# Instance-Based Learning

Jaime Arguello
INLS 613: Text Data Mining
jarguell@email.unc.edu

October 2, 2017

# 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 | $\boxed{0}$ | 0 | ? |

# Instance-Based Classification

## Motivation

training data

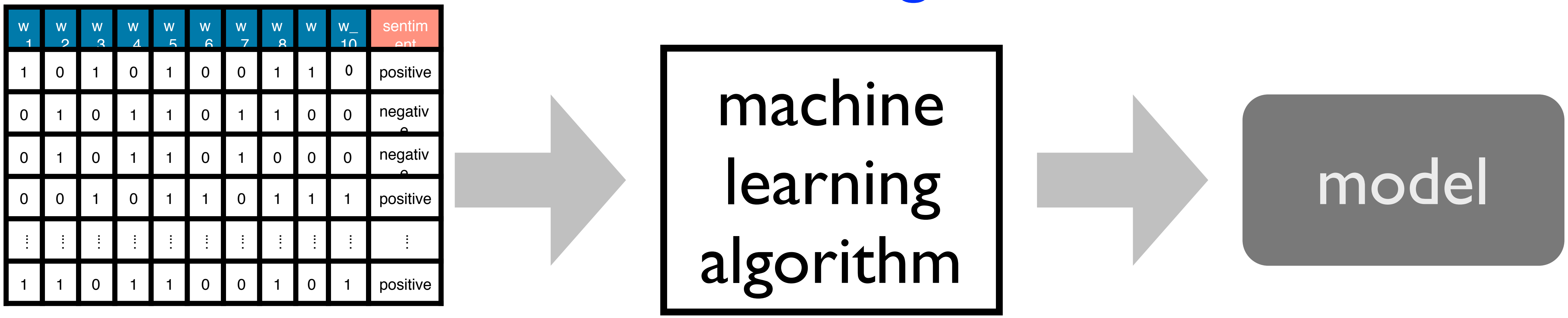
| w_1 | w_2 | w_3 | w_4 | w_5 | w_6 | w_7 | w_8 | w_9 | w_10 | sentiment |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | positive |
| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | negative |
| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | negative |
| 0 | 0 | 1 | 0 | 1 |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

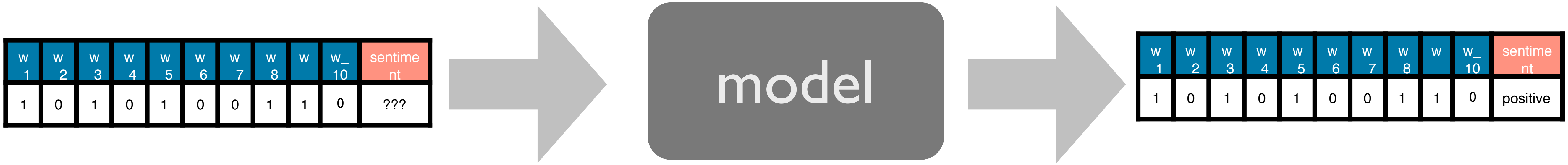


| w 1 | w 2 | w 3 | w 4 | w 5 | w 6 | w 7 | w 8 | w | w_ 10 | sentiment |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | positive |
| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | negative |
| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | negative |
| 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | positive |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| 1 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | positive |

machine learning algorithm

labeled examples

## testing



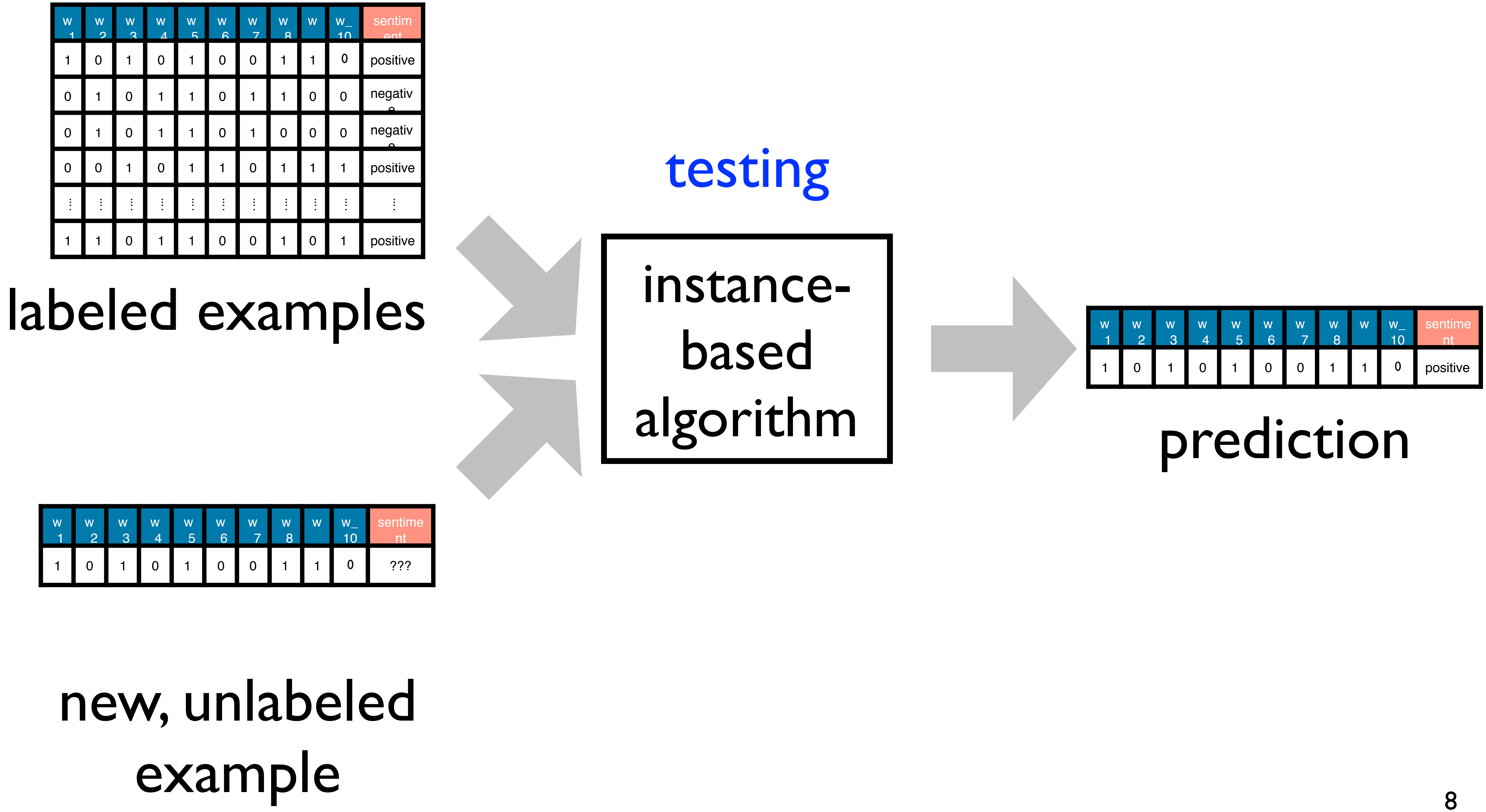
| w 1 | w 2 | w 3 | w 4 | w 5 | w 6 | w 7 | w 8 | w | w_ 10 | sentiment |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | ??? |

| w 1 | w 2 | w 3 | w 4 | w 5 | w 6 | w 7 | w 8 | w | w_ 10 | sentiment |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | positive |

new, unlabeled
example

prediction

# Instance-based Classification



| w 1 | w 2 | w 3 | w 4 | w 5 | w 6 | w 7 | w 8 | w | 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

| w 1 | w 2 | w 3 | w 4 | w 5 | w 6 | w 7 | w 8 | w | 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 | 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

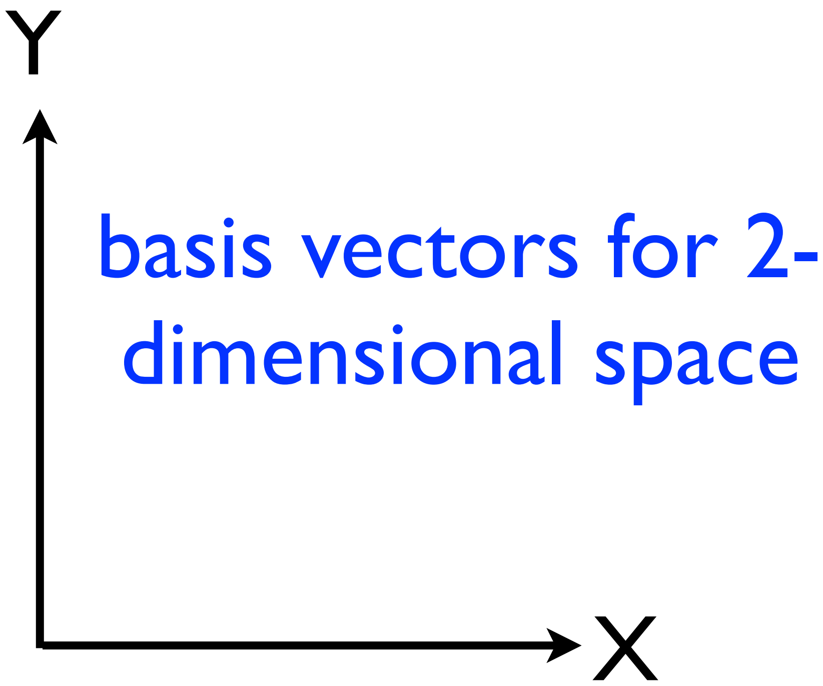
# Instance-based Classification

- Assumption: instances with similar feature values should have the same target label
- Necessary Ingredients:
  - a similarity/distance metric: a measure of similarity between instances
  - an averaging technique: a way of combining the labels from the most similar training instances

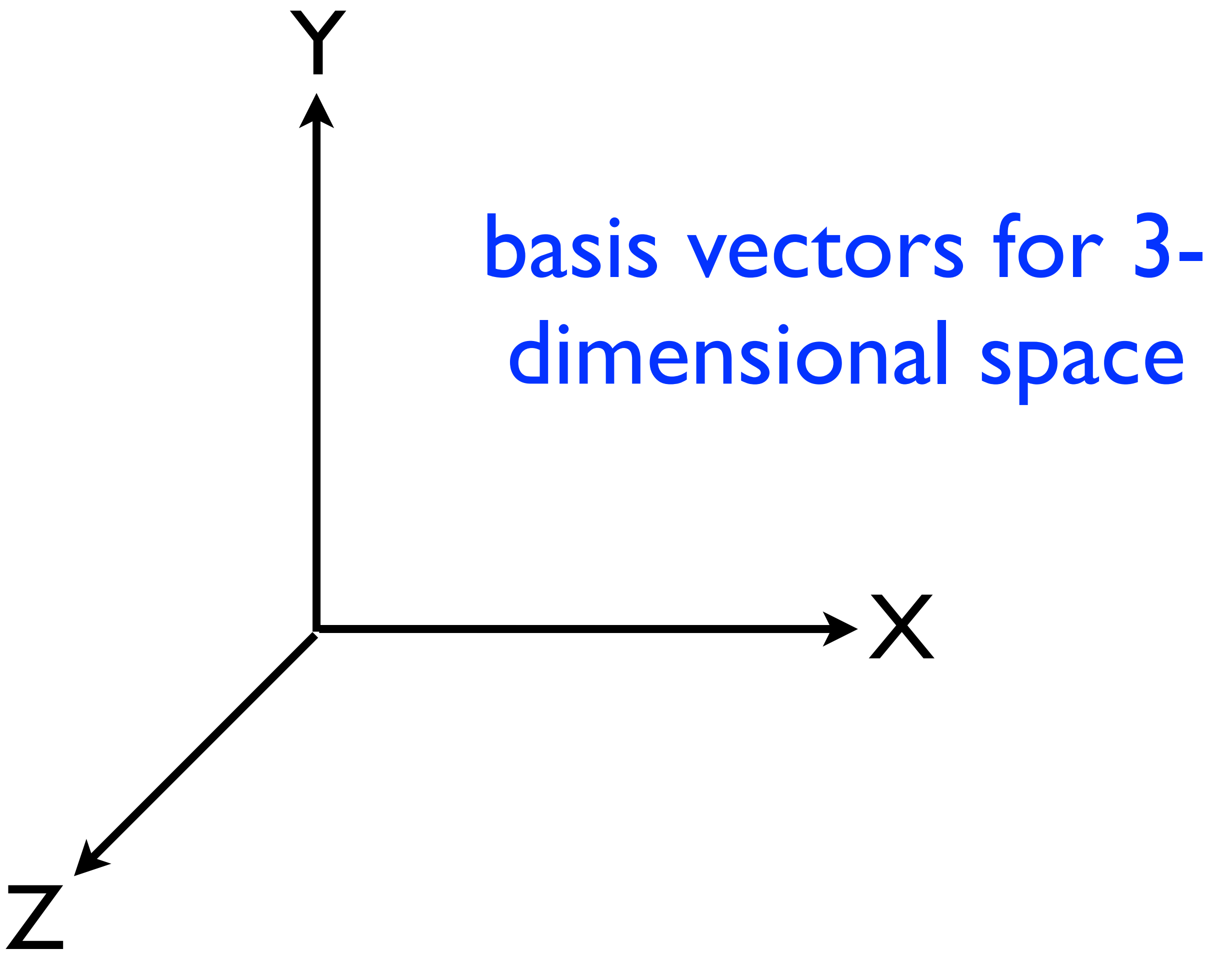
# Vector Space

# 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



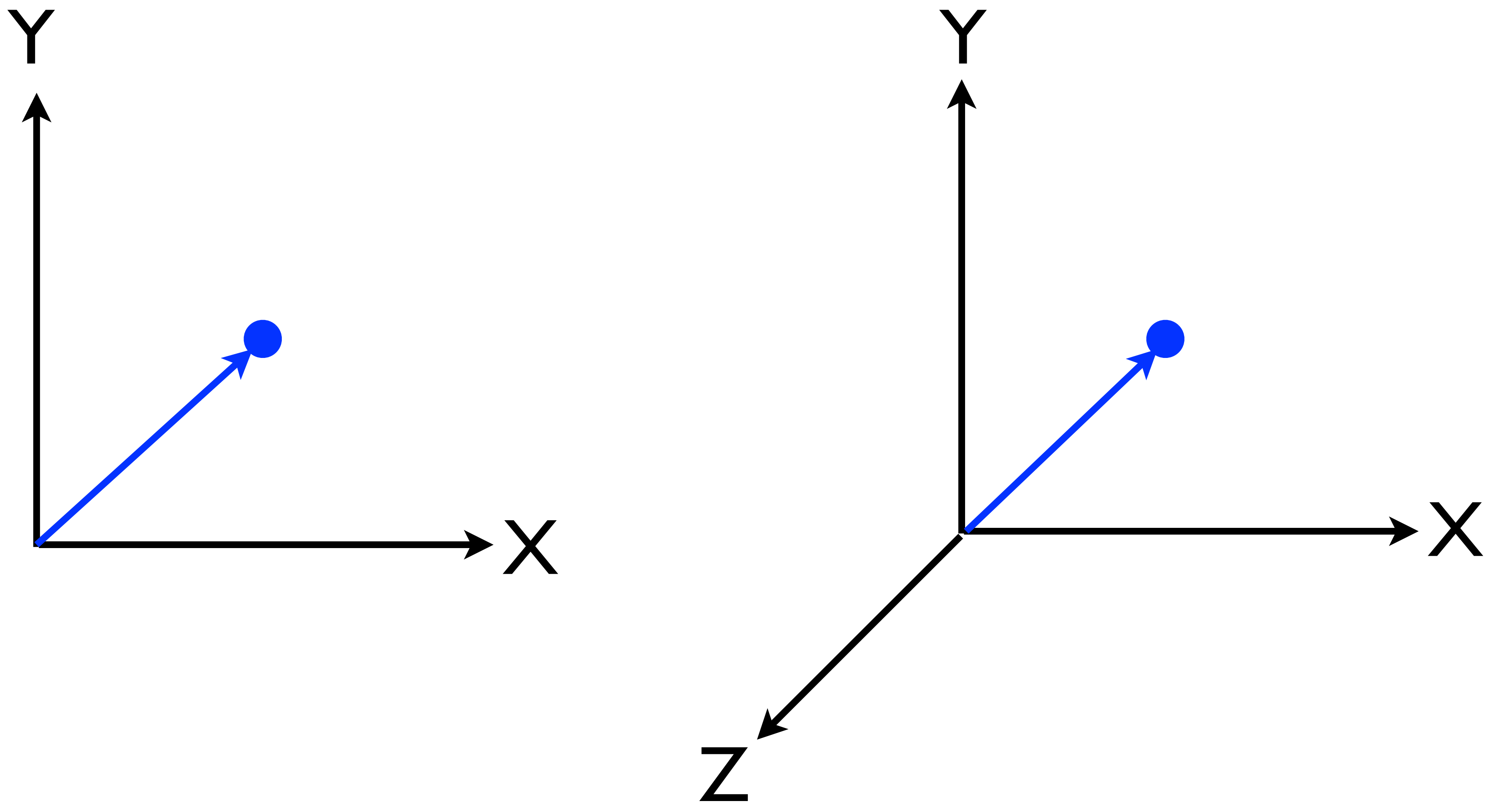
basis vectors for 2-dimensional space



basis vectors for 3-dimensional space

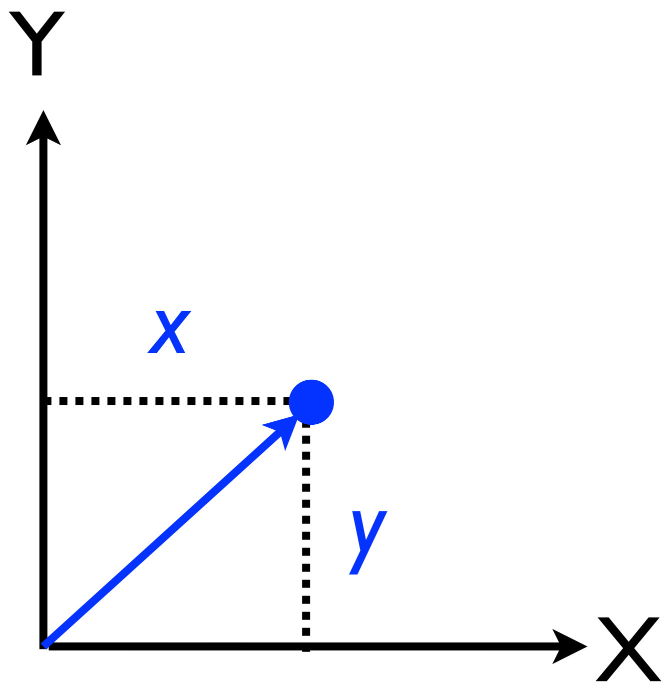
# What is a Vector?

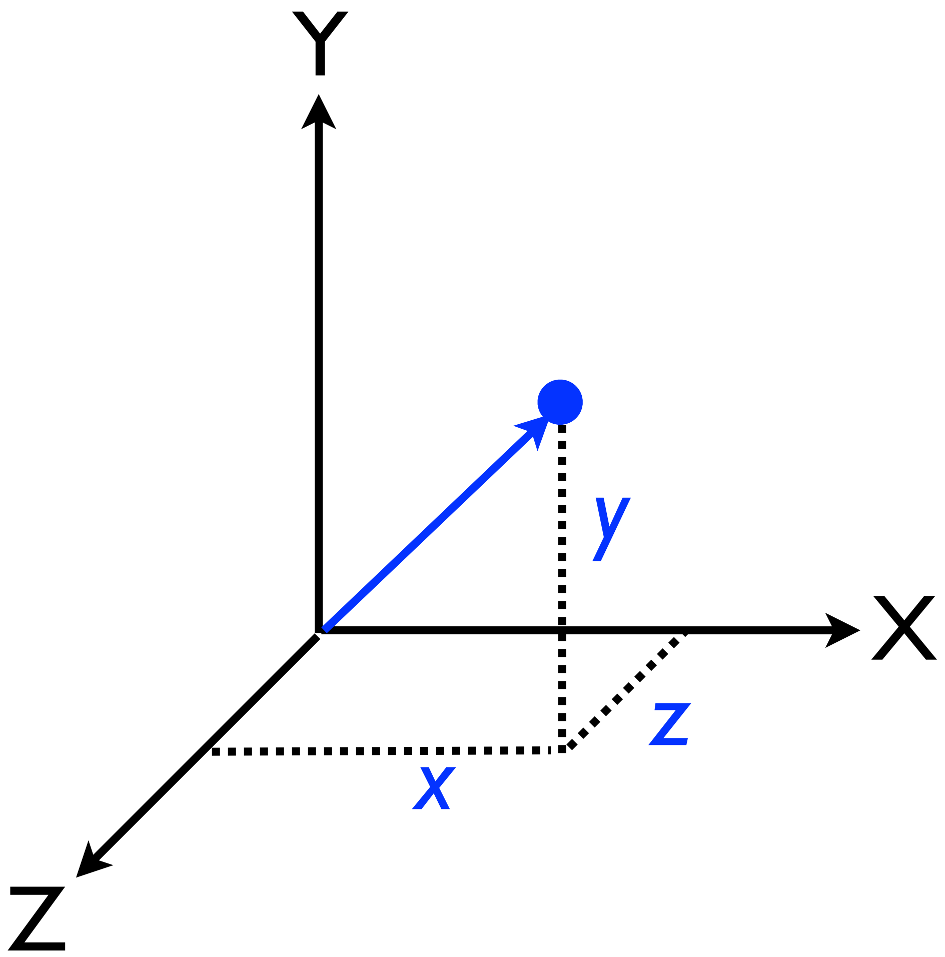
- A vector is a point in a vector space and has length (from the origin to the point) and direction



# What is a Vector?

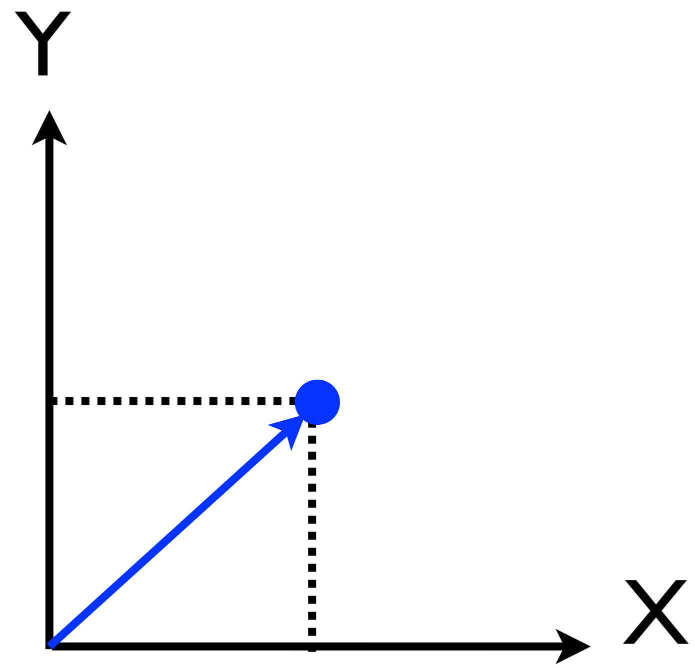
- A 2-dimensional vector can be written as $[x,y]$
- A 3-dimensional vector can be written as $[x,y,z]$



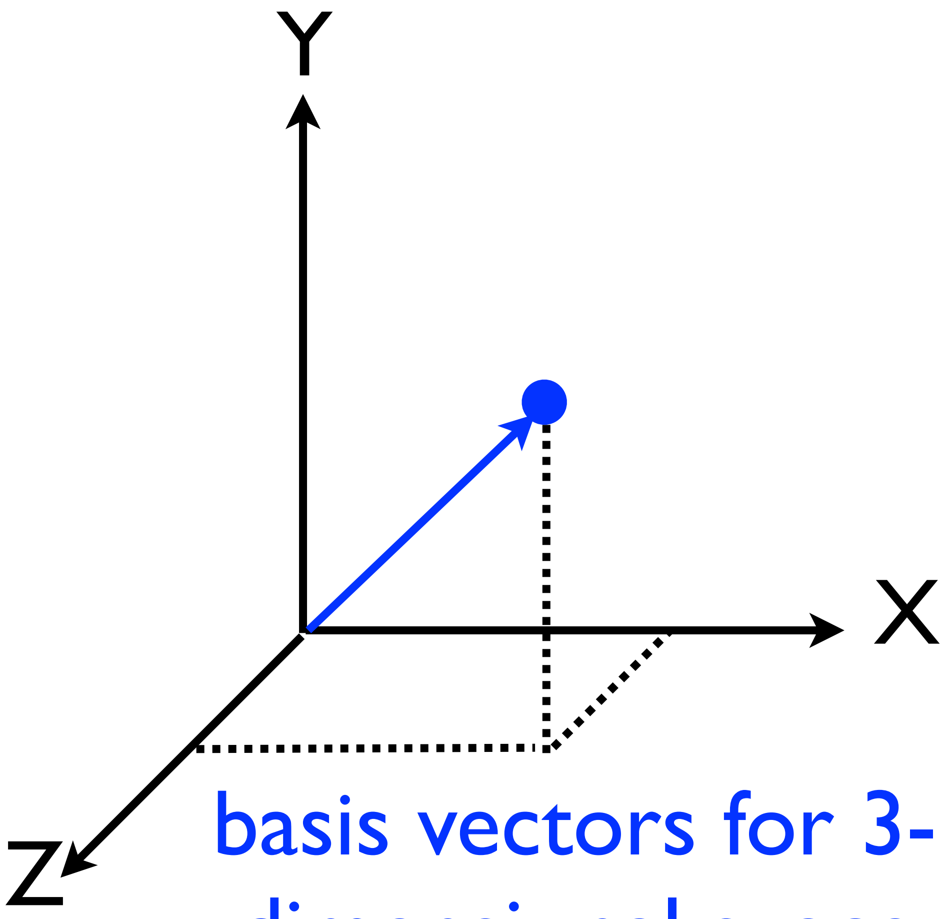


# What is a Vector Space?

- The basis vectors are linearly independent because knowing a vector's value along one dimension doesn't say anything about its value along another dimension



basis vectors for 2-dimensional space



basis vectors for 3-dimensional space

# 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

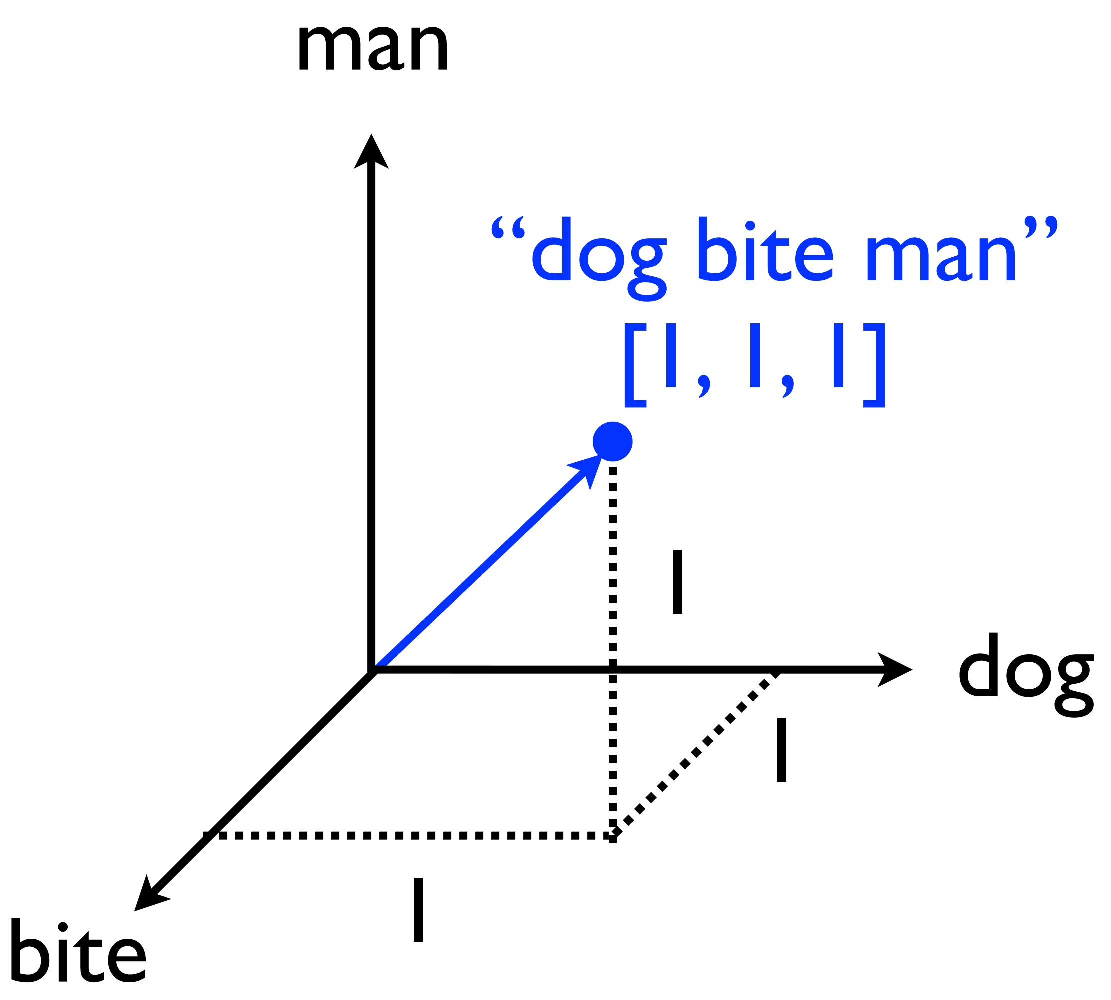
- Let V denote the set of features in our feature representation
- Any arbitrary instance can be represented as a vector in |V|-dimensional space
- For simplicity, let's assume three features: dog, bite, man (i.e., |V| = 3)
- Why? Because it's easy to visualize 3-D space

# 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 |

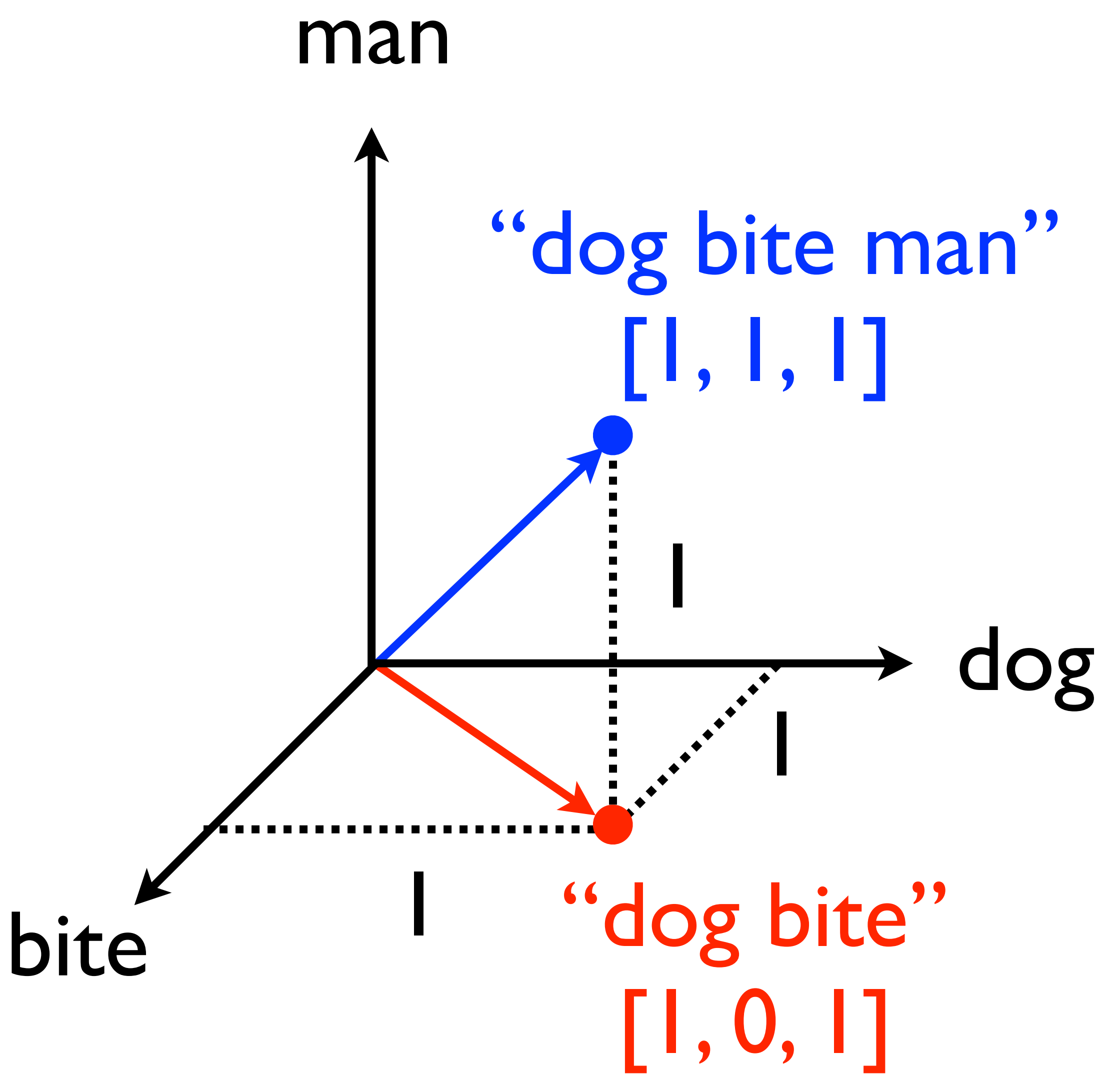


# Vector Space Representation

## 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 |

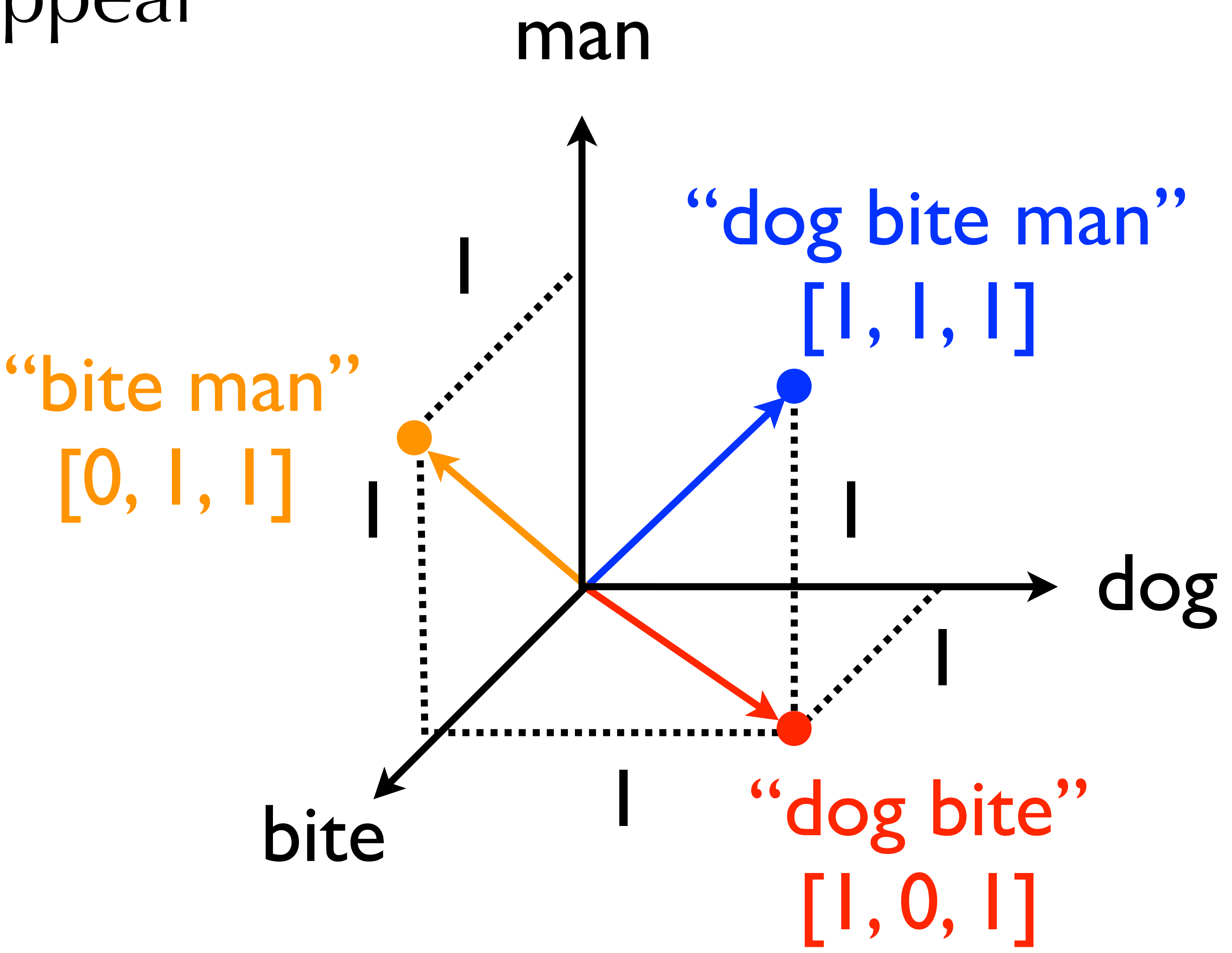


# Vector Space Representation

## with binary weights

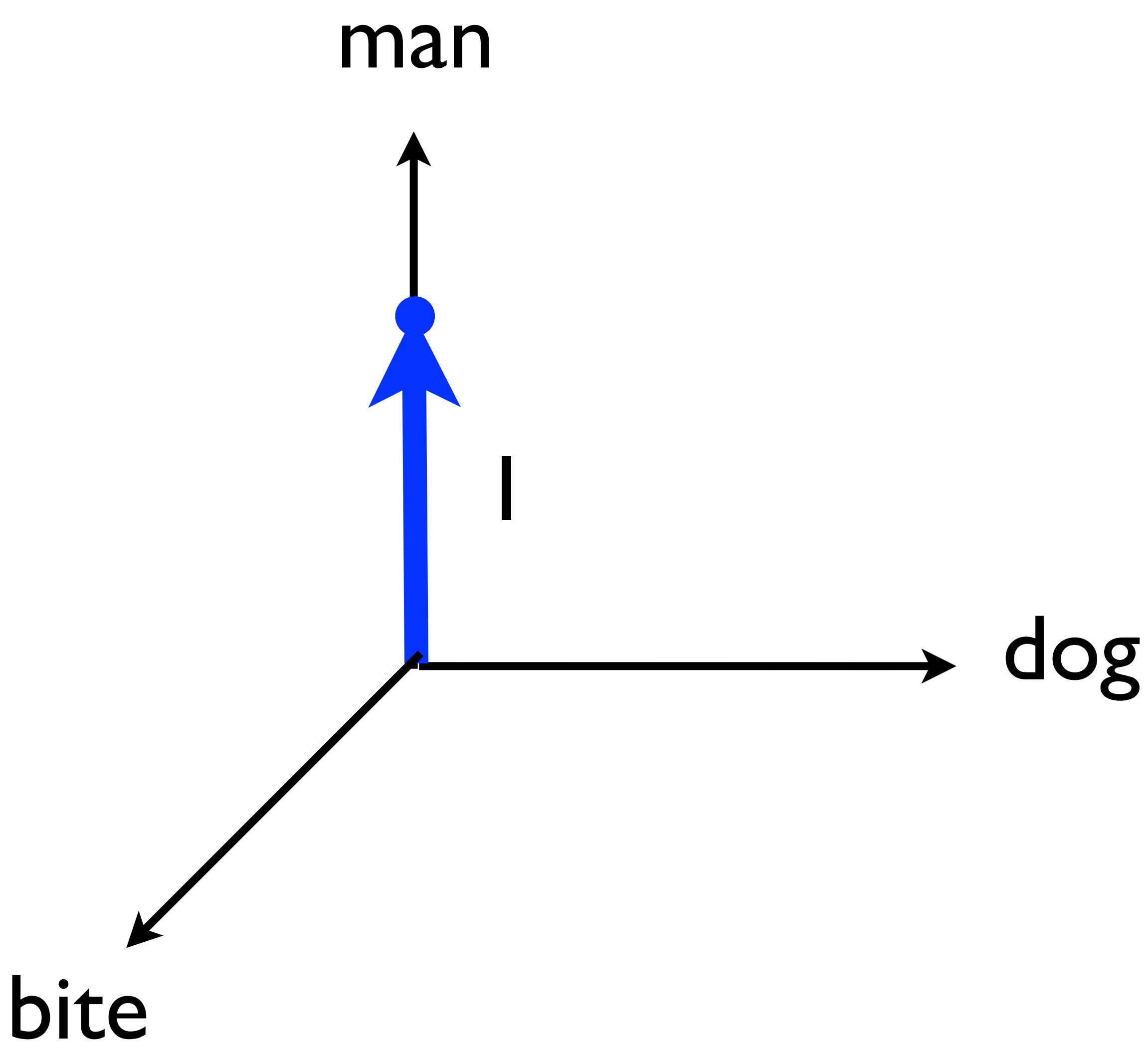
- 1 = the term appears at least once
- 0 = the term does not appear

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



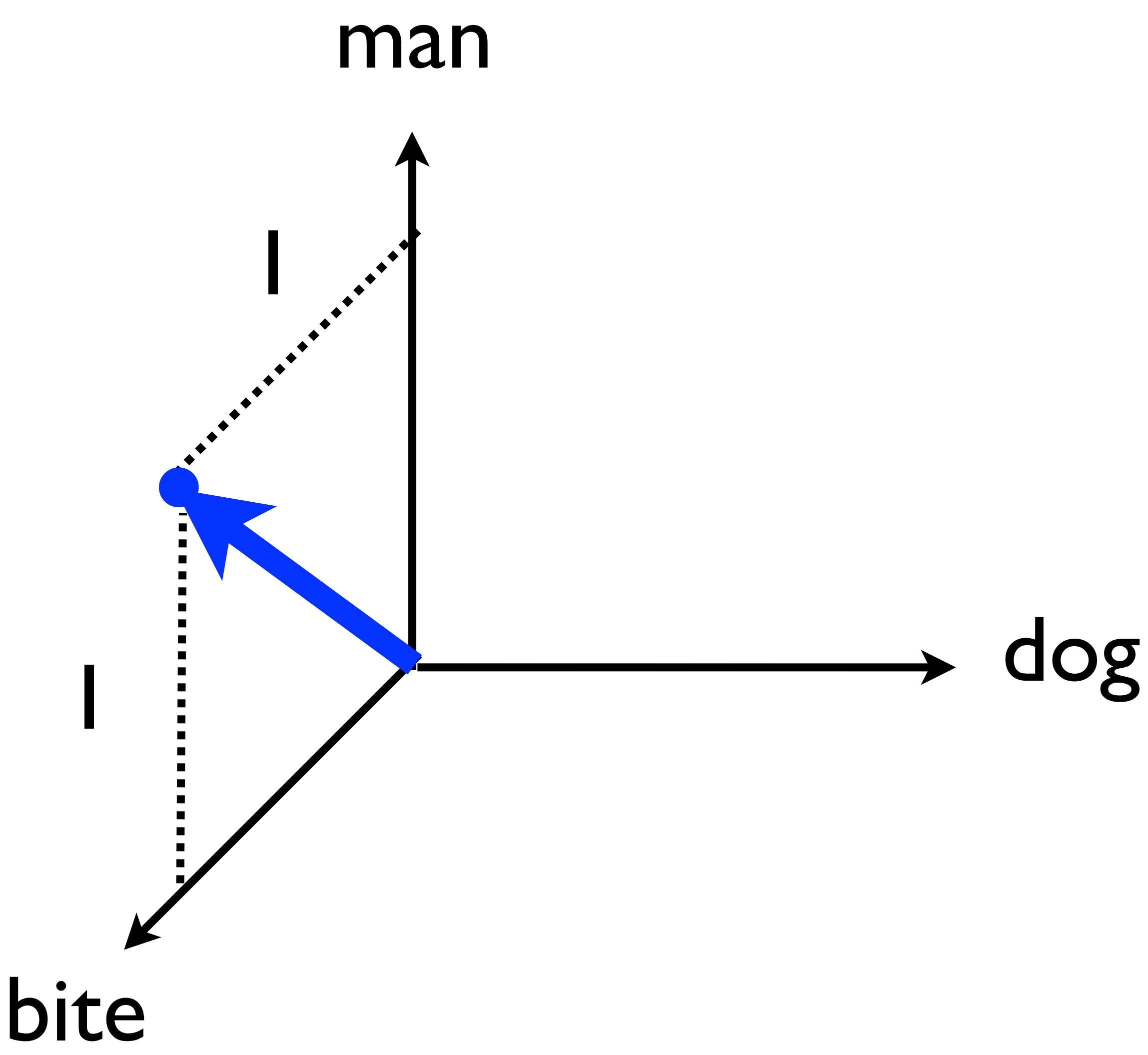
# Vector Space Representation
## with binary weights

- What span(s) of text does this vector represent?



# Vector Space Representation
## with binary weights

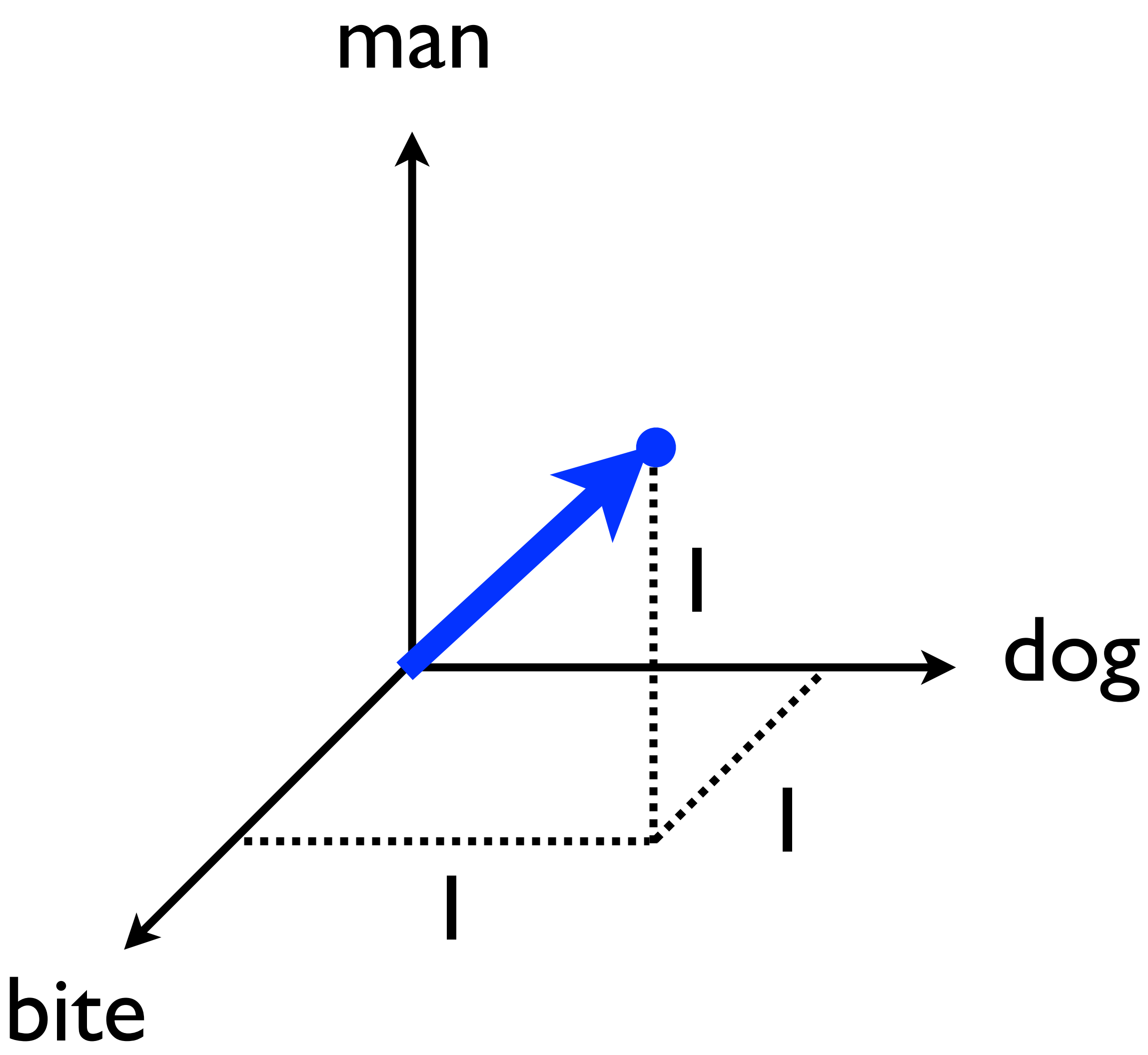
- What span(s) of text does this vector represent?



# Vector Space Representation

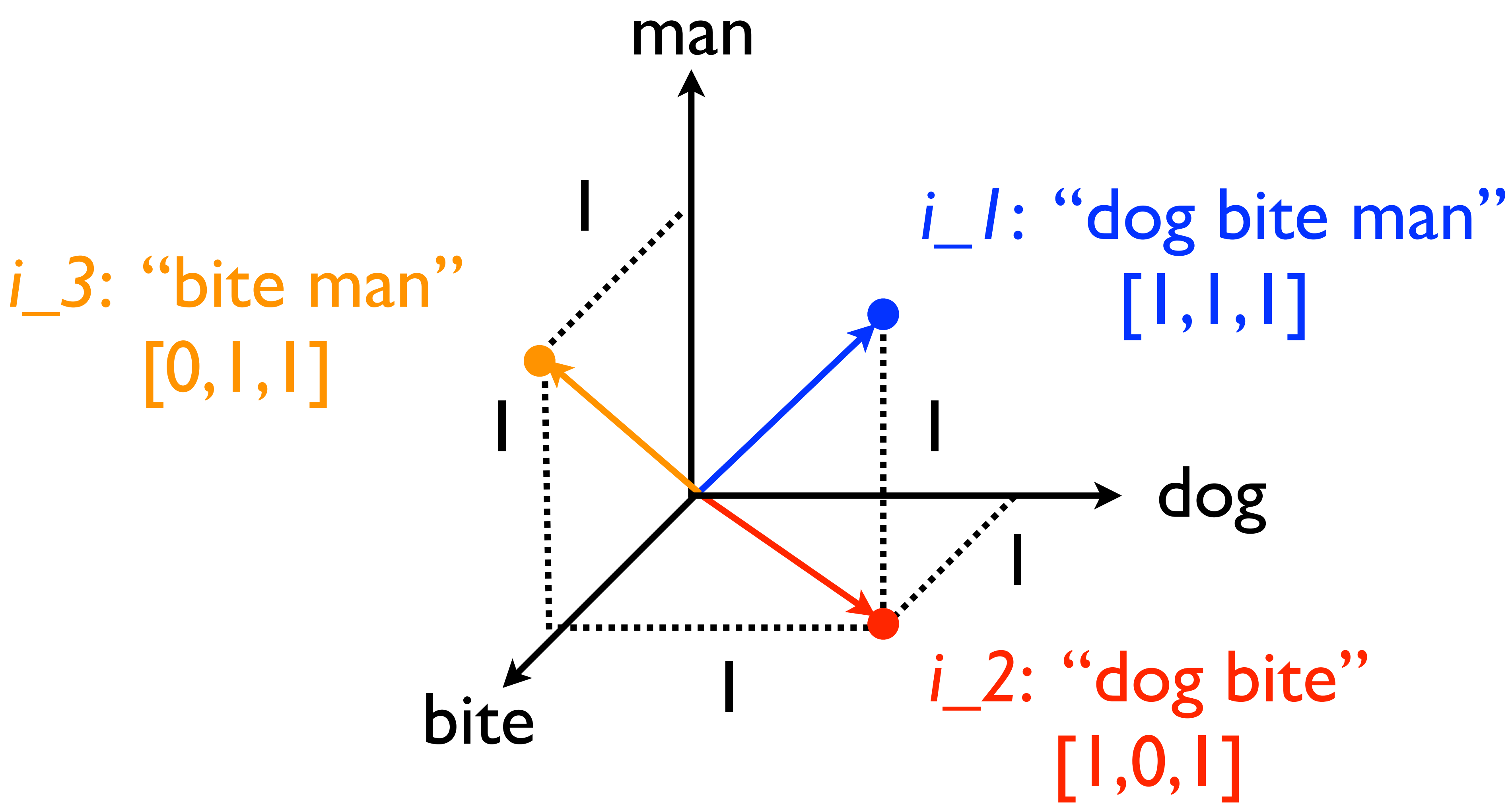
## with binary weights

- What span(s) of text does this vector represent?



# Vector Space Representation
## with binary weights

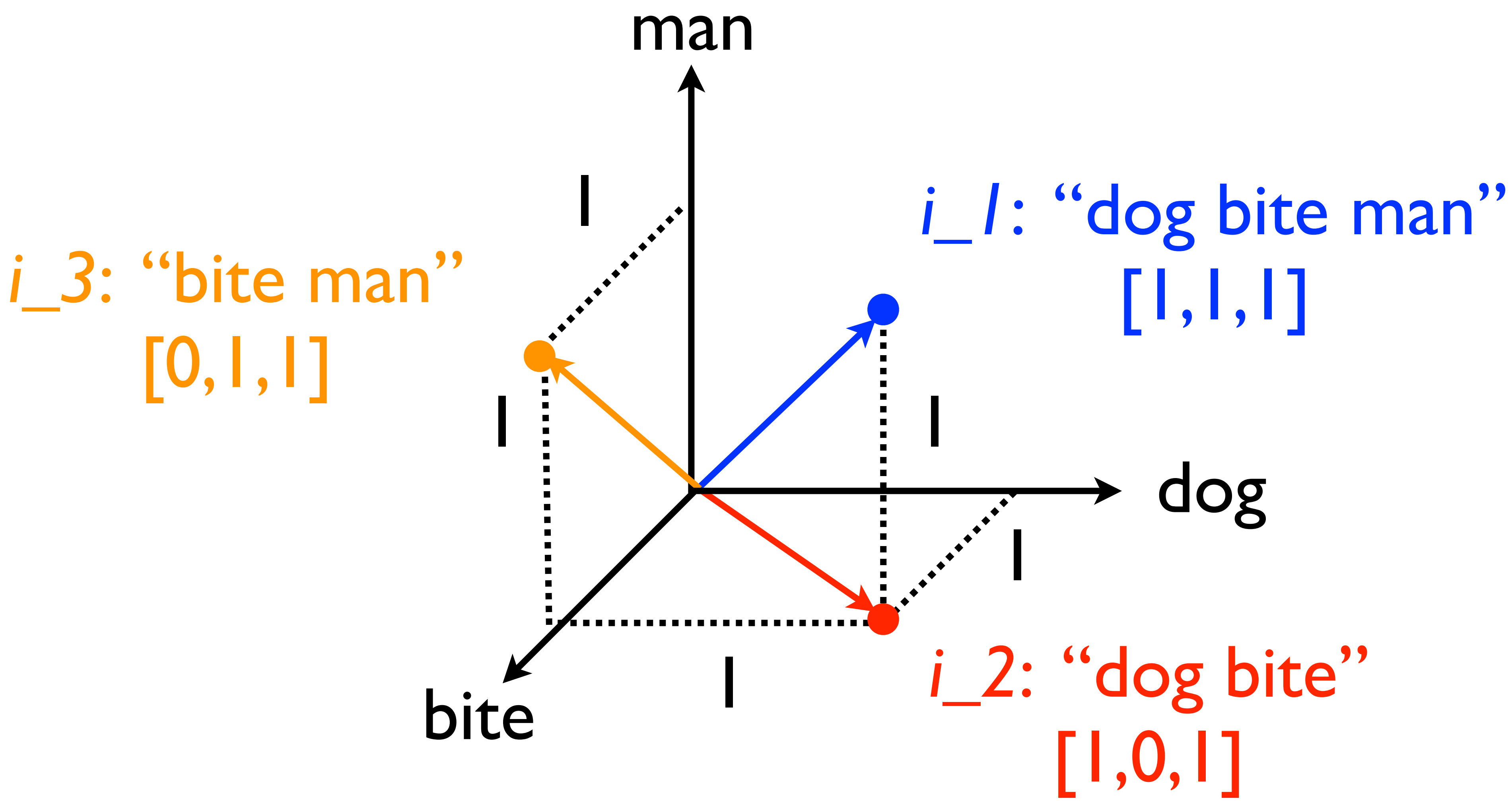
- Any arbitrary span of text can be represented as a vector in |V|-dimensional space



# 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 |
| $D(x,y) = \sqrt{\left(\sum_{i=1}^{\lvert\mathcal{V}\rvert}(x_i - y_i)^2\right)}$ | | | 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 |
| $D(x,y) = \sqrt{\left(\sum_{i=1}^{\lvert\mathcal{V}\rvert}(x_i - y_i)^2\right)}$ | | | 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. |
|---|---|---|
| 1 | a | 22 |
| 2 | rocky | 19 |
| 3 | to | 18 |
| 4 | the | 17 |
| 5 | is | 11 |
| 6 | and | 10 |
| 7 | in | 10 |
| 8 | for | 7 |
| 9 | his | 7 |
| 10 | he | 6 |
| 11 | adrian | 6 |
| 12 | with | 6 |
| 13 | who | 6 |
| 14 | that | 5 |
| 15 | apollo | 5 |
| 16 | creed | 5 |
| 17 | philadelphia | 5 |
| 18 | has | 4 |
| 19 | pet | 4 |
| 20 | boxing | 4 |
| 21 | up | 4 |
| 22 | an | 4 |
| 23 | boxer | 4 |
| 24 | s | 3 |
| 25 | balboa | 3 |
| 26 | it | 3 |
| 27 | heavyweigh | 3 |
| 28 | champion | 3 |
| 29 | fight | 3 |
| 30 | become | 3 |

# Term-Frequency

## how important is a term?

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

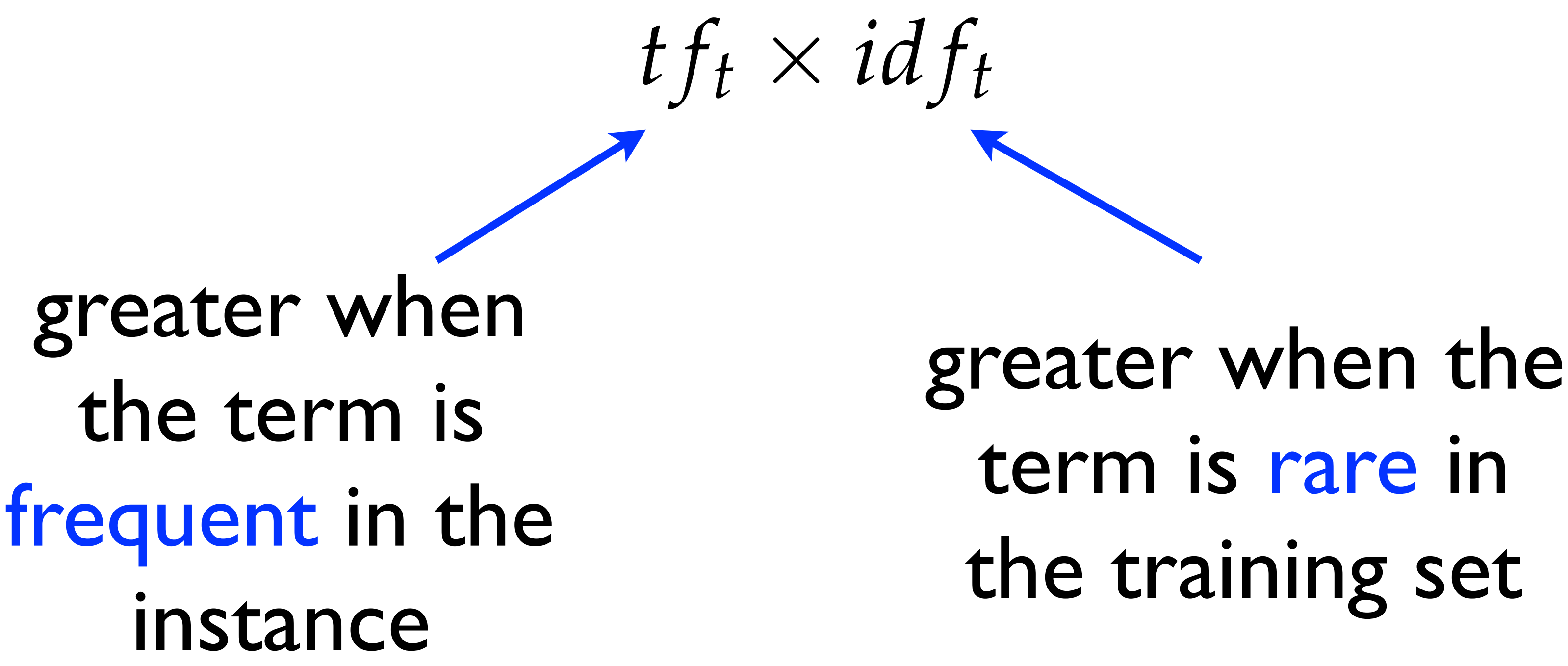
# Inverse Document Frequency (IDF)

## how important is a term?

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

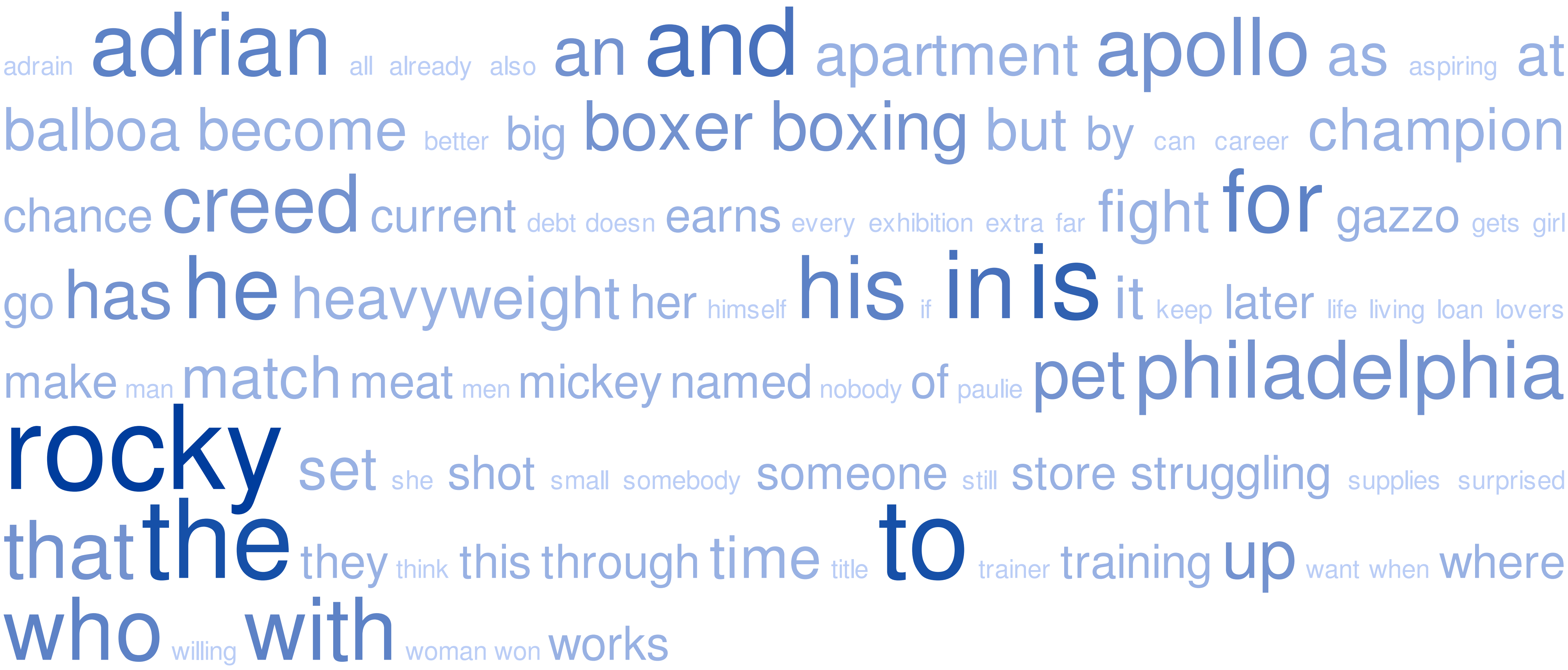
## how important is a term?

| 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 |

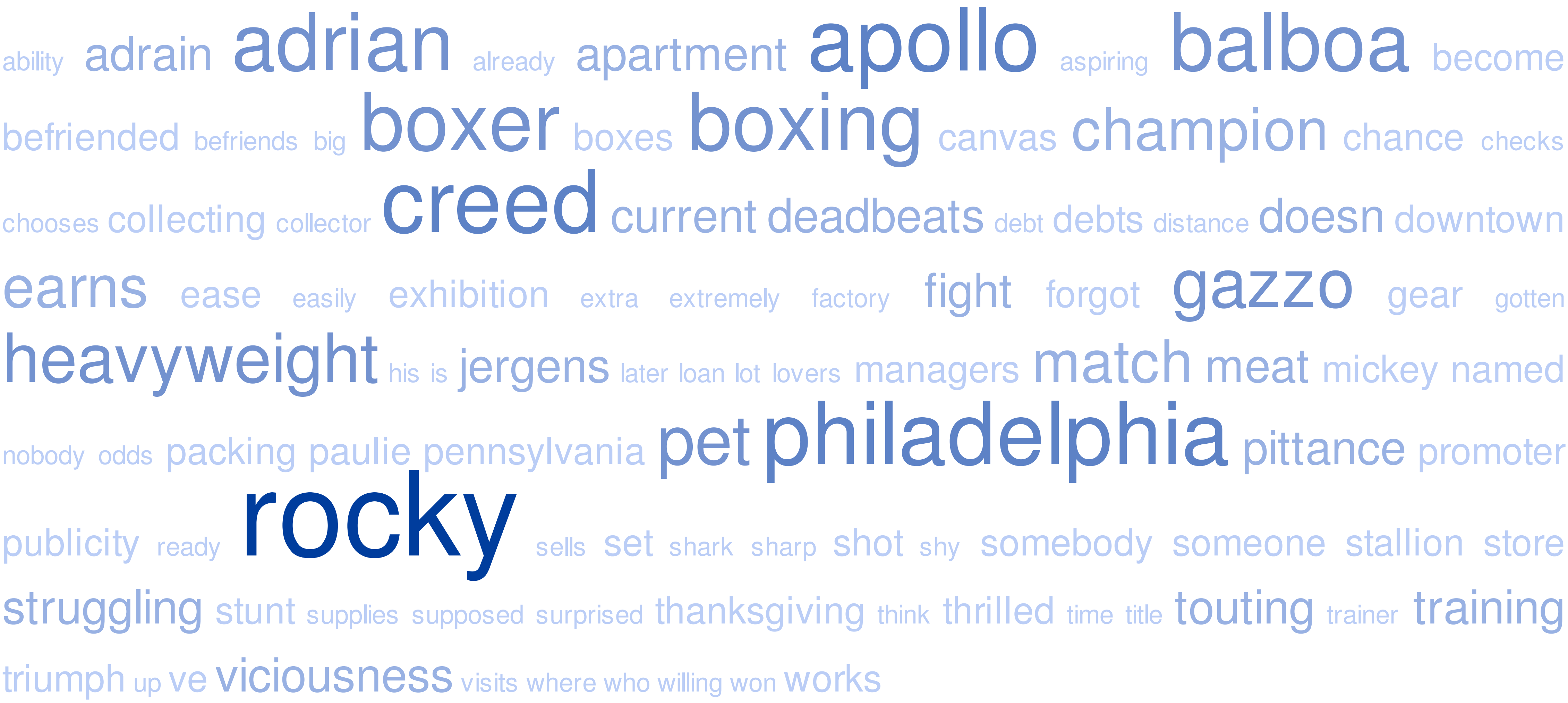
# TF.IDF/Caricature Analogy

- TF.IDF: accentuates terms that are frequent in the instance, but not frequent in general
- Caricature: exaggerates traits that are characteristic of the person compared to the average

# TF, IDF, or TF.IDF?



# TF, IDF, or TF.IDF?



# TF, IDF, or TF.IDF?

ability adrain adrian already apollo aspiring balboa
beat befriended befriends better boxer boxes boxing
canvas cash champion checks chooses collecting
collector creed current deadbeats debt debts
distance doesn downtown earns ease easily
exhibition explains extra extremely factory far forgot
gazzo gear giving gotten heavyweight idea interested
italian jergens keep living loan lot lovers managers match meat
mickey nobody odds packing paulie pennsylvania pet
philadelphia pittance promoter prove publicity
ready rocky sells shark sharp shop shy skills somebody spends
stallion struggling stunt supplies supposed surprised
thanksgiving think thrilled title touting trainer training
triumph unknown ve viciousness visits want willing win
won

# 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 |

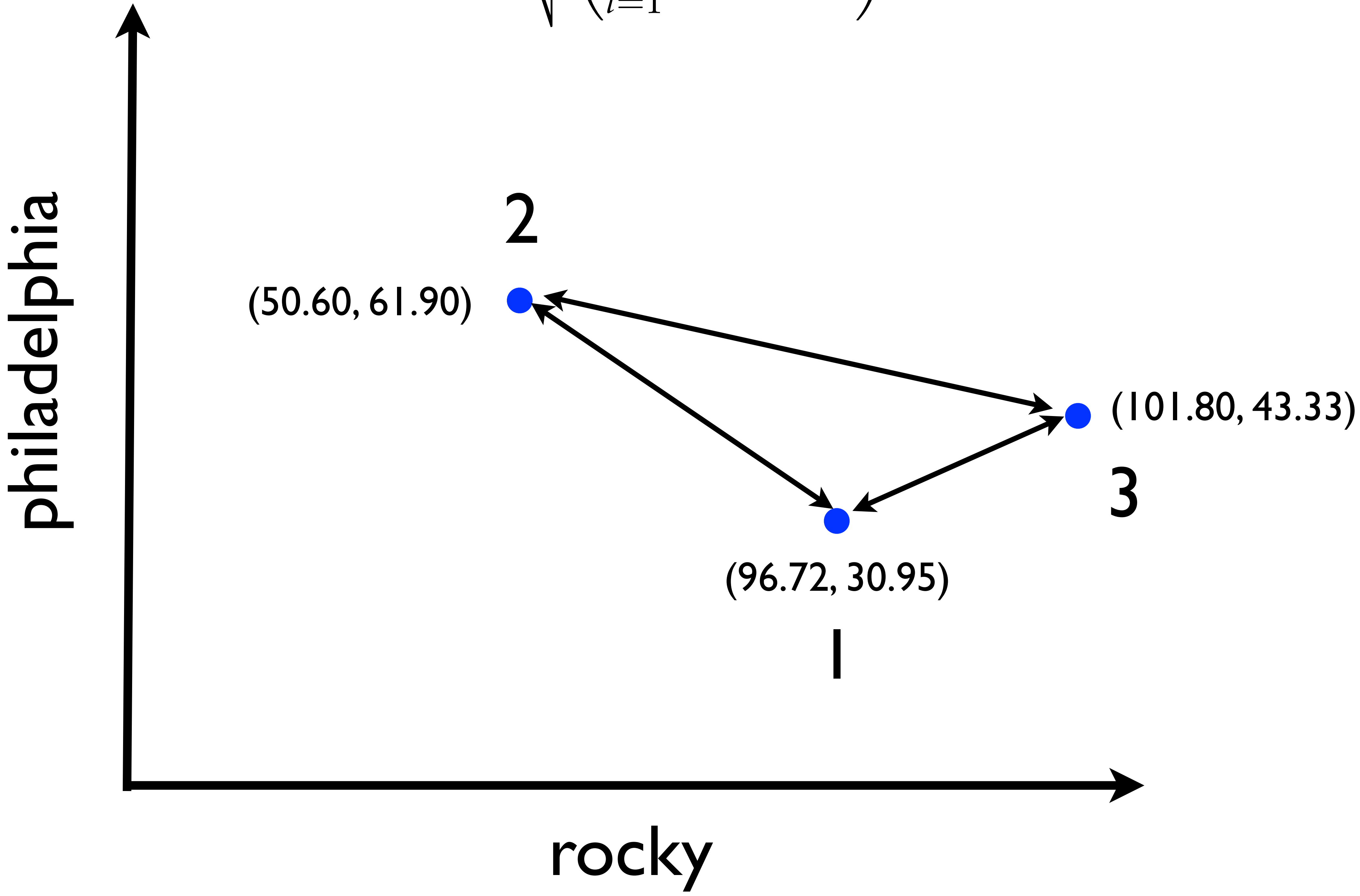
# Putting Everything Together

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

| | term | tf | N | df | idf | tf.idf |
|---|---|---|---|---|---|---|
| 1 | rocky | 19 | 230721 | 1420 | 5.09 | 96.72 |
| | philadelphia | 5 | 230721 | 473 | 6.19 | 30.95 |
| 2 | rocky | 10 | 230721 | 1420 | 5.09 | 50.60 |
| | philadelphia | 10 | 230721 | 473 | 6.19 | 61.90 |
| 3 | rocky | 20 | 230721 | 1420 | 5.09 | 101.80 |
| | philadelphia | 7 | 230721 | 473 | 6.19 | 43.33 |

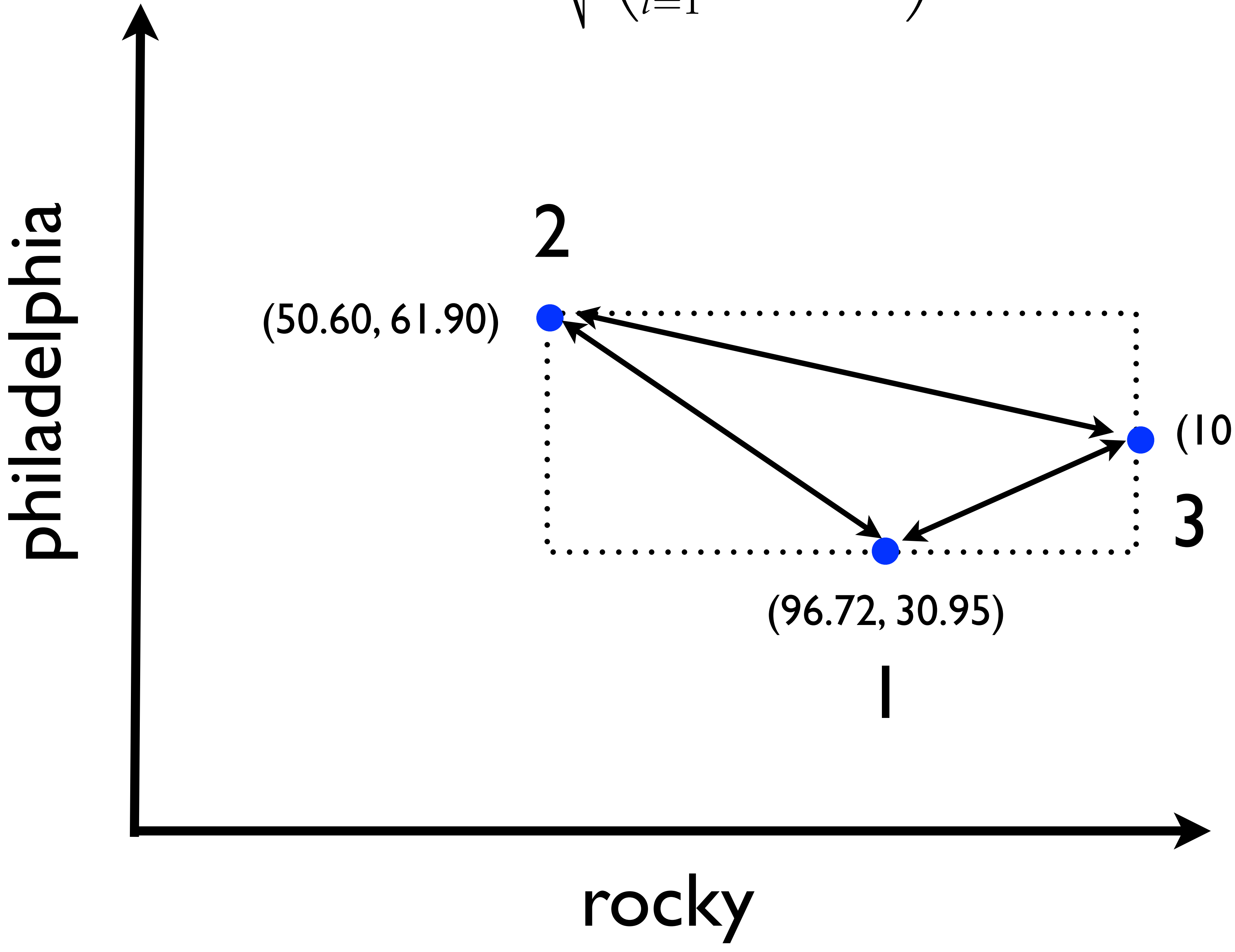
# Putting Everything Together

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

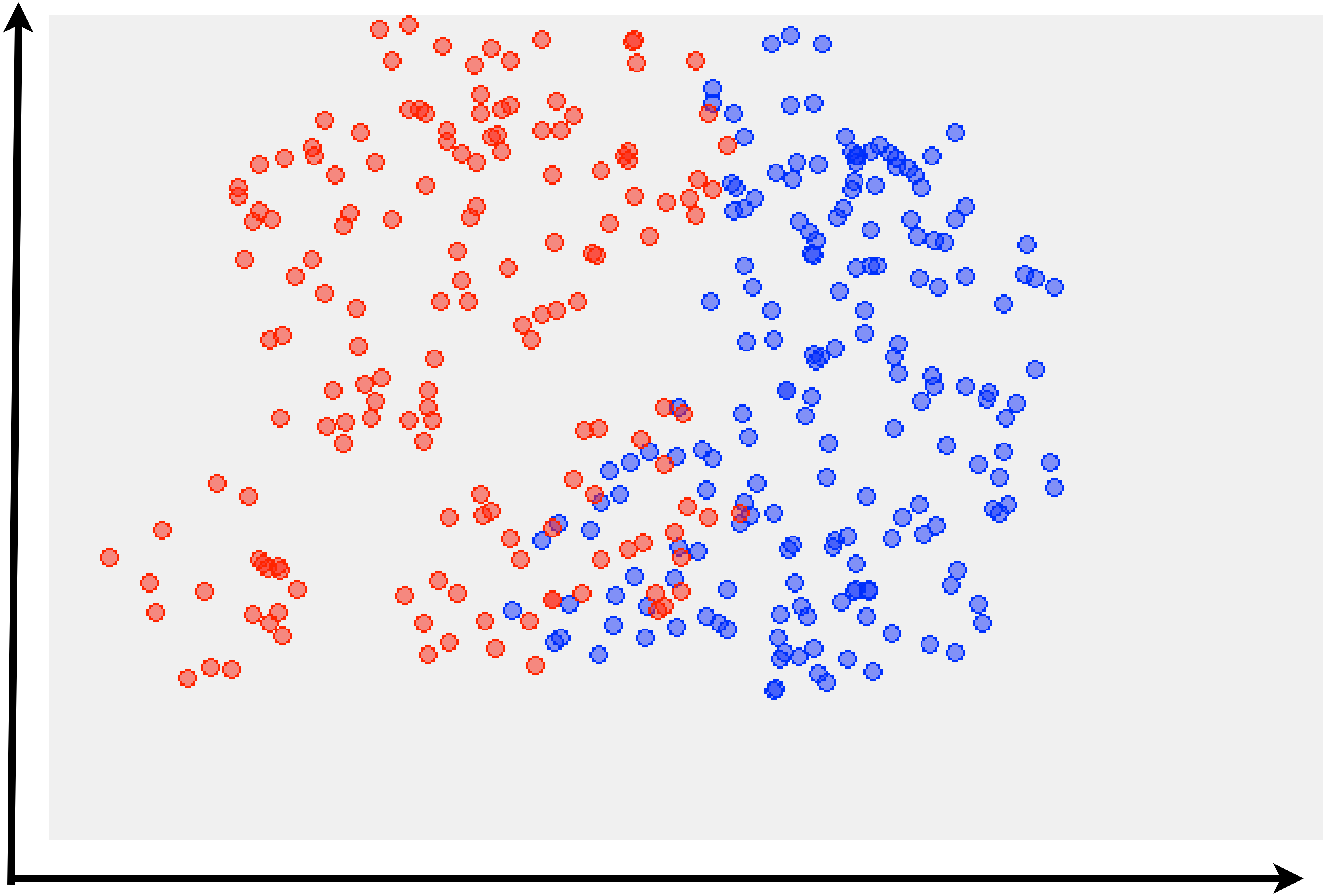


# Putting Everything Together

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

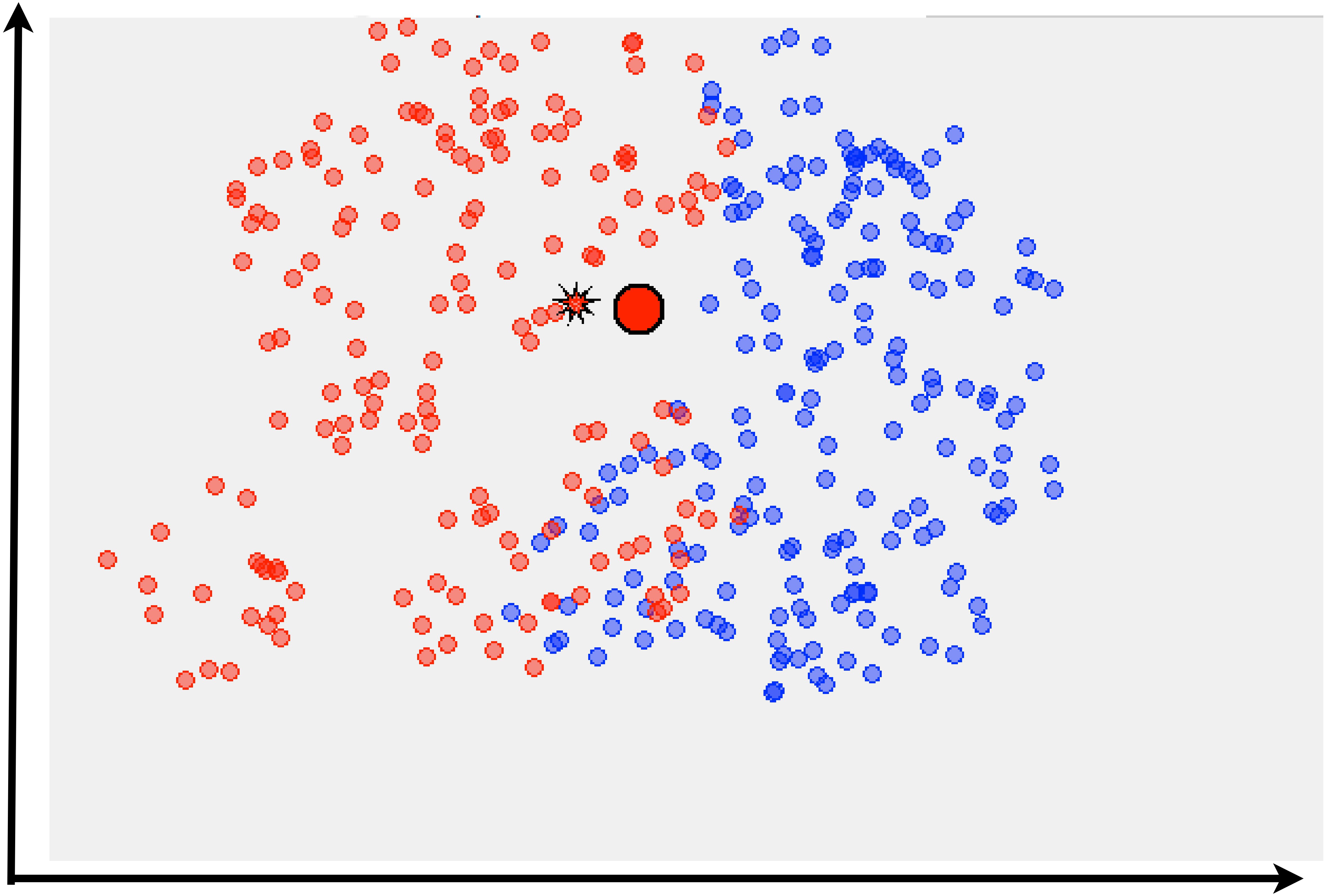


# Nearest-Neighbor Classification

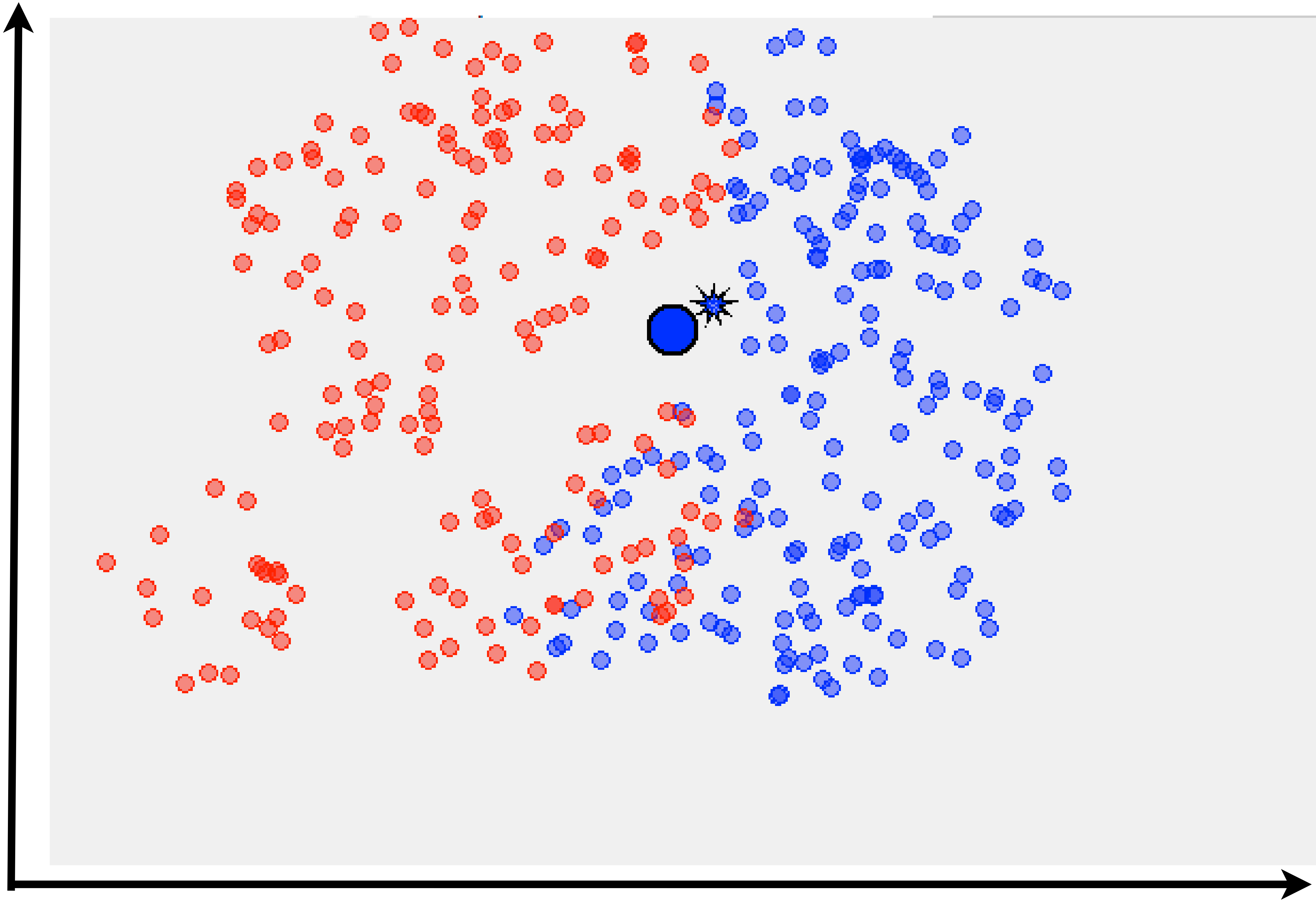


source: http://www.math.le.ac.uk/people/ag153/homepage/KNN/

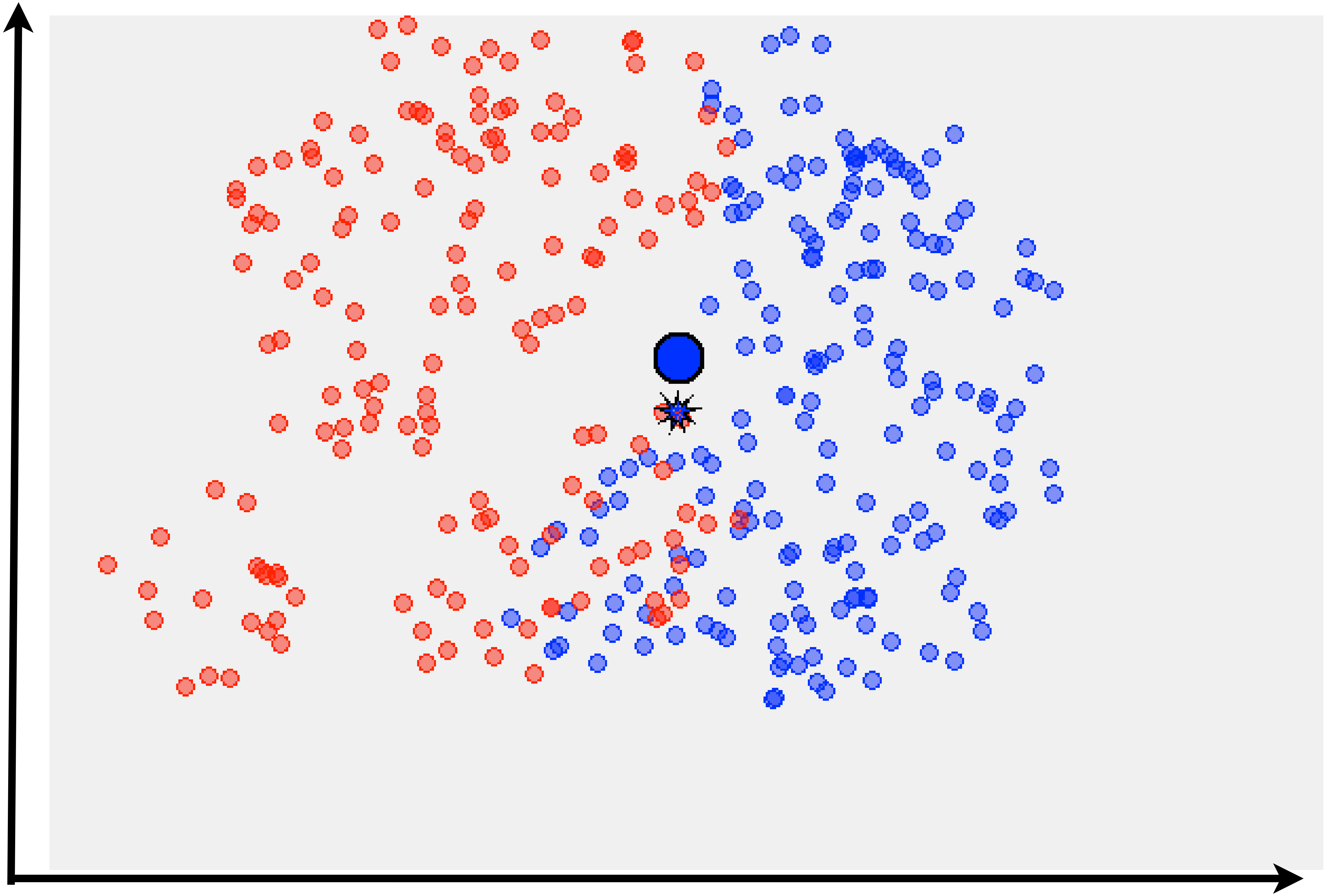
# Nearest-Neighbor Classification



# Nearest-Neighbor Classification



# Nearest-Neighbor Classification



# Nearest-Neighbor Classification

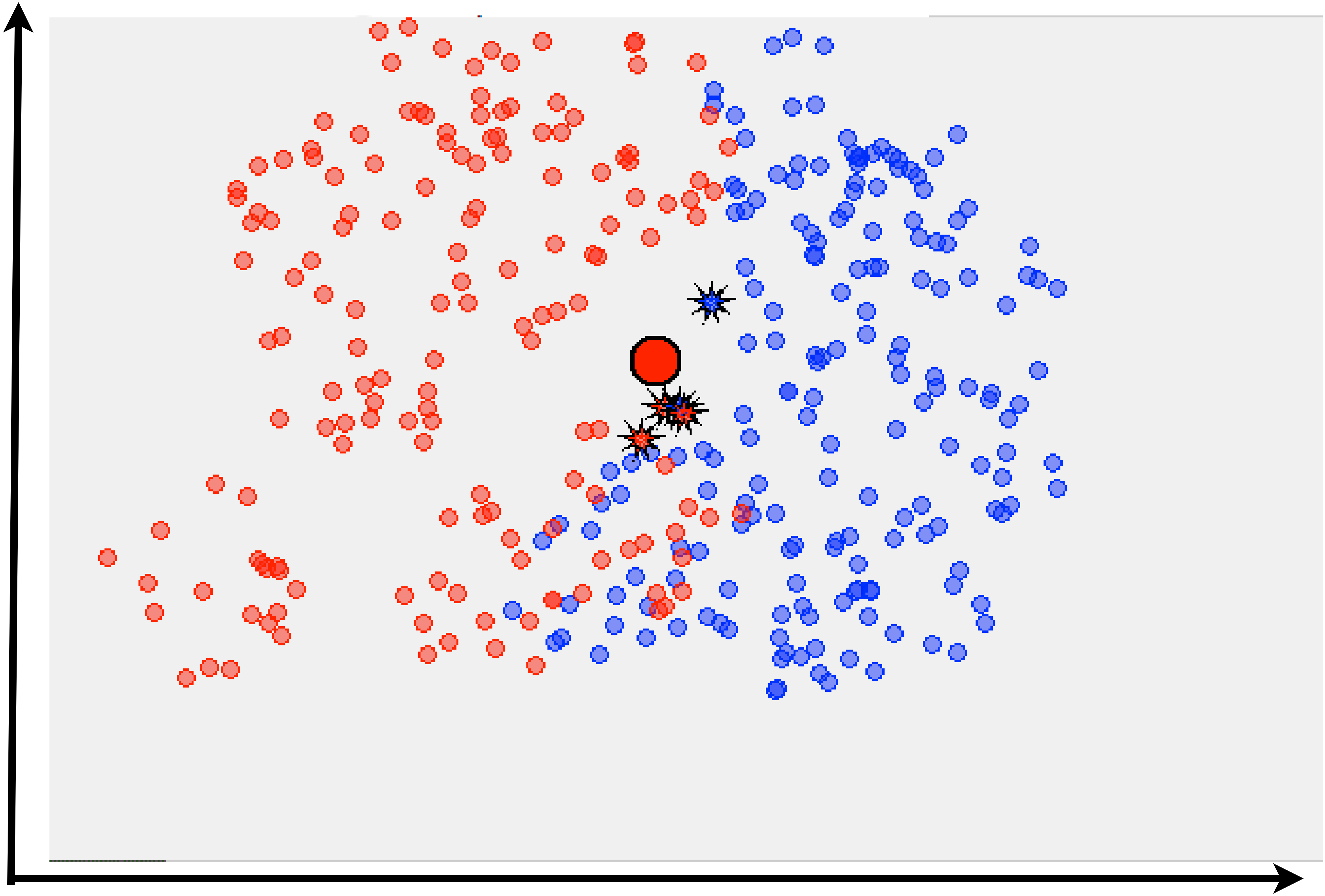
- Given a test instance, assign the label associated with the nearest training set instance
- What is a potential limitation of this approach?

# Nearest-Neighbor Classification

- Given a test instance, assign the label associated with the nearest training set instance
- What is 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

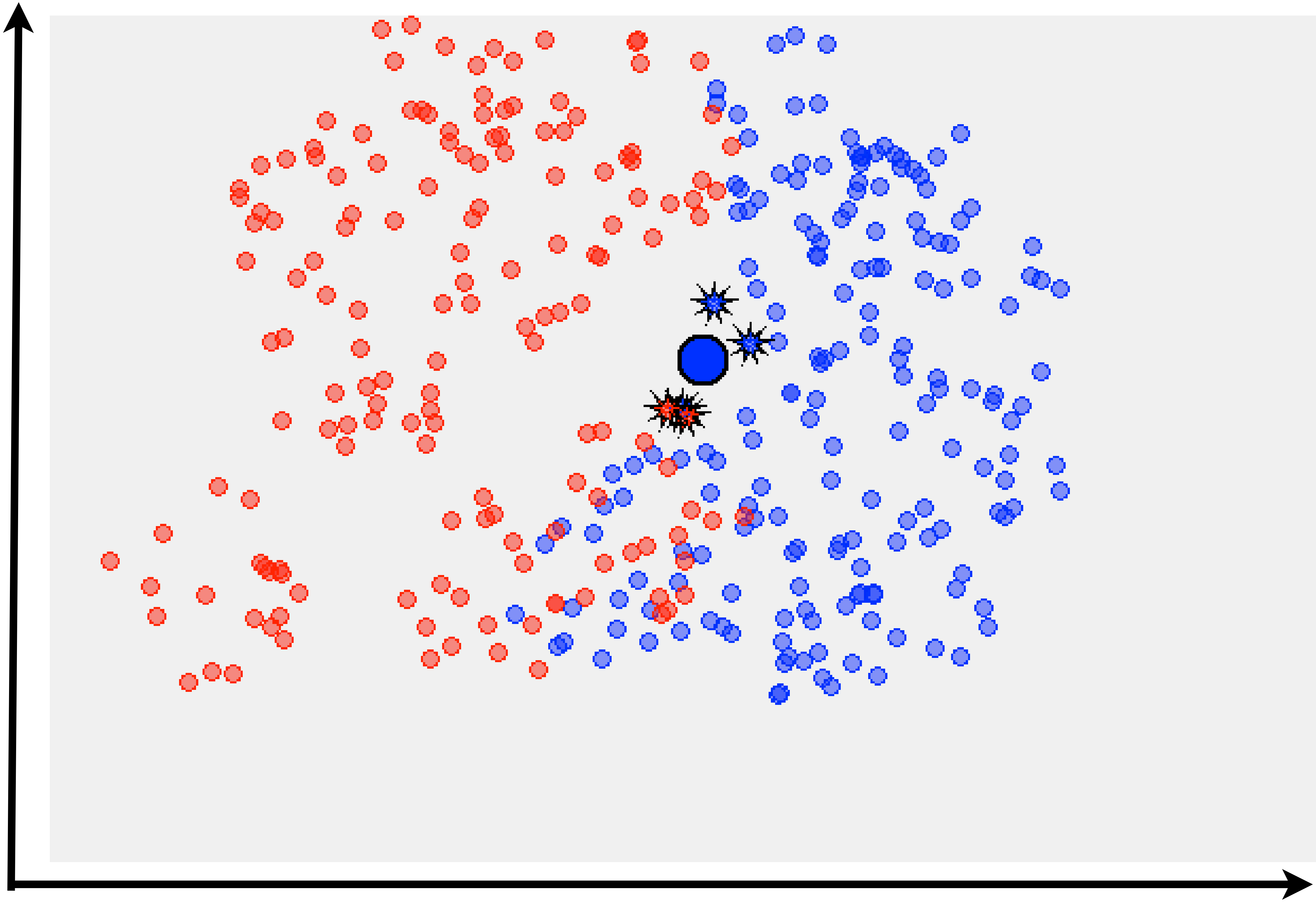
# K Nearest-Neighbor (KNN)

(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 is 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 is 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

# K Nearest-Neighbor (KNN)

## practical matters

- Feature normalization
- Feature weighting
- Computational complexity

# K Nearest-Neighbor (KNN)

## practical matters: feature normalization

- KNN assumes that feature values (and differences in feature value) are comparable between features
- For example, TF.IDF term-weighting places more emphasis on rare terms
- In some cases, we want the opposite
- That is, we want features to be treated equally
- This can be tricky if feature values are not comparable

# K Nearest-Neighbor (KNN)

## practical matters: feature normalization

| w_1 | w_2 | w_3 | w_4 | w_5 | w_6 | w_7 | w_8 | w_9 | w_10 | sentiment |
|---|---|---|---|---|---|---|---|---|---|---|
| 10.5 | 1.2 | 100.4 | 4.54 | 33.4 | 503.4 | 76.8 | 0.54 | 2.31 | 145.6 | positive |
| 13.5 | 1.5 | 101.4 | 5.65 | 34.5 | 400.3 | 79.7 | 0.36 | 5.35 | 353.3 | negative |
| 20.4 | 1.6 | 143.5 | 7.47 | 24.5 | 323.2 | 74.3 | 0.75 | 10.54 | 550.5 | negative |
| 12.4 | 1.4 | 164.2 | 5.76 | 65.6 | 543.2 | 43.4 | 0.23 | 1.65 | 365.2 | positive |
| 12.5 | 3.2 | 156.4 | 4.54 | 67.5 | 234.5 | 45.3 | 0.54 | 1.67 | 543.2 | negative |
| 15.7 | 1.8 | 154.6 | 8.67 | 65.7 | 156.5 | 55.5 | 0.45 | 5.64 | 300.4 | positive |

- Features that capture different types of evidence may have very different ranges

- What can we do so that they have roughly equal contribution?

# K Nearest-Neighbor (KNN)

## min/max normalization

| w_1 | w_2 | w_3 | w_4 | w_5 | w_6 | w_7 | w_8 | w_9 | w_10 | sentiment |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0.21 | 0.9 | 0.92 | 0.6 | 0.07 | 0 | positive |
| 0.3 | 0.15 | 0.02 | 0.27 | 0.23 | 0.63 | 1 | 0.25 | 0.42 | 0.51 | negative |
| 1 | 0.2 | 0.68 | 0.71 | 0 | 0.43 | 0.85 | 1 | 1 | 1 | negative |
| 0.19 | 0.1 | 1 | 0.3 | 0.96 | 1 | 0 | 0 | 0 | 0.54 | positive |
| 0.2 | 1 | 0.88 | 0 | 1 | 0.2 | 0.05 | 0.6 | 0 | 0.98 | negative |
| 0.53 | 0.3 | 0.85 | 1 | 0.96 | 0 | 0.33 | 0.42 | 0.45 | 0.38 | positive |

$$w_{i,j}^{\text{norm}} = \frac{w_{i,j} - \min(w_{i,*})}{\max(w_{i,*}) - \min(w_{i,*})}$$

# K Nearest-Neighbor (KNN)

## practical matters: feature weighting

- In some cases, some features are more important than others
- TF.IDF assumption: the most important features are the rare ones
    - A feature that distinguishes between instances will also distinguish between the target class values
- Alternative: learn feature weights from the training data

# K Nearest-Neighbor (KNN)

## practical matters: feature weighting

- Weighted Euclidean Distance:

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

# K Nearest-Neighbor (KNN)

## practical matters: feature weighting

- Split the training set into two sets
- Make predictions on the second set using the first set
- For each second-set instance that is misclassified based on its first-set nearest neighbor:
    - Find the features where the instances are the most similar
    - Increase their weights (i.e. accentuate their differences)

# K Nearest-Neighbor (KNN)

## practical matters: making predictions

- How fast/slow is KNN is making predictions?

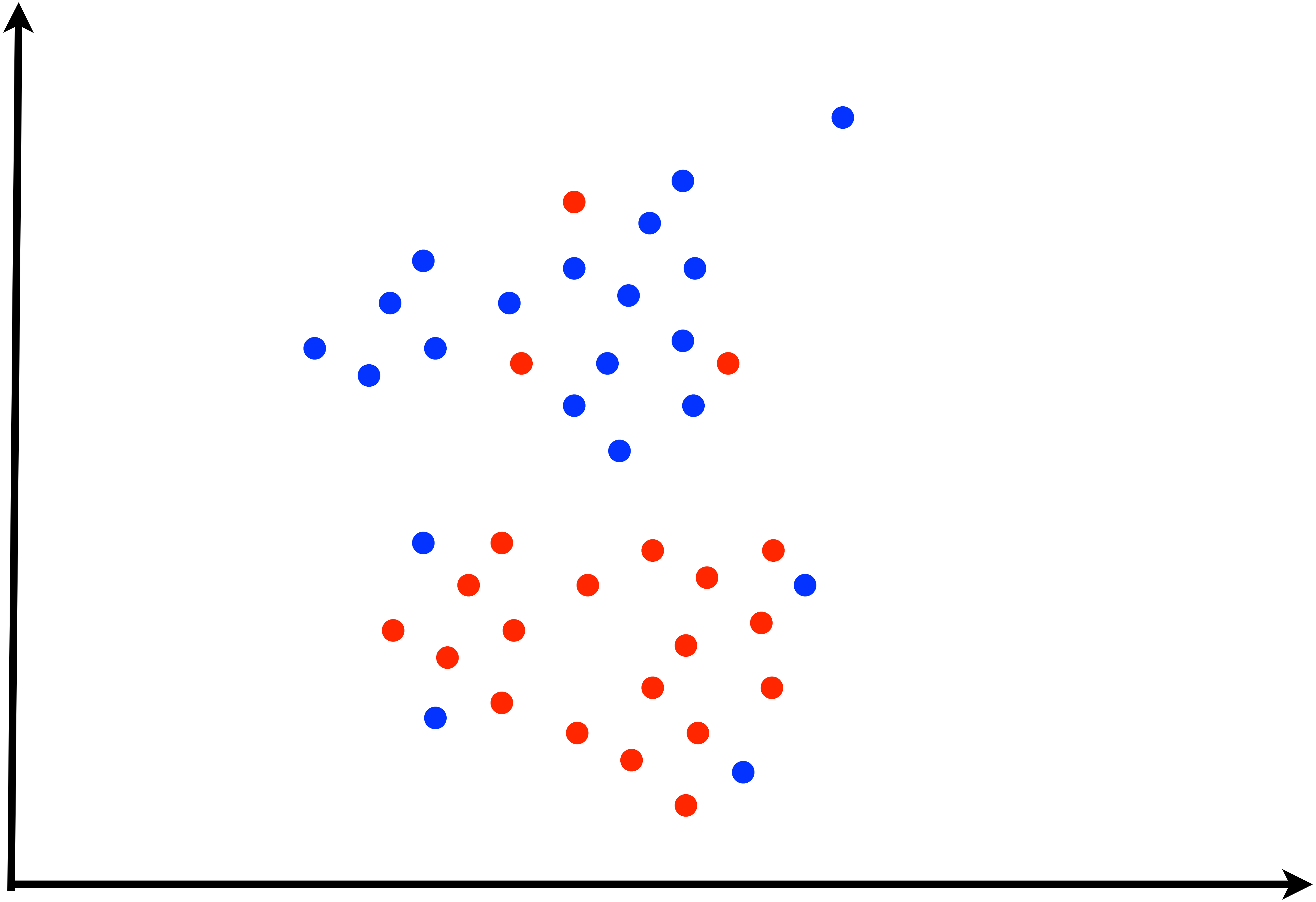
# K Nearest-Neighbor (KNN)

## practical matters: making predictions

- How fast/slow is KNN is making predictions?
- KNN can be very slow
- It needs to compute the similarity/distance between the test instance and <u>every</u> training instance
- Is there anything we can do to speed the process?

# K Nearest-Neighbor (KNN)

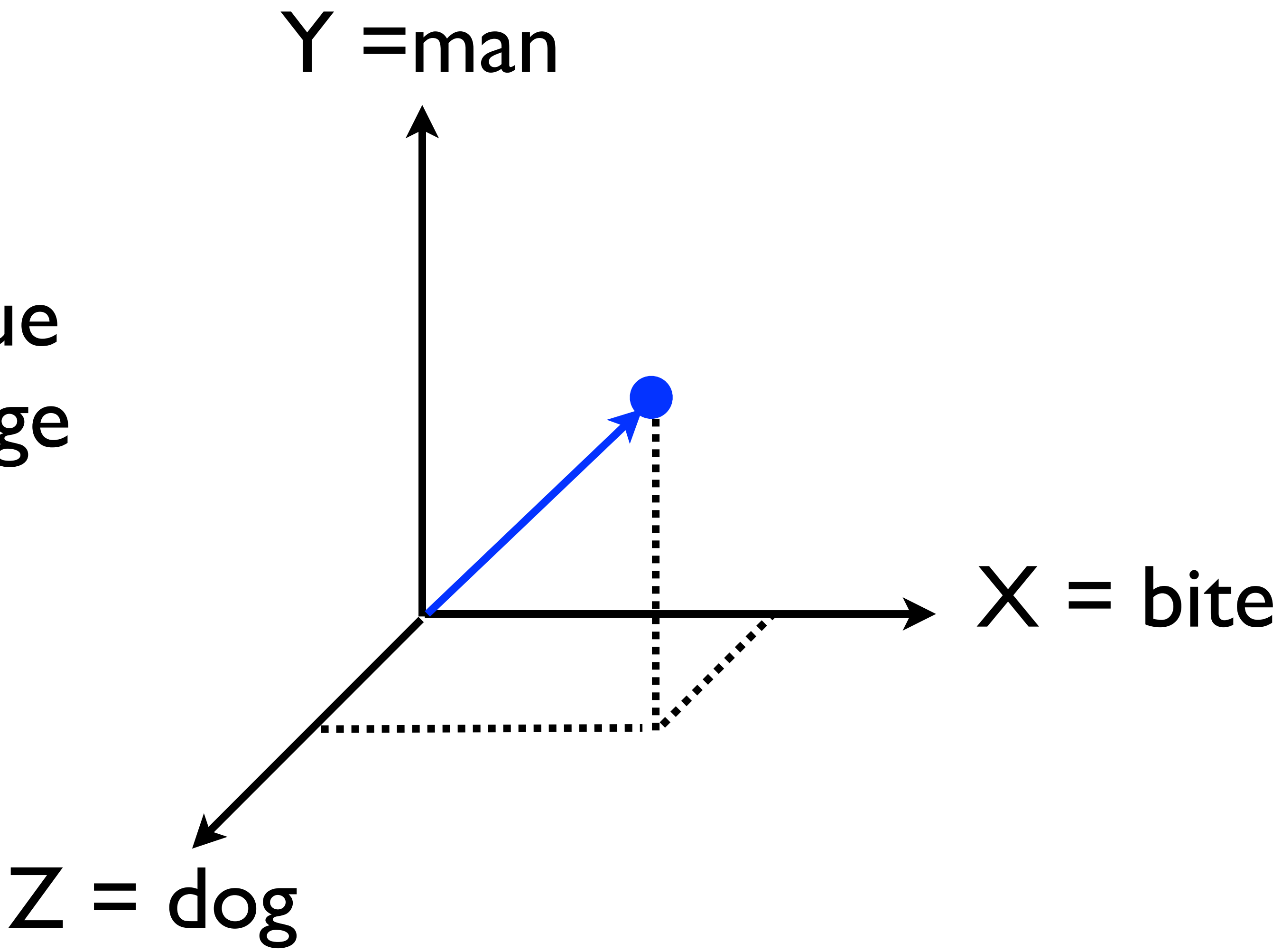
## practical matters: making predictions



# Independence Assumption

- The basis vectors (X, Y, Z) are linearly independent because knowing a vector's value on one dimension doesn't say anything about its value along another dimension

does this hold true for natural language text?



basis vectors for 3-dimensional space

# Mutual Information

## IMDB Corpus

- If this were true, what would these mutual information values be?

| w1 | w2 | MI | w1 | w2 | MI |
|---|---|---|---|---|---|
| francisco | san | ? | dollars | million | ? |
| angeles | los | ? | brooke | rick | ? |
| prime | minister | ? | teach | lesson | ? |
| united | states | ? | canada | canadian | ? |
| 9 | 11 | ? | un | ma | ? |
| winning | award | ? | nicole | roman | ? |
| brooke | taylor | ? | china | chinese | ? |
| con | un | ? | japan | japanese | ? |
| un | la | ? | belle | roman | ? |
| belle | nicole | ? | border | mexican | ? |

# Mutual Information

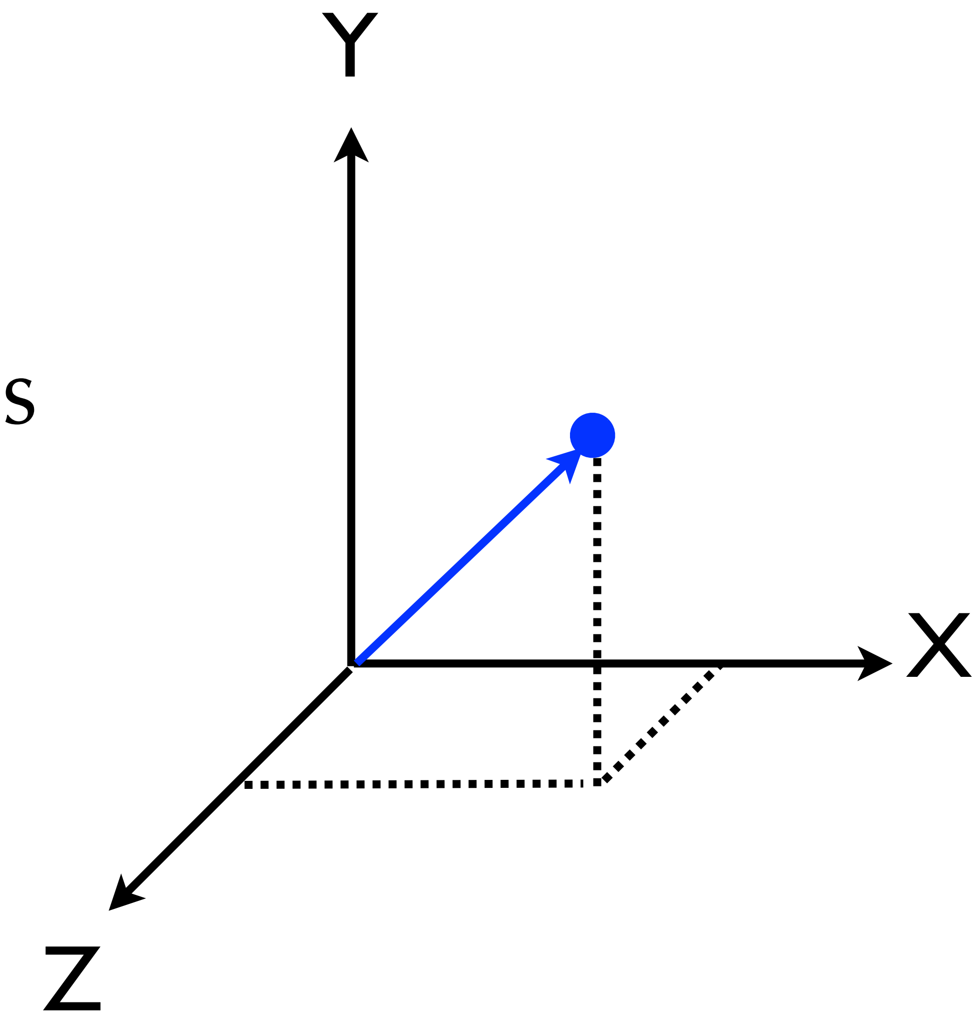
## IMDB Corpus

- These mutual information values should be zero!

| w1 | w2 | MI | w1 | w2 | MI |
|---|---|---|---|---|---|
| francisco | san | 6.619 | dollars | million | 5.437 |
| angeles | los | 6.282 | brooke | rick | 5.405 |
| prime | minister | 5.976 | teach | lesson | 5.370 |
| united | states | 5.765 | canada | canadian | 5.338 |
| 9 | 11 | 5.639 | un | ma | 5.334 |
| winning | award | 5.597 | nicole | roman | 5.255 |
| brooke | taylor | 5.518 | china | chinese | 5.231 |
| con | un | 5.514 | japan | japanese | 5.204 |
| un | la | 5.512 | belle | roman | 5.202 |
| belle | nicole | 5.508 | border | mexican | 5.186 |

# Independence Assumption

- Representing texts as vectors assumes that terms are independent
- The fact that one occurs says nothing about another one occurring
- This is viewed as a limitation
- However, the implications of this limitation are still debated
- A very popular solution



# Summary

- Instance-based classification relies on one assumption:
  - similar instances should have the same label
- Ingredients:
  - similarity metric: to find the nearest neighbors
  - averaging technique: to combine their true labels into a final prediction
- K-NN: use the geometric distance to find the K nearest neighbors and take the majority label