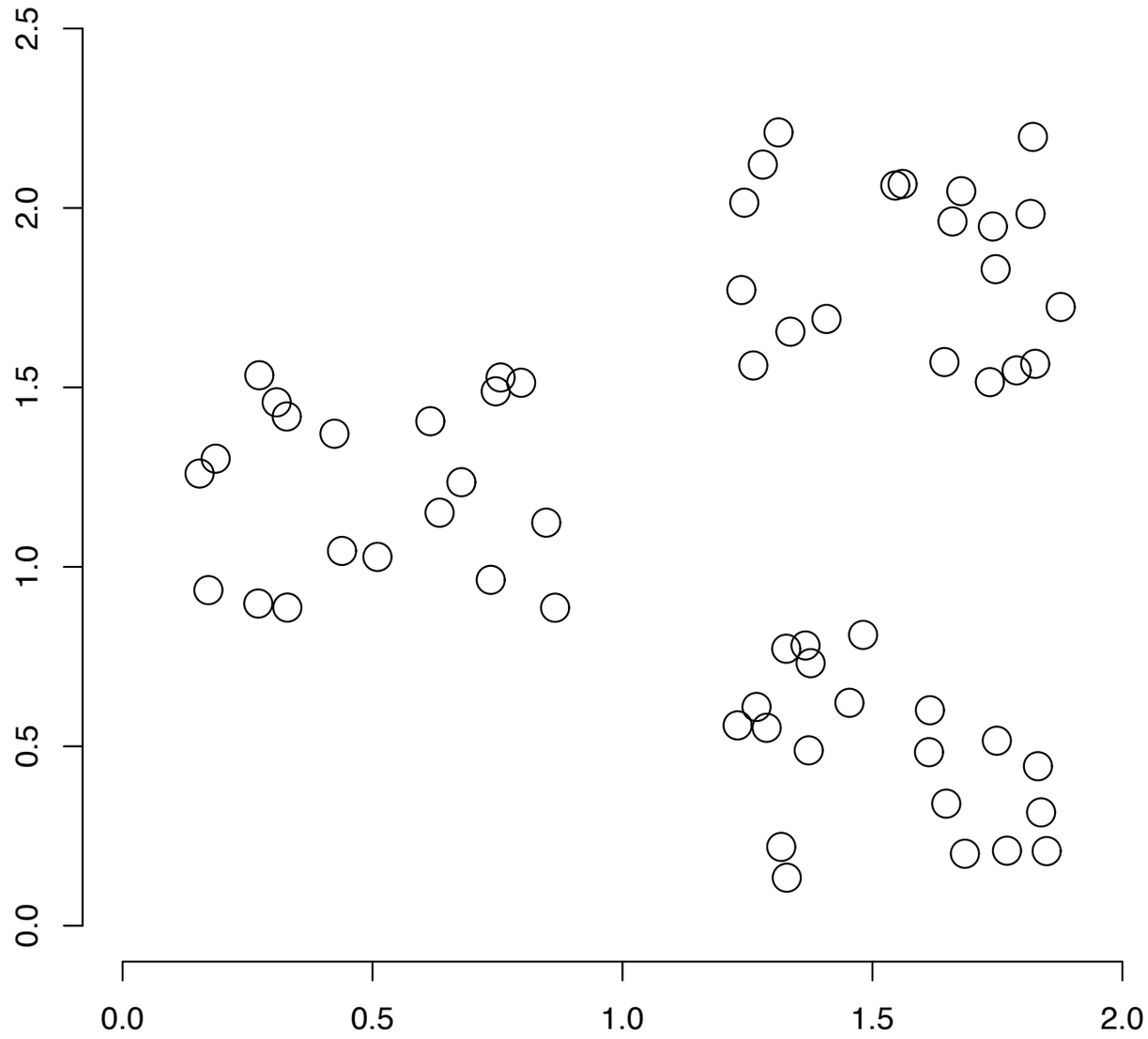
review

$$D(x,y) = \sqrt{\left(\sum_{i=1}^{|\mathcal{V}|}(x_i - y_i)^2\right)}$$



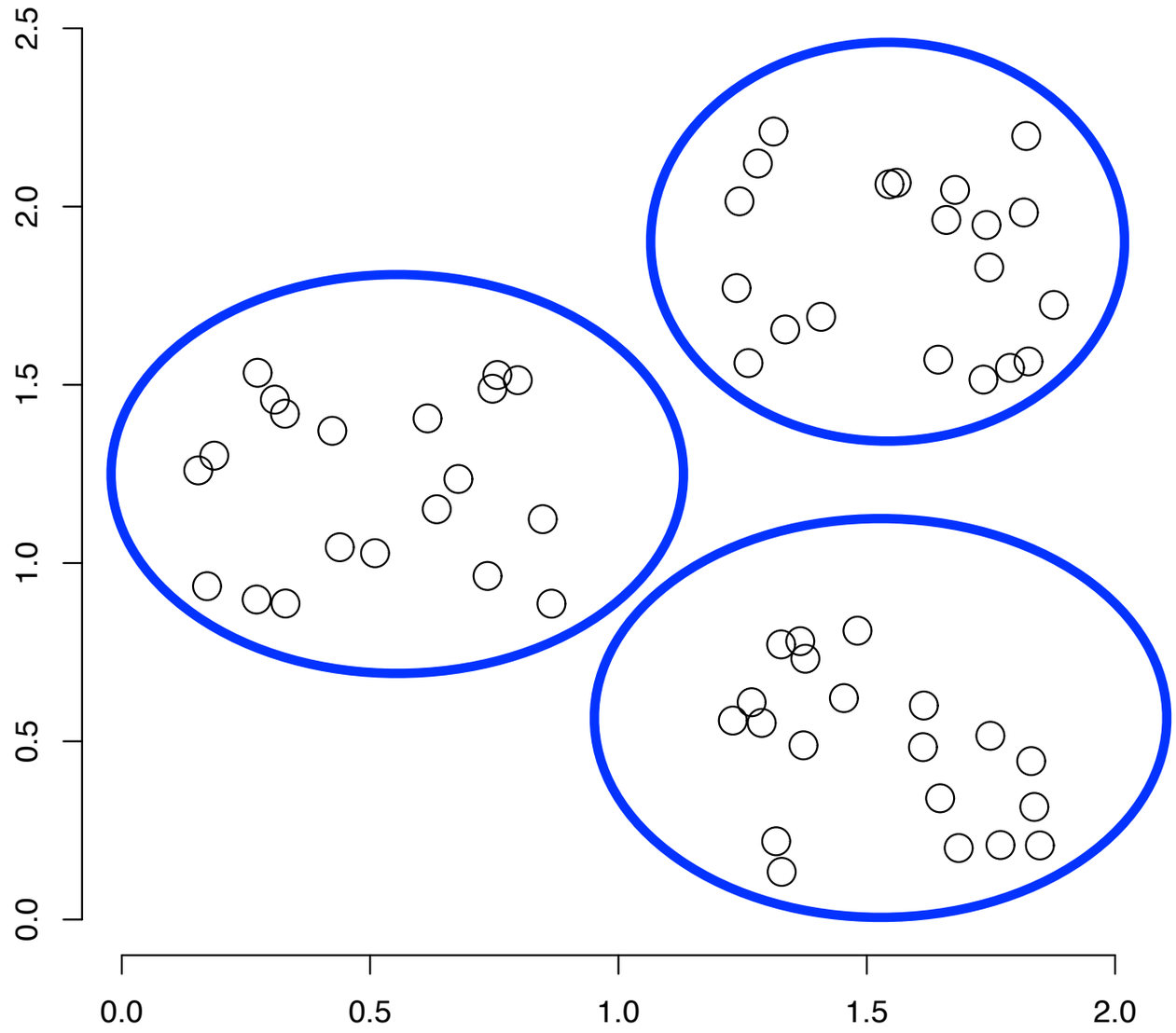
Clustering

- What would we expect a clustering algorithm to do with this dataset?



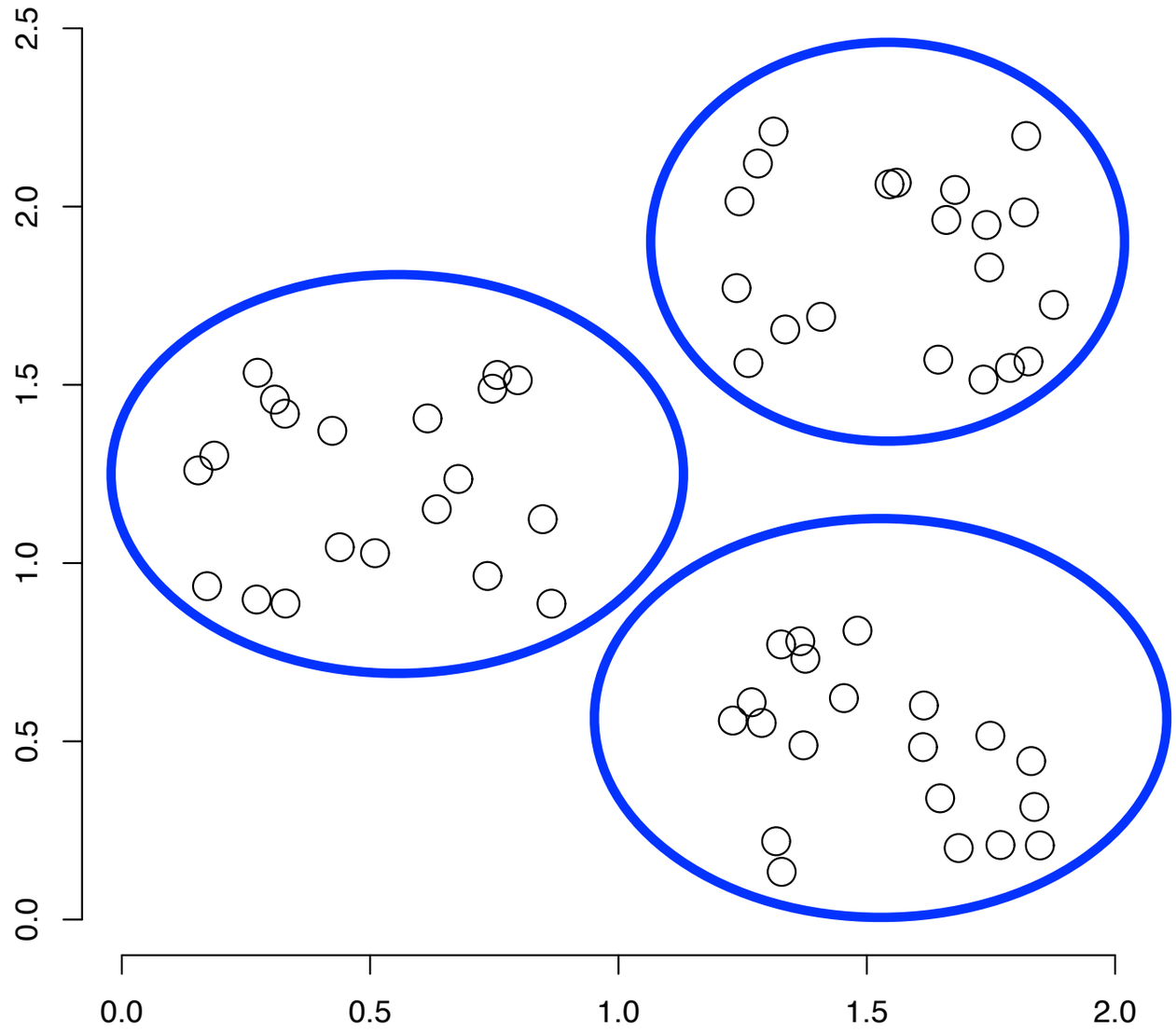
Clustering

- What would we expect a clustering algorithm to do with this dataset?



Clustering

- Propose an algorithm that might be able to do this!

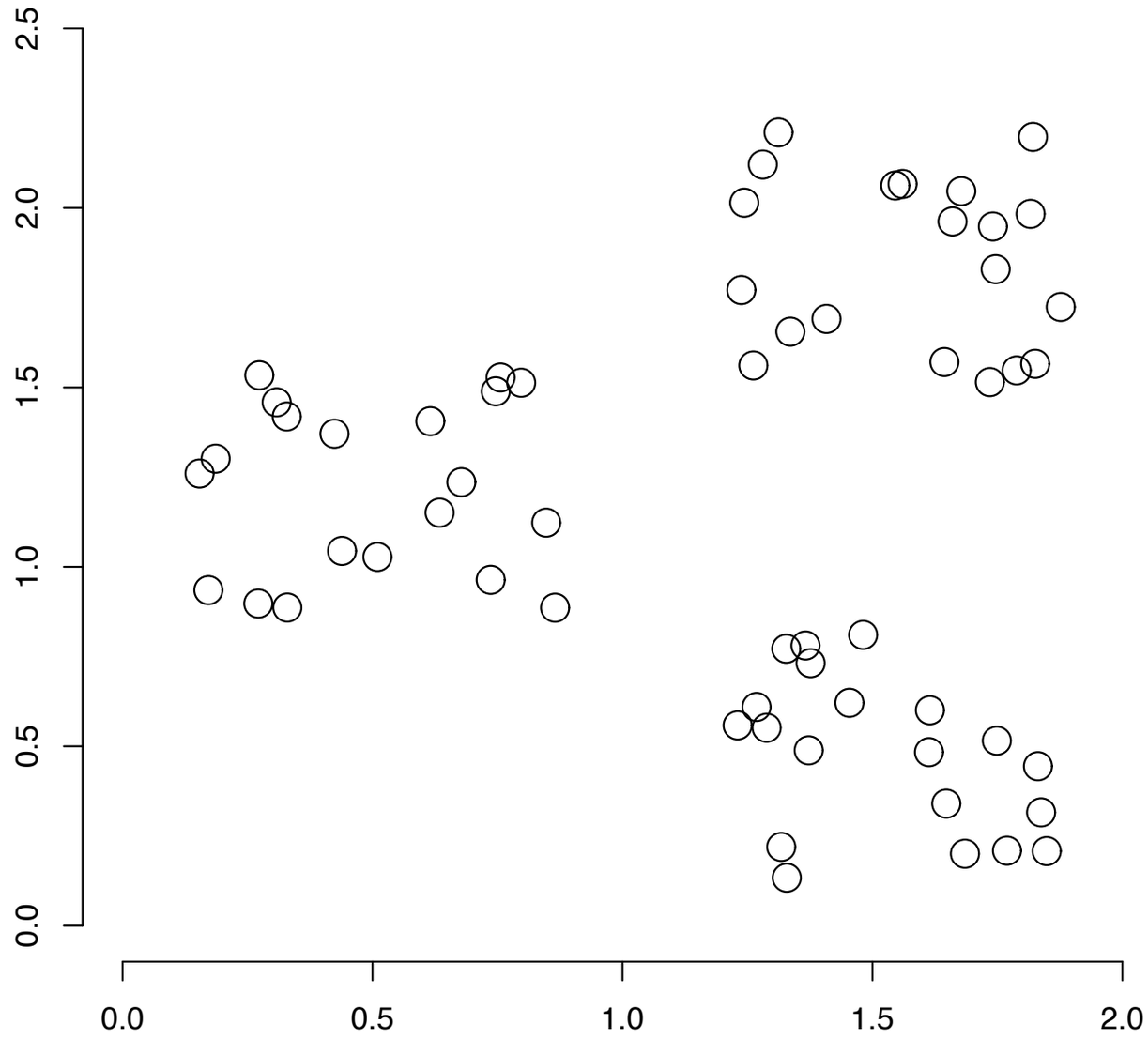


Clustering

- Input: number of desired clusters K
- Output: assignment of documents to K clusters
- Algorithm:
  - randomly select K documents (seeds)
  - assign each remaining document to its nearest seed

Clustering

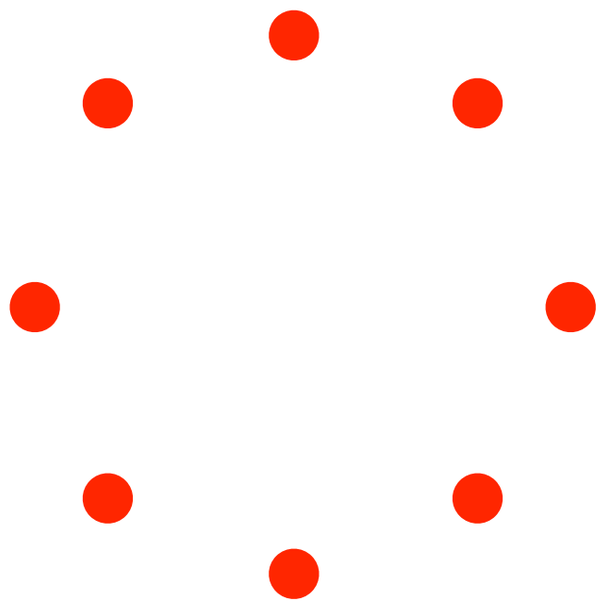
- Could this work?



K-Means Clustering

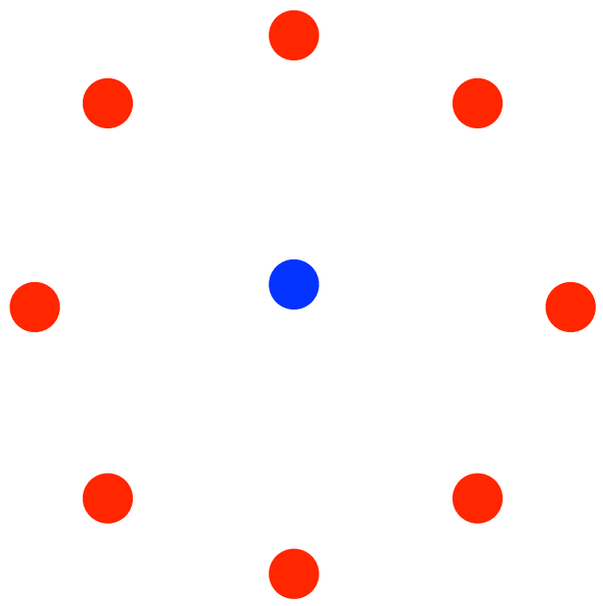
K-means Clustering
cluster centroid

- The key to understanding K-means clustering is to understand the idea of a cluster centroid
- Given a cluster, you can think of its centroid as a point (or vector) that corresponds to its “center of mass”



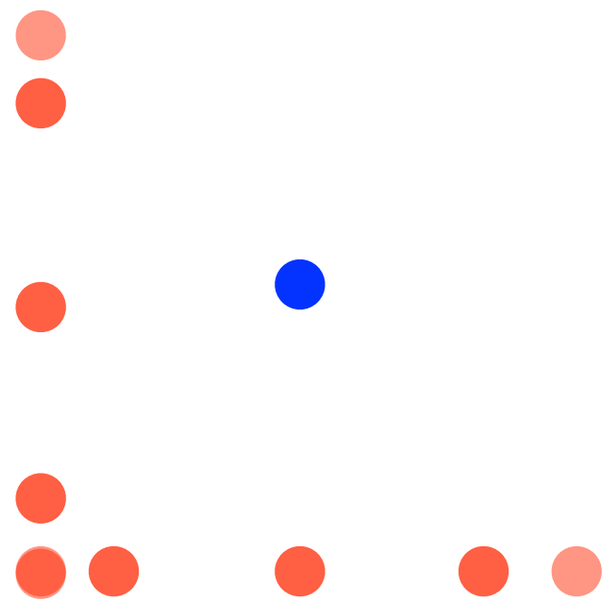
K-means Clustering
cluster centroid

- The key to understanding K-means clustering is to understand the idea of a cluster centroid
- Given a cluster, you can think of its centroid as a point (or vector) that corresponds to its “center of mass”



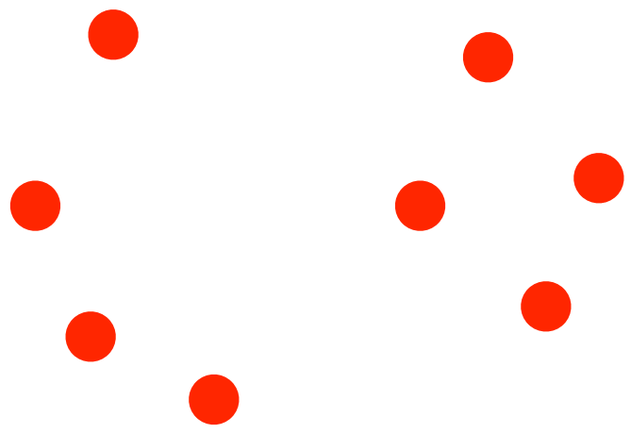
K-means Clustering
cluster centroid

- The key to understanding K-means clustering is to understand the idea of a cluster centroid
- Given a cluster, you can think of its centroid as a point (or vector) that corresponds to its “center of mass”



K-means Clustering
cluster centroid

- The key to understanding K-means clustering is to understand the idea of a cluster centroid
- Given a cluster, you can think of its centroid as a point (or vector) that corresponds to its “center of mass”

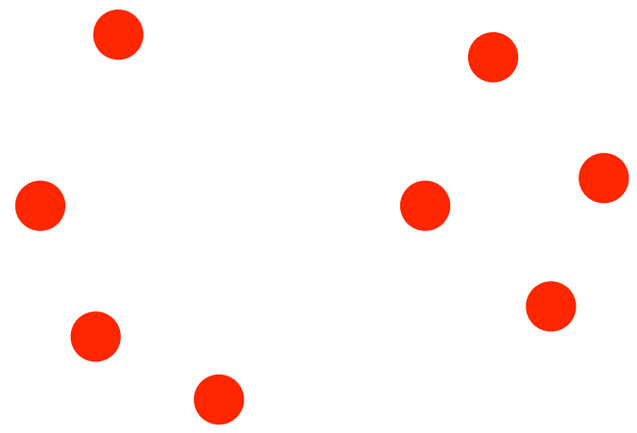


K-means Clustering
cluster centroid

- The key to understanding K-means clustering is to understand the idea of a cluster centroid
- Given a cluster, you can think of its centroid as a point (or vector) that corresponds to its “center of mass”

K-means Clustering
cluster centroid

- The key to understanding K-means clustering is to understand the idea of a cluster centroid
- Given a cluster, you can think of its centroid as a point (or vector) that corresponds to its “center of mass”

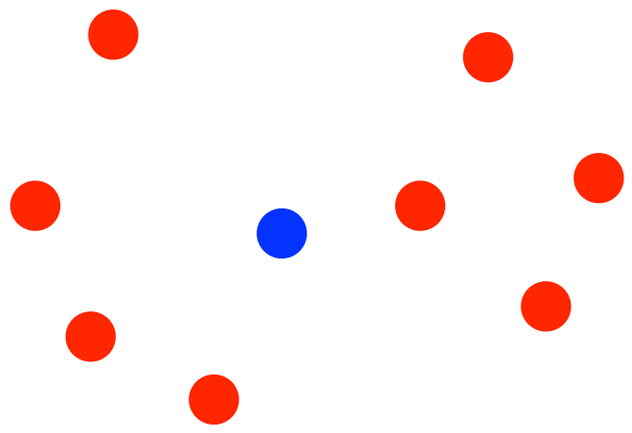


K-means Clustering
cluster centroid

- The key to understanding K-means clustering is to understand the idea of a cluster centroid
- Given a cluster, you can think of its centroid as a point (or vector) that corresponds to its “center of mass”

K-means Clustering
cluster centroid

- The key to understanding K-means clustering is to understand the idea of a cluster centroid
- Given a cluster, you can think of its centroid as a point (or vector) that corresponds to its “center of mass”



K-means Clustering

cluster centroid

docs assigned to cluster 1

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10
1	0	1	0	1	0	0	1	1	0
0	1	0	1	1	0	1	1	0	0
0	1	0	1	1	0	1	0	0	0
0	0	1	0	1	1	0	1	1	1
0	0	1	0	1	1	0	1	1	1
1	1	0	1	1	0	0	1	0	1

cluster 1 centroid

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10
?	?	?	?	?	?	?	?	?	?

K-means Clustering

cluster centroid

docs assigned to cluster 1

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10
1	0	1	0	1	0	0	1	1	0
0	1	0	1	1	0	1	1	0	0
0	1	0	1	1	0	1	0	0	0
0	0	1	0	1	1	0	1	1	1
0	0	1	0	1	1	0	1	1	1
1	1	0	1	1	0	0	1	0	1

cluster 1 centroid (average!)

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_10
0.33	0.50	0.50	0.50	1.00	0.33	0.33	0.83	0.50	0.50

K-means Clustering

cluster centroid

- For each dimension i , set:

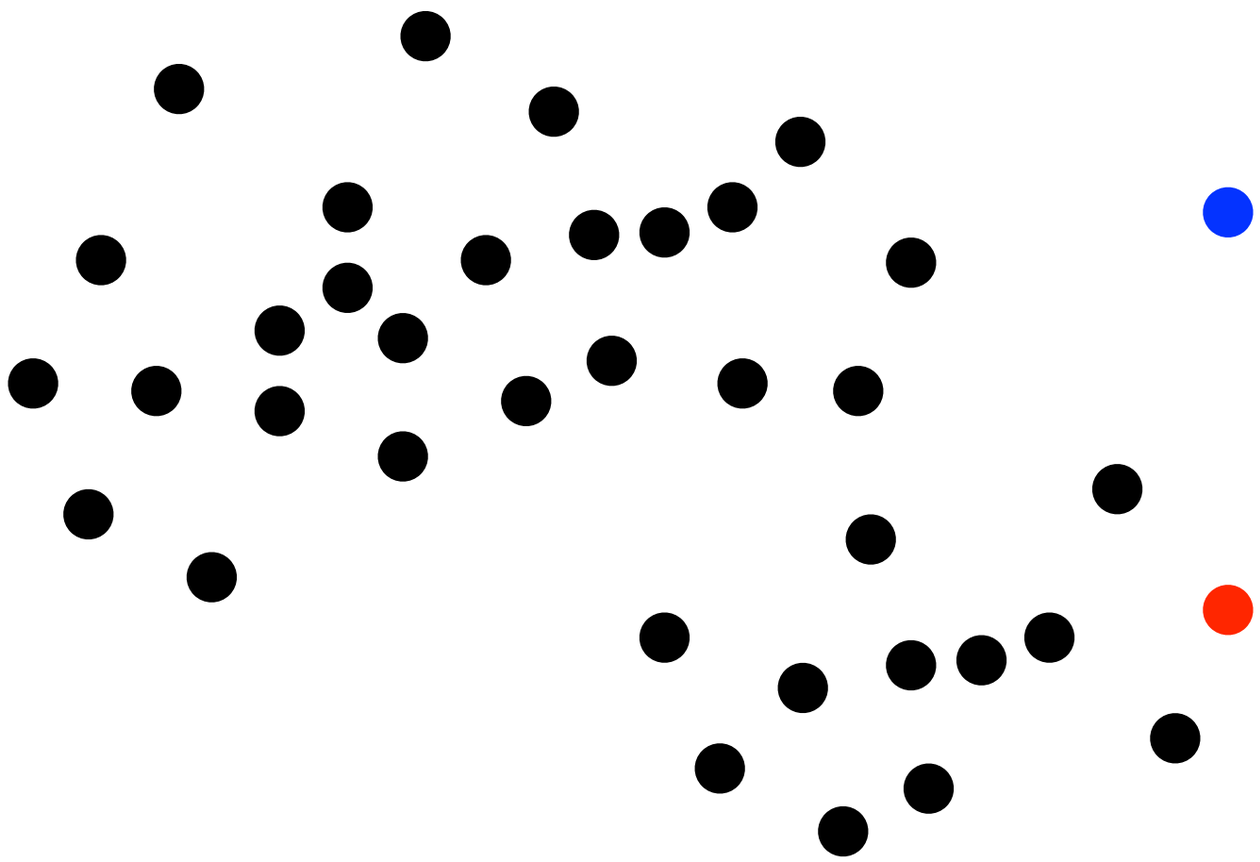
$$c_i = \frac{1}{|C|} \sum_{d \in C} d_i$$

K-means Clustering

- Input: number of desired clusters K
- Output: assignment of documents to K clusters
- Algorithm:
  - Step 1: randomly select K documents (seeds)
  - Step 2: assign each document to its nearest seed
  - Step 3: compute all K cluster centroids
  - Step 4: re-assign each document to its nearest centroid
  - Step 5: re-compute all K cluster centroids
  - Step 6: repeat steps 4 and 5 until terminating condition

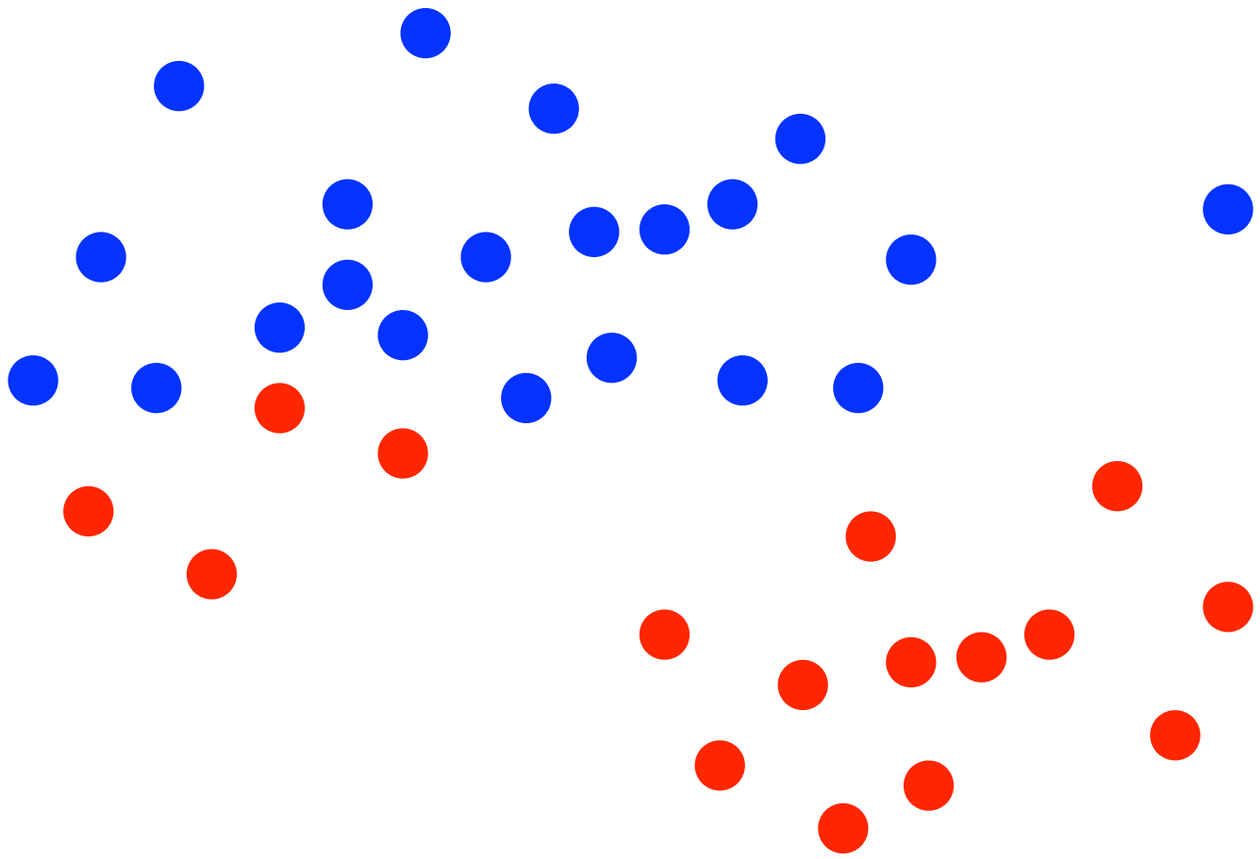
K-means Clustering

- Step 1: randomly select K documents (seeds)



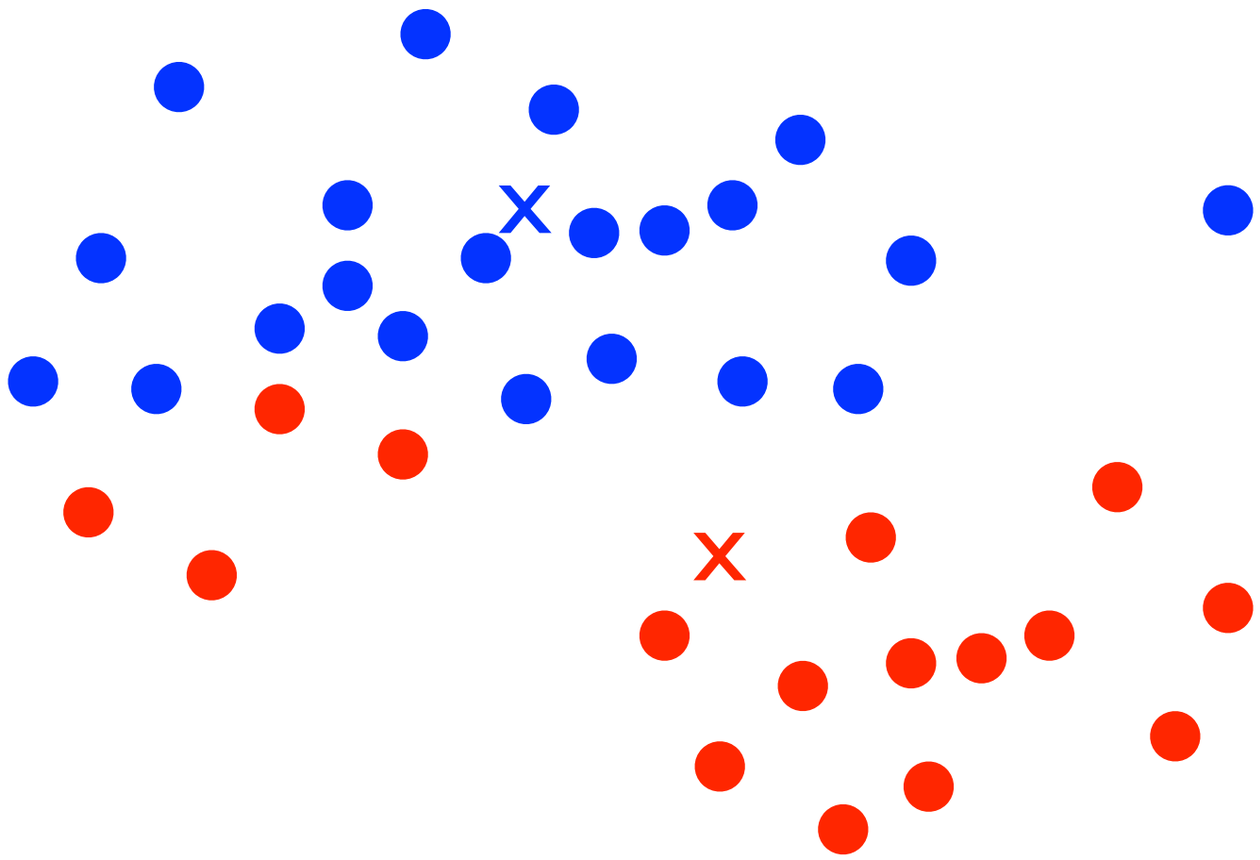
K-means Clustering

- Step 2: assign each document to its nearest seed



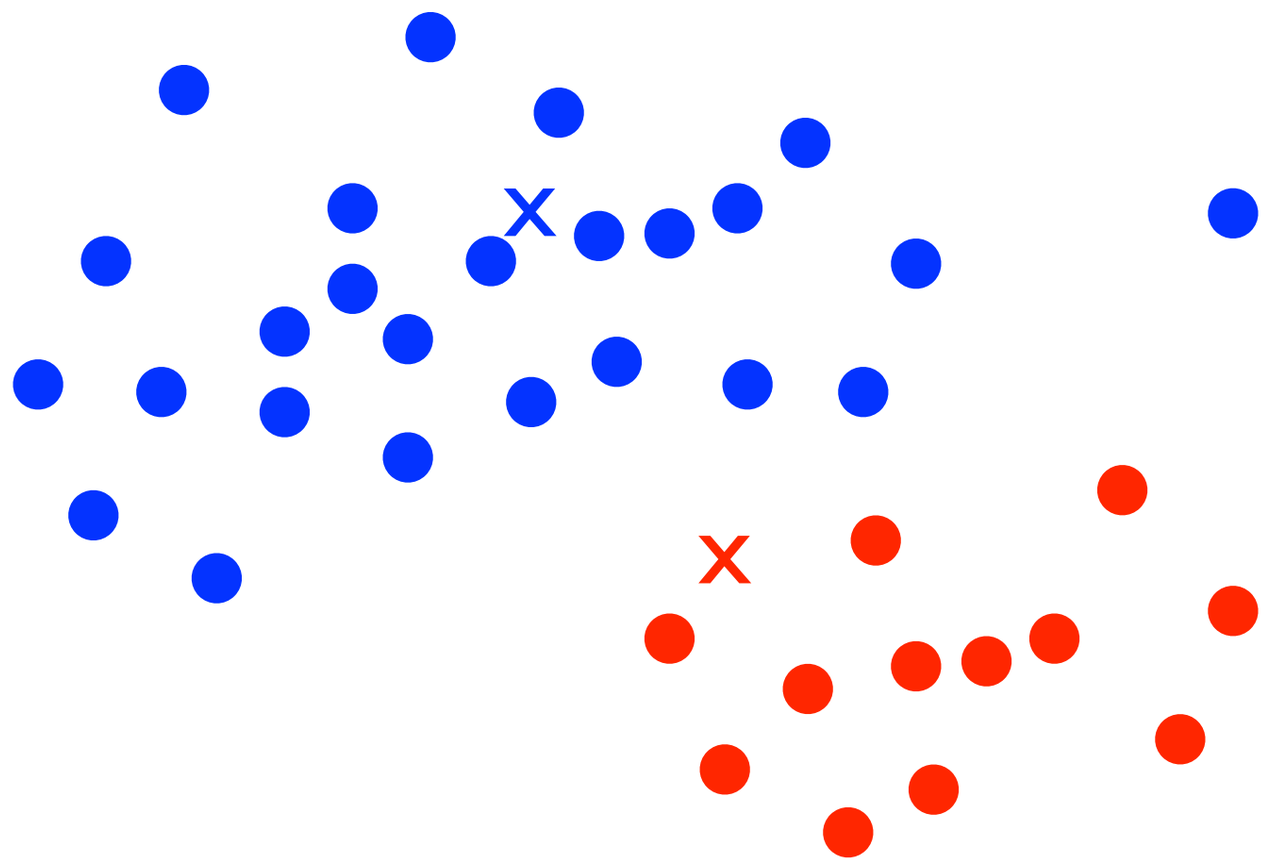
K-means Clustering

- Step 3: compute all K cluster centroids



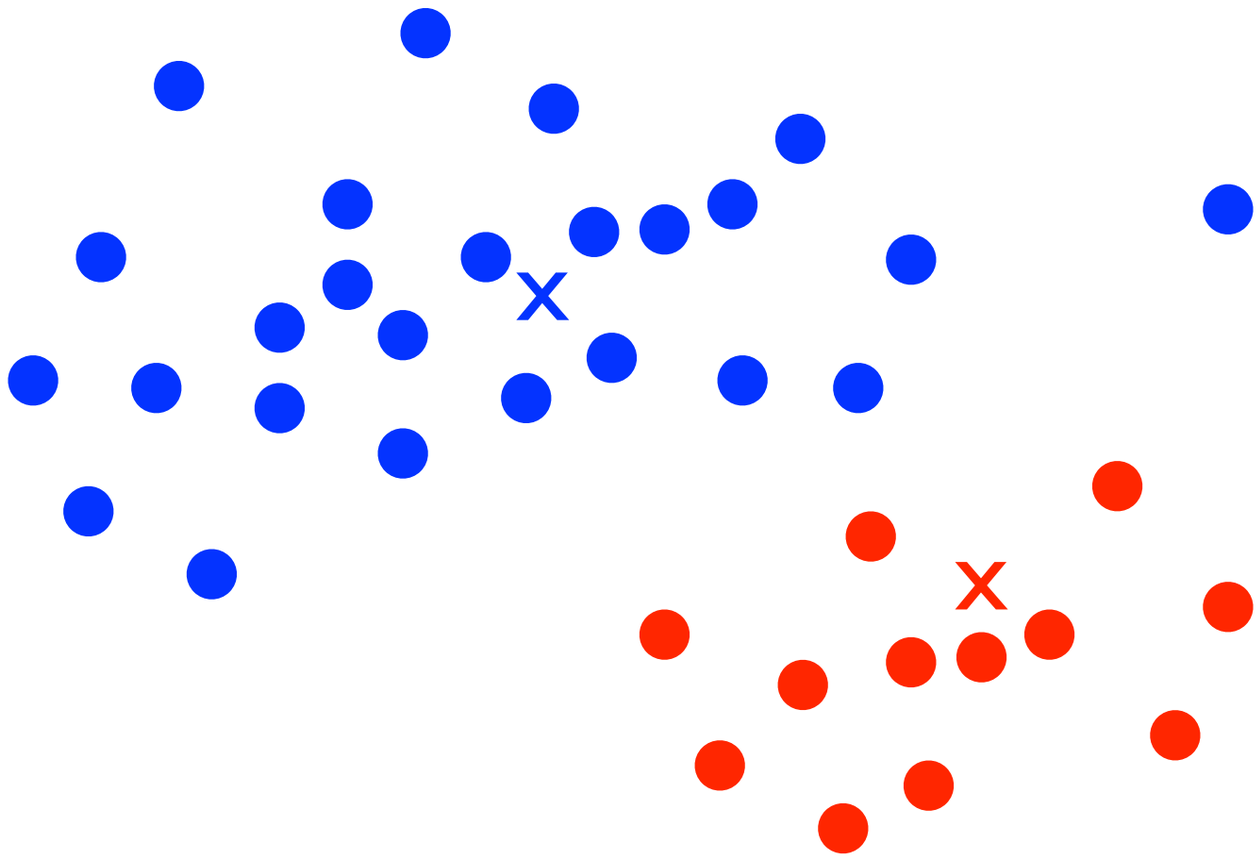
K-means Clustering

- Step 4: re-assign each document to its nearest centroid



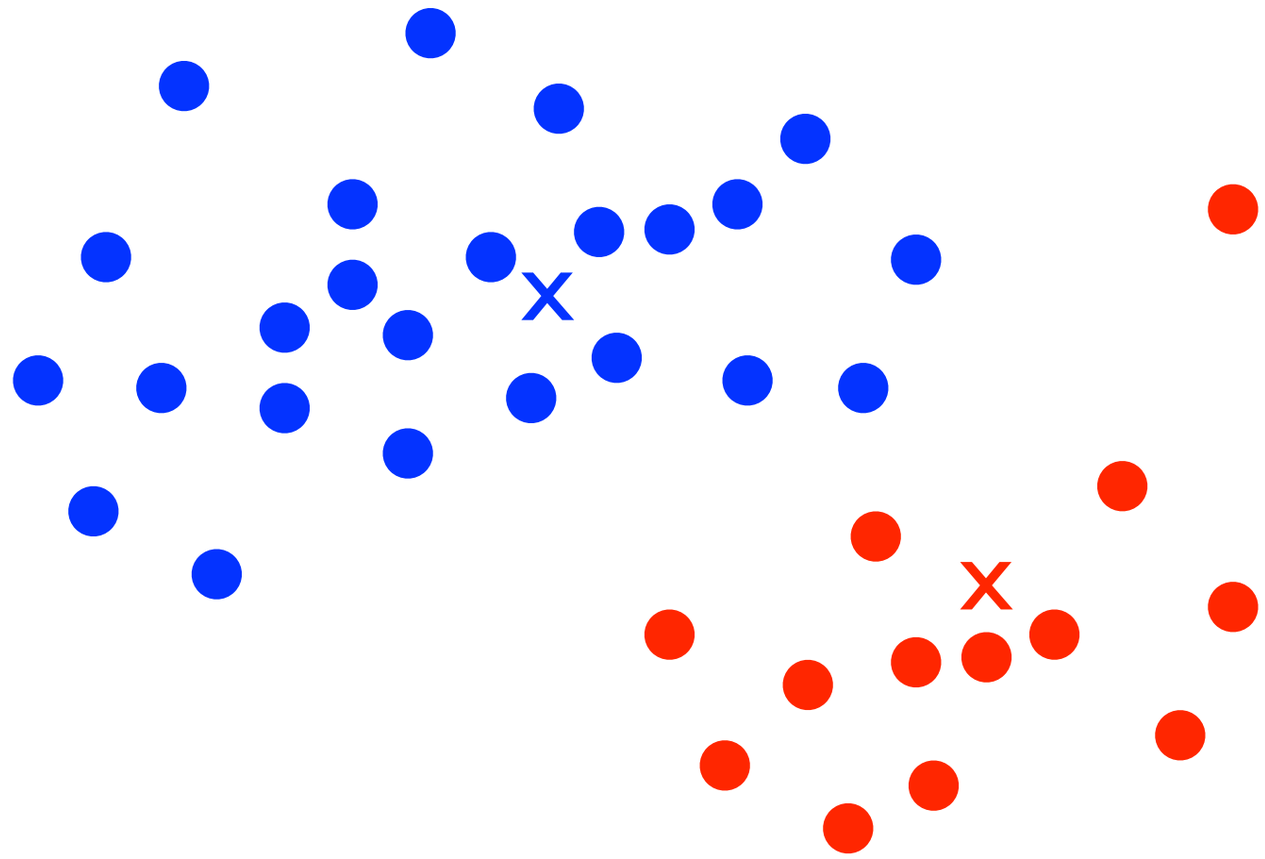
K-means Clustering

- Step 4: re-compute all K cluster centroids



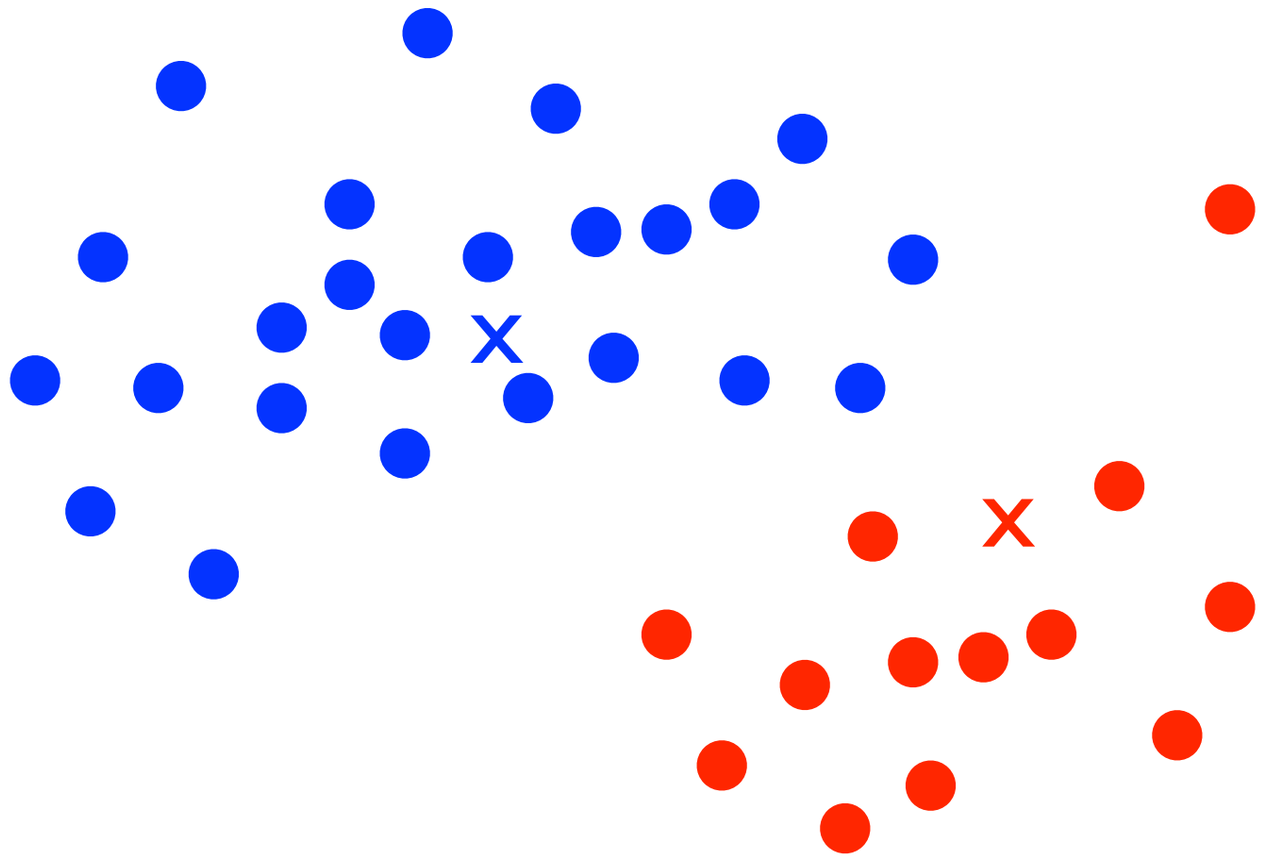
K-means Clustering

- Step 5: re-assign each document to its nearest centroid



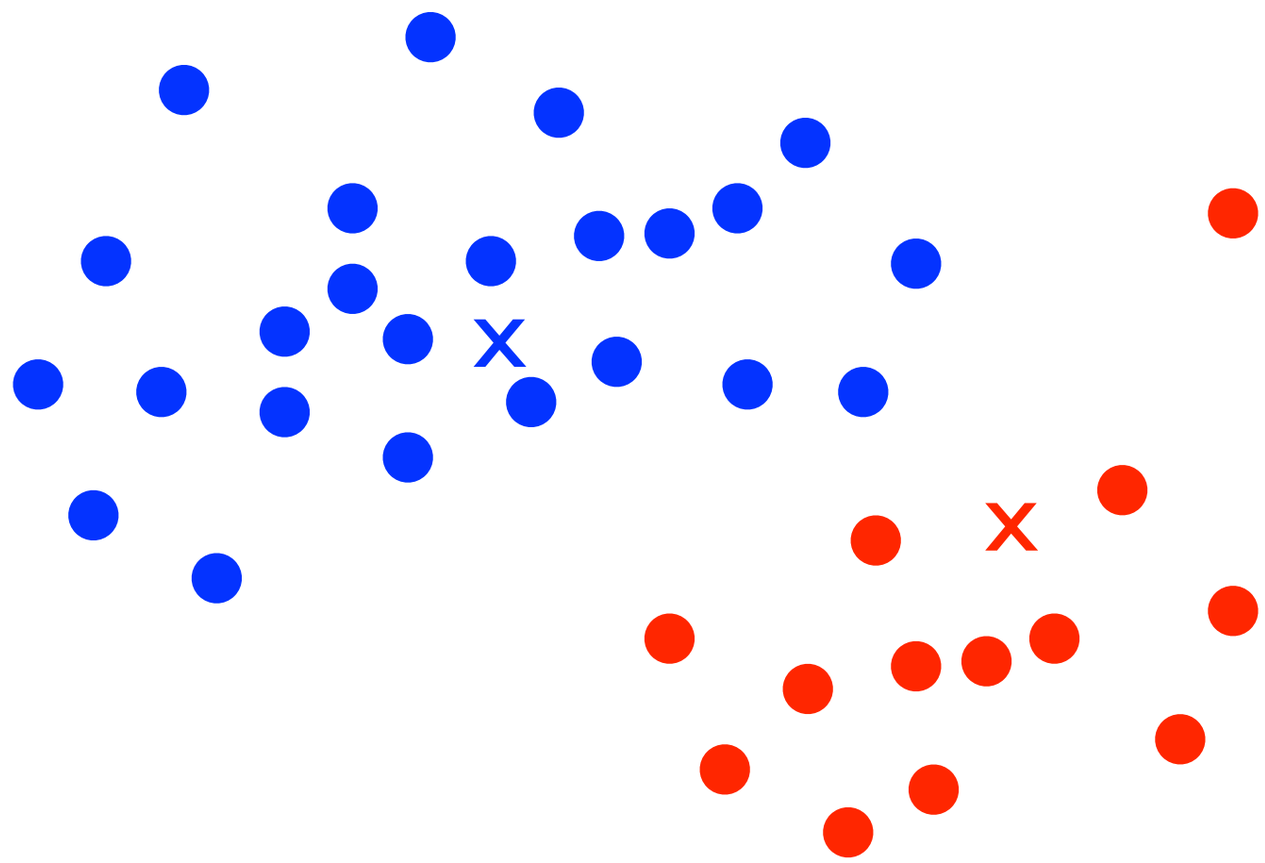
K-means Clustering

- Step 4: re-compute all K cluster centroids



K-means Clustering

- Step 5: re-assign each document to its nearest centroid



K-means Clustering

- Input: number of desired clusters K
- Output: assignment of documents to K clusters
- Algorithm:
  - Step 1: randomly select K documents (seeds)
  - Step 2: assign each document to its nearest seed
  - Step 3: compute all K cluster centroids
  - Step 4: re-assign each document to its nearest centroid
  - Step 5: re-compute all K cluster centroids
  - Step 6: repeat steps 4 and 5 until terminating condition

Outline: Predictive and Exploratory Analysis

Concepts, Instances, and Features

Human Annotation

Text Representation

Learning Algorithms

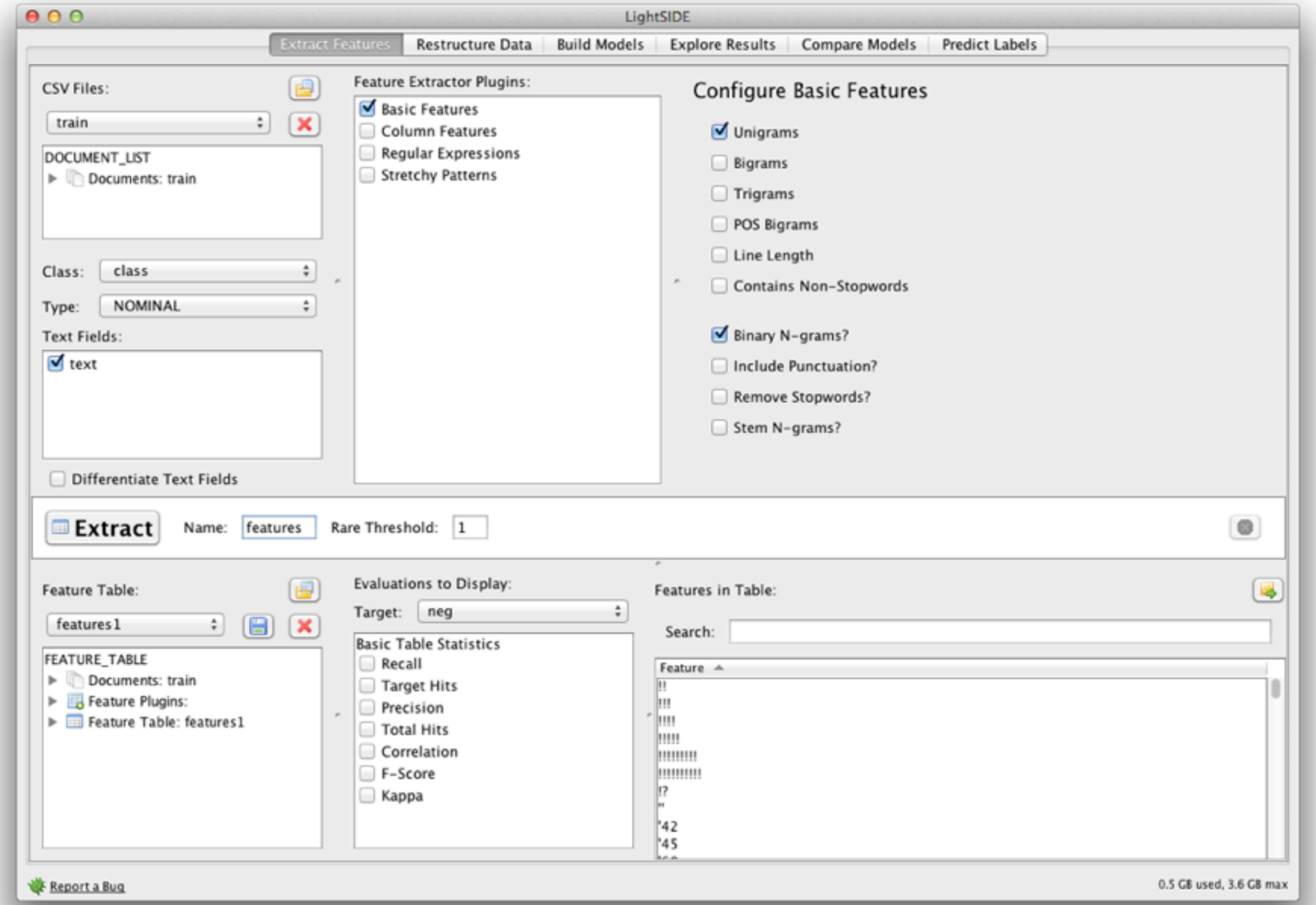
Evaluation metrics

Experimentation

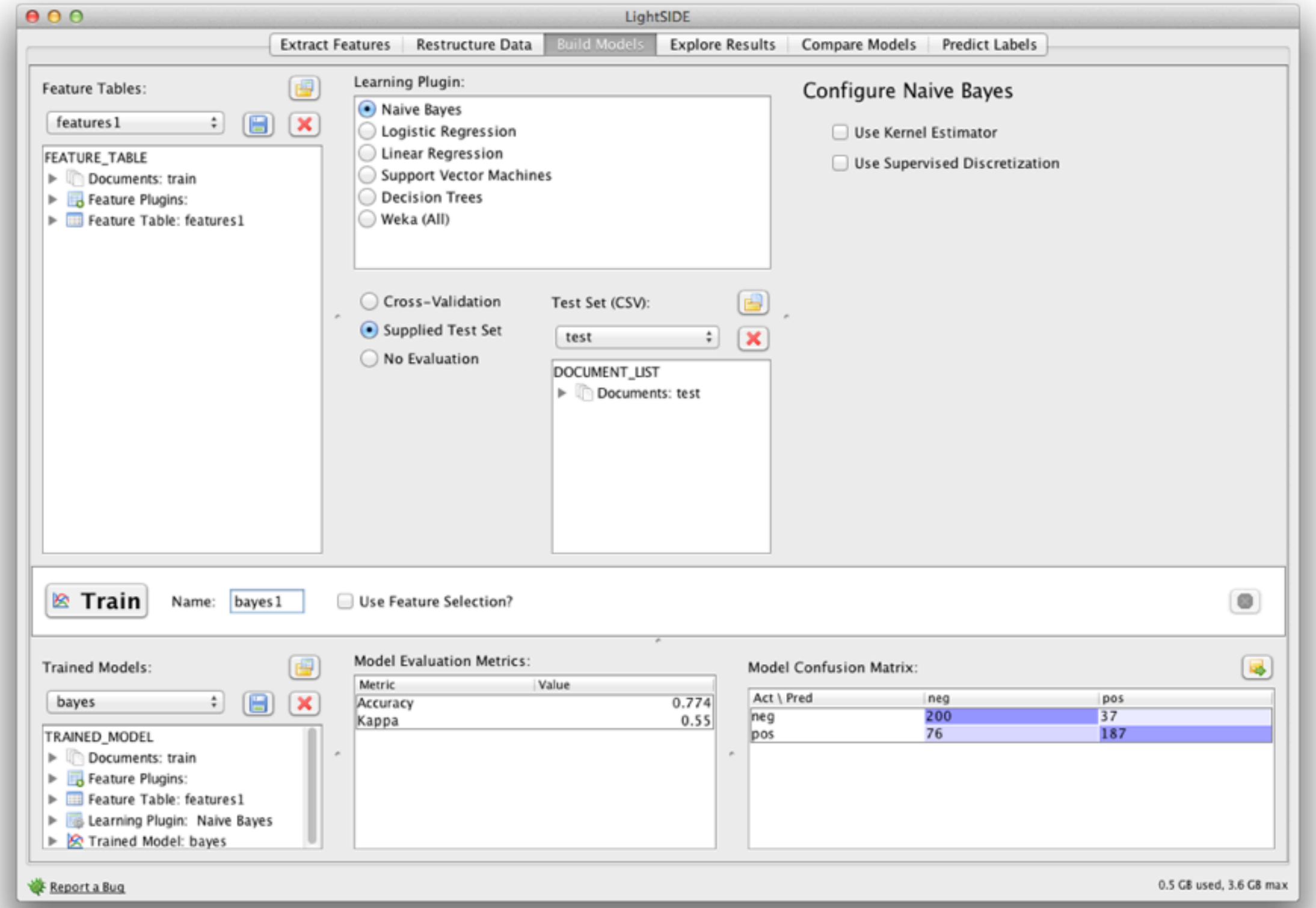
Clustering

Hands-on Exercise

LightSIDE


LightSIDE
Extract Features
Restructure Data
Build Models
Explore Results
Compare Models
Predict Labels
CSV Files:
train
DOCUMENT_LIST
Documents: train
Class:
class
Type:
NOMINAL
Text Fields:
text
Differentiate Text Fields
Feature Extractor Plugins:
Basic Features
Column Features
Regular Expressions
Stretchy Patterns
Configure Basic Features
Unigrams
Bigrams
Trigrams
POS Bigrams
Line Length
Contains Non-Stopwords
Binary N-grams?
Include Punctuation?
Remove Stopwords?
Stem N-grams?
Extract
Name:
features
Rare Threshold:
1
Feature Table:
features1
FEATURE_TABLE
Documents: train
Feature Plugins:
Feature Table: features1
Evaluations to Display:
Target:
neg
Basic Table Statistics
Recall
Target Hits
Precision
Total Hits
Correlation
F-Score
Kappa
Features in Table:
Search:
Feature
!!
!!!
!!!!
!!!!!
!!!!!!!!!
!!!!!!!!!!
!?
"
'42
'45
Report a Bug
0.5 GB used, 3.6 GB max

LightSIDE


LightSIDE
Extract Features
Restructure Data
Build Models
Explore Results
Compare Models
Predict Labels
Feature Tables:
features1
FEATURE_TABLE
Documents: train
Feature Plugins:
Feature Table: features1
Learning Plugin:
Naive Bayes
Logistic Regression
Linear Regression
Support Vector Machines
Decision Trees
Weka (All)
Configure Naive Bayes
Use Kernel Estimator
Use Supervised Discretization
Cross-Validation
Supplied Test Set
No Evaluation
Test Set (CSV):
test
DOCUMENT_LIST
Documents: test
Train
Name:
bayes1
Use Feature Selection?
Trained Models:
bayes
TRAINED_MODEL
Documents: train
Feature Plugins:
Feature Table: features1
Learning Plugin: Naive Bayes
Trained Model: bayes
Model Evaluation Metrics:
Metric
Value
Accuracy
0.774
Kappa
0.55
Model Confusion Matrix:
Act \ Pred
neg
pos
neg
200
37
pos
76
187
Report a Bug
0.5 GB used, 3.6 GB max