Multimedia Information Retrieval and Technology

Lecture 13 Vector space classification

By : Fangyu Wu

Room: SD555

Xi’an Jiaotong-Liverpool University

西交利物浦大学

I. Rocchio classification

II. *k* nearest neighbor

III. Linear classifier:

Recall_ Naïve bayes

- The Naive Bayes classifier is a probabilistic classifier.
- We compute the probability of a document d being in a class c as follows:

$$P(c|d) \propto P(c) \prod_{1 \leq k \leq n_d} P(t_k|c)$$

- $P(t_k|c)$ is the conditional probability of term t_k occurring in a document of class c
- $P(t_k|c)$ as a measure of how much evidence t_k contributes that c is the correct class.
- $P(c)$ is the prior probability of c.

Recall_ Naïve bayes

$$\hat{P}_{MLE}(c) = \frac{N_c}{N}$$

$$\hat{P}_{MLE}(t|c) = \frac{T_{ct} + 1}{\sum_{t' \in V} (T_{ct'} + 1)} = \frac{T_{ct} + 1}{\sum_{t' \in V} T_{ct'} + B}$$

Multinomial NB model:

T_{ct} is the term frequency of term *t* in training documents of class *c;*

Recall: The Bernoulli model

The probability of nonoccurrence is factored in when computing $P(c|d)$:

Models absence of terms explicitly.

$$c_{map} = \arg\max \hat{P}(c) \prod_{t_k \in q} \hat{P}(t_k|c)) \prod_{t_k \notin q} [1 - \hat{P}(t_k|c)]$$

Recall: The Bernoulli model

$$\hat{P}_{MLE}(t|c) = \frac{df_{ct} + 1}{N_c + No\ of\ classes}$$

df_{ct} is the document frequency of term t in class c ;
N_c is the number of documents in class c.

Recall: Vector Space Representation

Each document is a vector, one component for each term (= word).

Normally normalize vectors to unit length.

High-dimensional vector space:

- Terms are axes
- 10,000+ dimensions, or even 100,000+
- Docs are vectors in this space

How can we do classification in this space?

Classification Using Vector Spaces

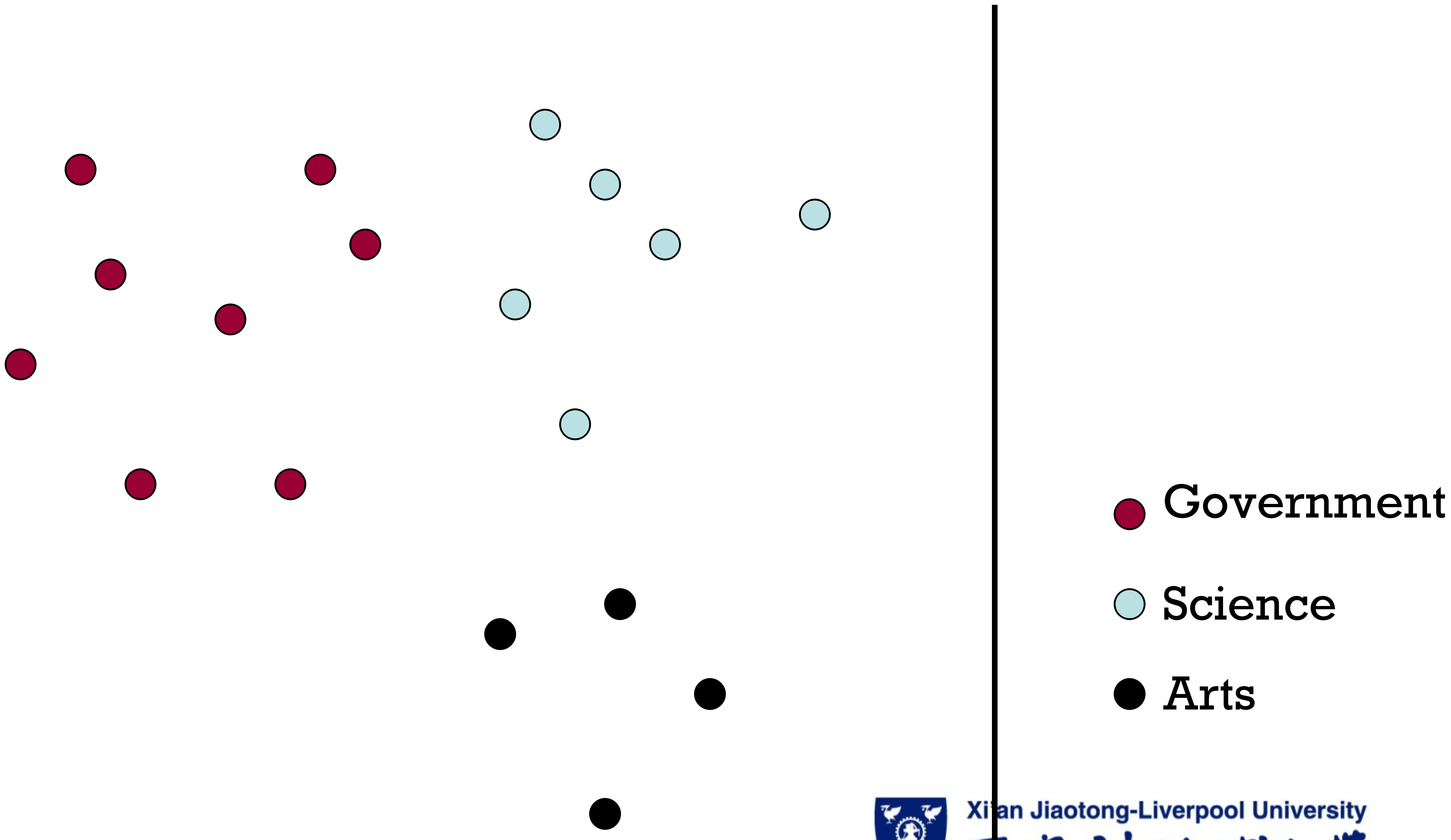
In vector space classification, training set corresponds to a labeled set of points (equivalently, vectors)

Premise 1: Documents in the same class form a contiguous region of space

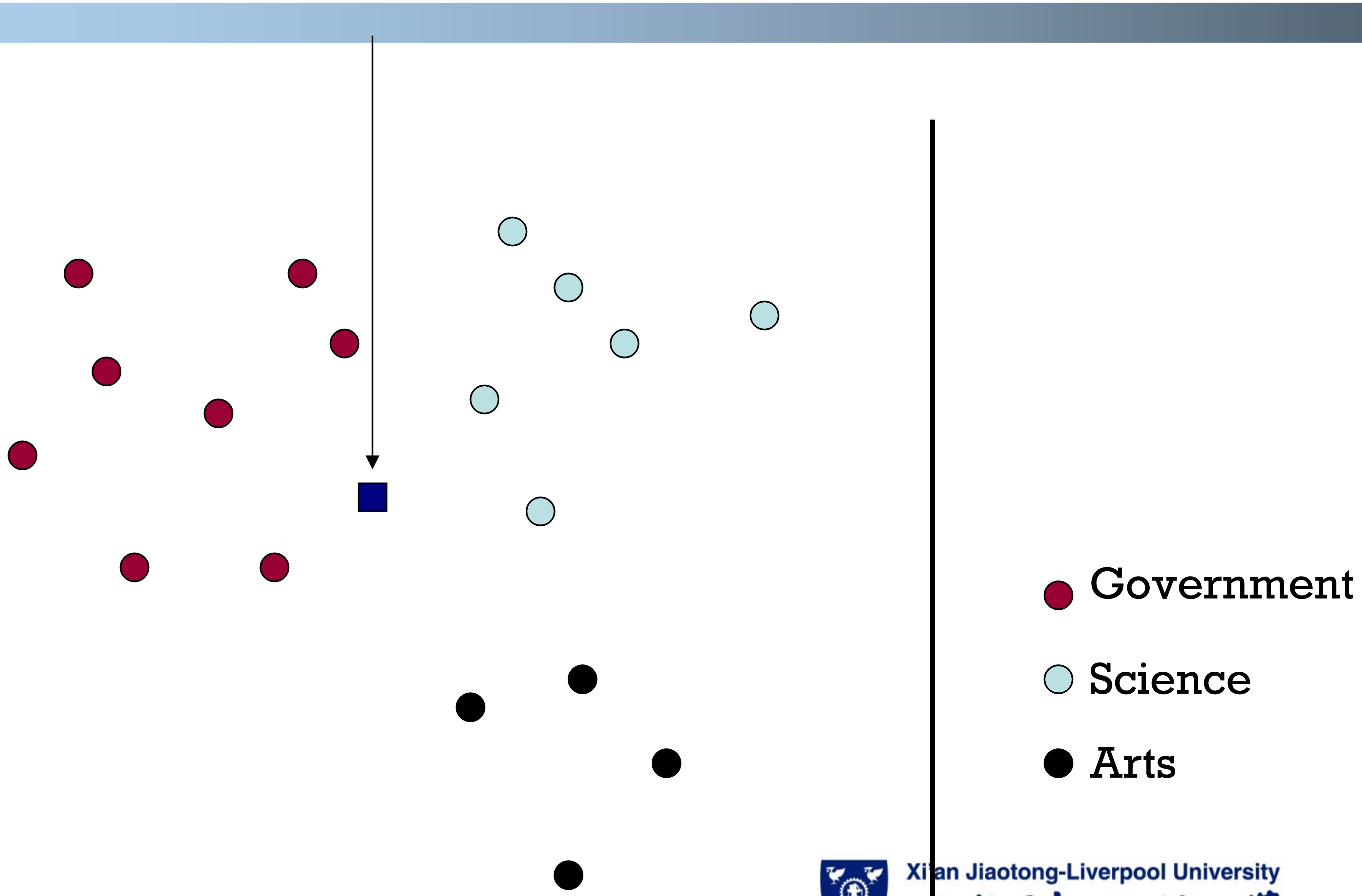
Premise 2: Documents from different classes don’t overlap (much)

Learning a classifier: build surfaces to delineate classes in the space

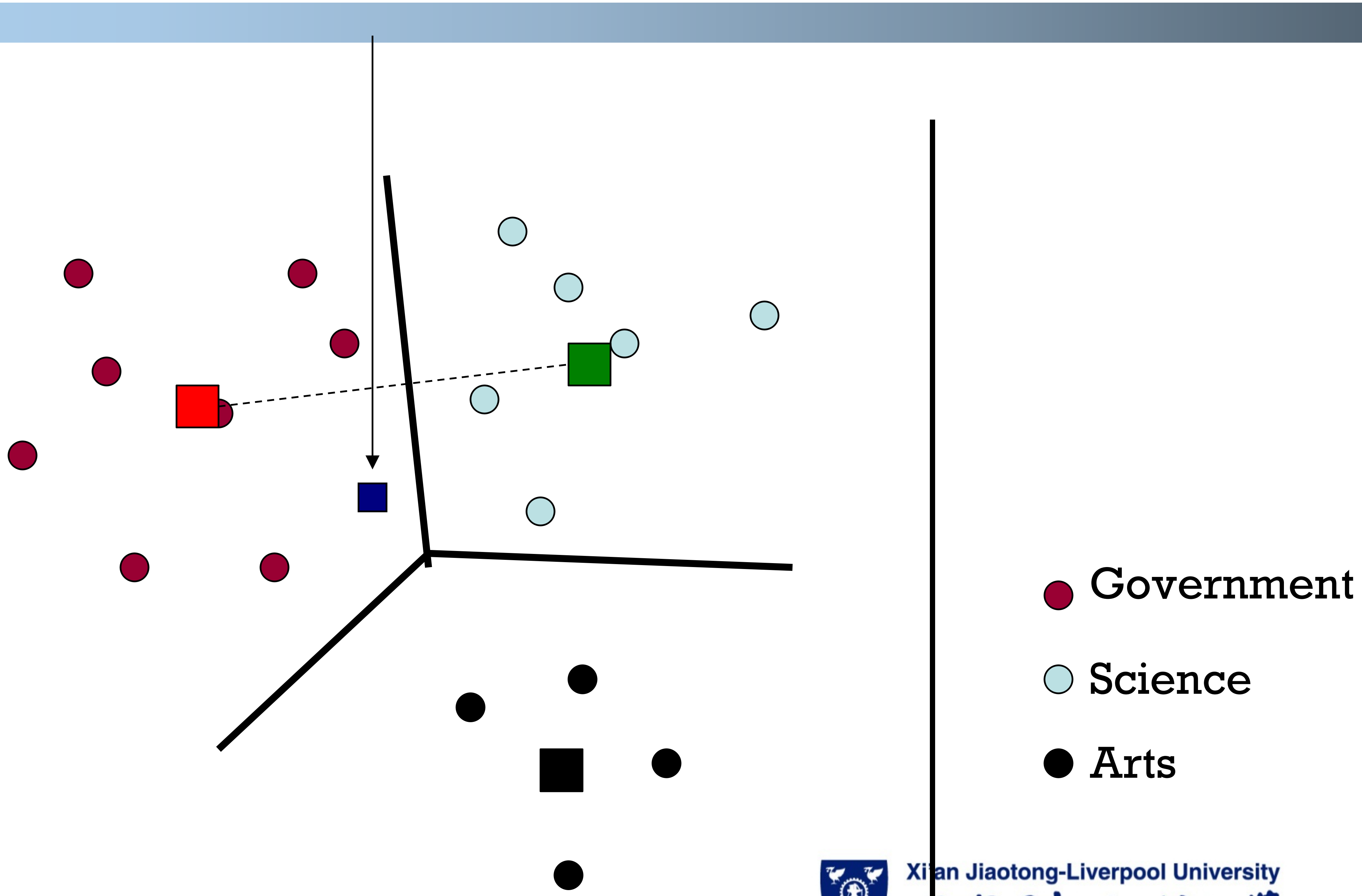
Documents in a Vector Space



Test Document of what class?



Test Document = Government



Our focus: how to find good separators

Definition of centroid

$$\vec{\mu}_c = \frac{1}{|D_c|} \sum_{d \in D_c} \vec{d}$$

Where D_c is the set of all documents that belong to class *c* and $\vec{d}$ is the vector space representation of document *d*.

Rocchio classification

The boundary between two classes in Rocchio classification is the set of points with equal distance from the two centroids.

The generalization of a **decision boundaries** in *M*-dimensional space is a hyperplane, which we define as the set of points $\vec{x}$ that satisfy:

$$\vec{w}^T\vec{x} = b$$

Hyperplanes

This definition of **hyperplanes** includes lines and 2-dimensional planes

- any line in 2D can be defined by $w_1x_1 + w_2x_2 = b$
- any plane in 3D can be defined by $w_1x_1 + w_2x_2 + w_3x_3 = b$

A line divides a plane in two, a plane divides 3-dimensional space in two, and hyperplanes divide higher dimensional spaces in two.

Rocchio classification

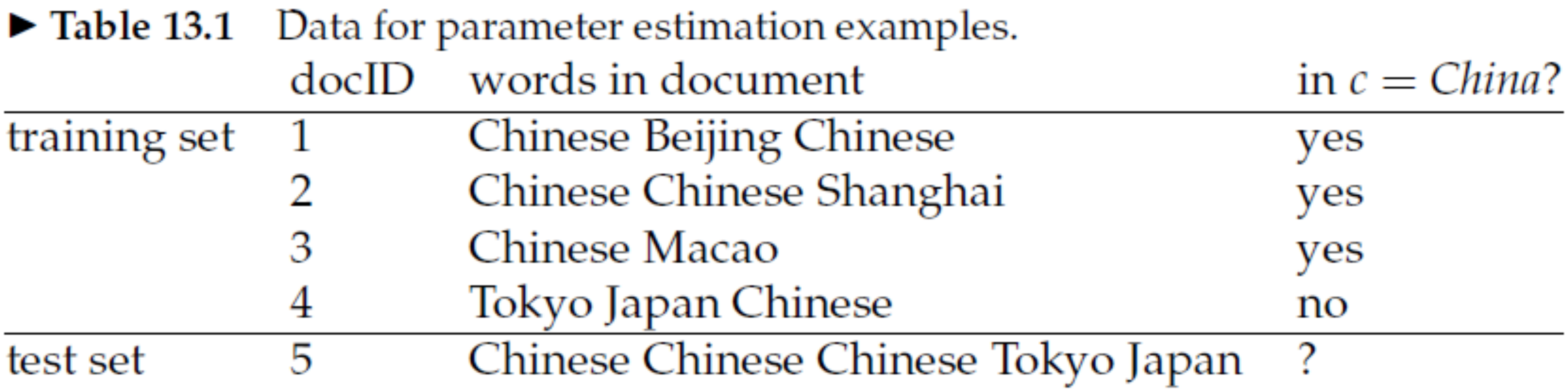
Thus, the boundaries of class regions in Rocchio classification are **hyperplanes**.

The classification rule in Rocchio is to classify a point in accordance with the region it falls into.

Equivalently, we determine the centroid $\vec{u}(c)$ that the point (the test document) is closest to and then assign it to *c*.

Example

1) Compute the tf-idf vector (with normalization) representations of the five documents in the table below.
2) Then calculate the two class centroids u_c and $u_{\bar{c}}$
3) Decide the class of document 5.

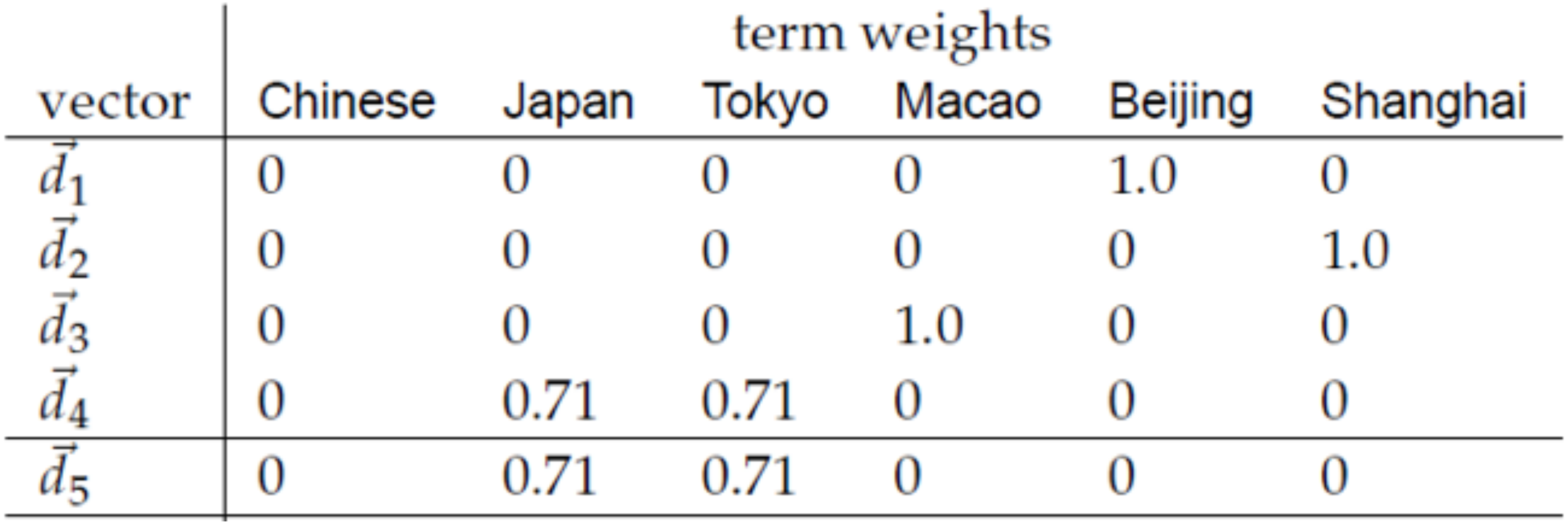


▶ **Table 13.1** Data for parameter estimation examples.

	docID	words in document	in $c = China$?
training set	1	Chinese Beijing Chinese	yes
	2	Chinese Chinese Shanghai	yes
	3	Chinese Macao	yes
	4	Tokyo Japan Chinese	no
test set	5	Chinese Chinese Chinese Tokyo Japan	?

	tf-idf weights vector (with normalization)					
vector	Chinese	Japan	Tokyo	Macao	Beijing	Shanghai
	0	0	0	0	1	0
$\vec{d}_1$	0	0	0	0	0	1
				1		
$\vec{d}_2$	0	0.7				
$\vec{d}$						

Xi'an Jiaotong-Liverpool University
西交利物浦大学



	term weights					
vector	Chinese	Japan	Tokyo	Macao	Beijing	Shanghai
$\vec{d}_1$	0	0	0	0	1.0	0
$\vec{d}_2$	0	0	0	0	0	1.0
$\vec{d}_3$	0	0	0	1.0	0	0
$\vec{d}_4$	0	0.71	0.71	0	0	0
$\vec{d}_5$	0	0.71	0.71	0	0	0

Solution

$$u_c = 1/3(\vec{d}_1 + \vec{d}_2 + \vec{d}_3)$$

And

$$u_{\bar{c}} = 1/1 \cdot (\vec{d}_4)$$

The distances of the test document from the centroids are $|u_c - \vec{d}_5| \approx 1.15$ and $|u_{\bar{c}} - \vec{d}_5| = 0$. Thus, Rocchio assigns $\vec{d}_5$ to $\bar{c}$.

Xi'an Jiaotong-Liverpool University
西交利物浦大學

Evaluating Categorization

Classification accuracy: r/n

where n is the total number of test docs and r is the number of test docs correctly classified.

Exercise

Consider the following supervised corpus of news headlines, where the document class is in bold :

[World News] “Iraq election", “French executive injured", “Teen survives avalanche"

[Business] “Chief executive smiles", “Krispy Kreme executive resigns“

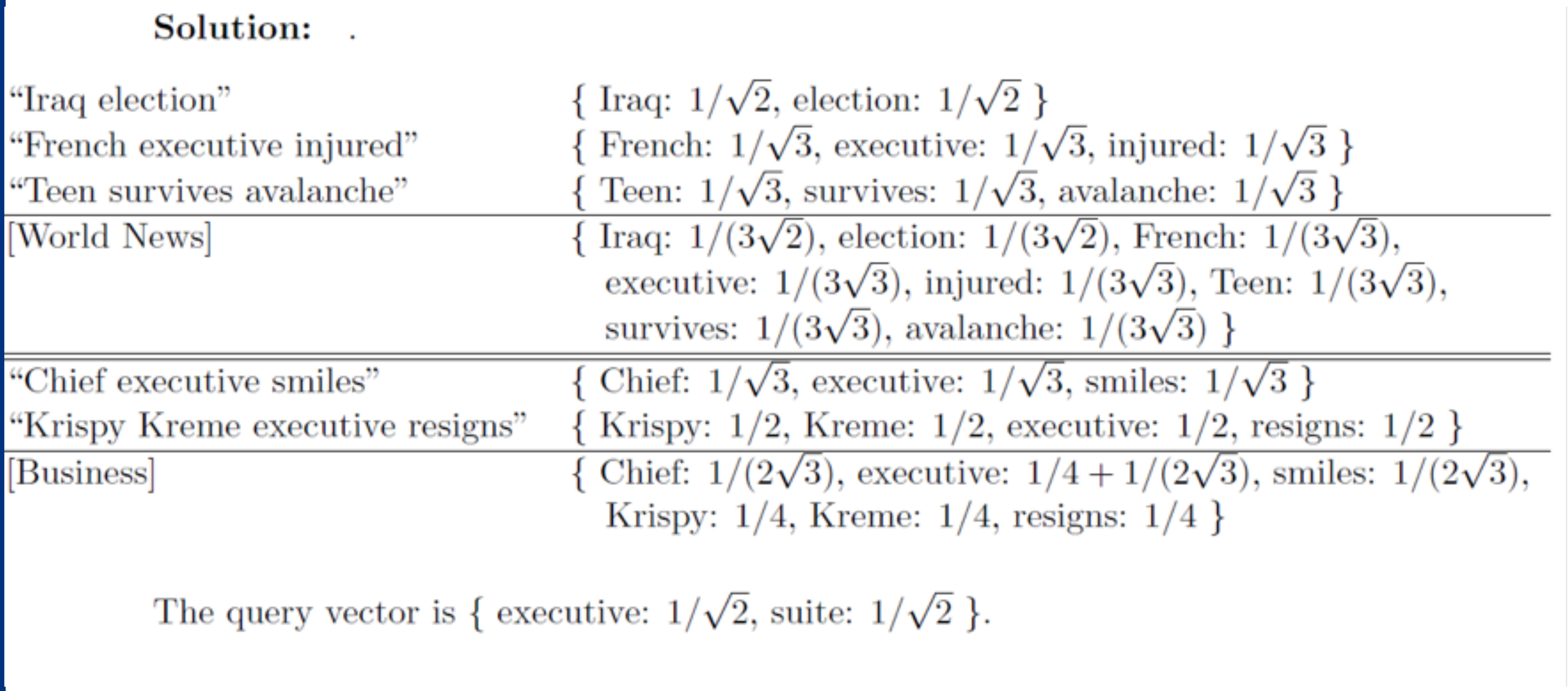
Using this corpus, we will try to predict the class of the document “executive suite".

1. Rocchio Classication Algorithm

(a) Compute the centroid of each class. Express the centroids and the query document as raw term frequency vectors(normalized).

(b) Determine the class of the document using the Rocchio classication algorithm.

Xi’an Jiaotong-Liverpool University
西交利物浦大学



Solution: .

"Iraq election"	{ Iraq: $1/\sqrt{2}$, election: $1/\sqrt{2}$ }
"French executive injured"	{ French: $1/\sqrt{3}$, executive: $1/\sqrt{3}$, injured: $1/\sqrt{3}$ }
"Teen survives avalanche"	{ Teen: $1/\sqrt{3}$, survives: $1/\sqrt{3}$, avalanche: $1/\sqrt{3}$ }
[World News]	{ Iraq: $1/(3\sqrt{2})$, election: $1/(3\sqrt{2})$, French: $1/(3\sqrt{3})$, executive: $1/(3\sqrt{3})$, injured: $1/(3\sqrt{3})$, Teen: $1/(3\sqrt{3})$, survives: $1/(3\sqrt{3})$, avalanche: $1/(3\sqrt{3})$ }
"Chief executive smiles"	{ Chief: $1/\sqrt{3}$, executive: $1/\sqrt{3}$, smiles: $1/\sqrt{3}$ }
"Krispy Kreme executive resigns"	{ Krispy: $1/2$, Kreme: $1/2$, executive: $1/2$, resigns: $1/2$ }
[Business]	{ Chief: $1/(2\sqrt{3})$, executive: $1/4 + 1/(2\sqrt{3})$, smiles: $1/(2\sqrt{3})$, Krispy: $1/4$, Kreme: $1/4$, resigns: $1/4$ }

The query vector is { executive: $1/\sqrt{2}$, suite: $1/\sqrt{2}$ }.

Solution: We measure the distance between the query vector and each centroid, and choose the class with the smallest distance.

- With normalization, the query is classified as [Business].
- Without normalization, the query is labeled [World News].

I. Rocchio classification

II. *k* nearest neighbor

III. Linear classifier:

k nearest neighbor

1 NN is not very robust - one document can be mislabeled or misleading.

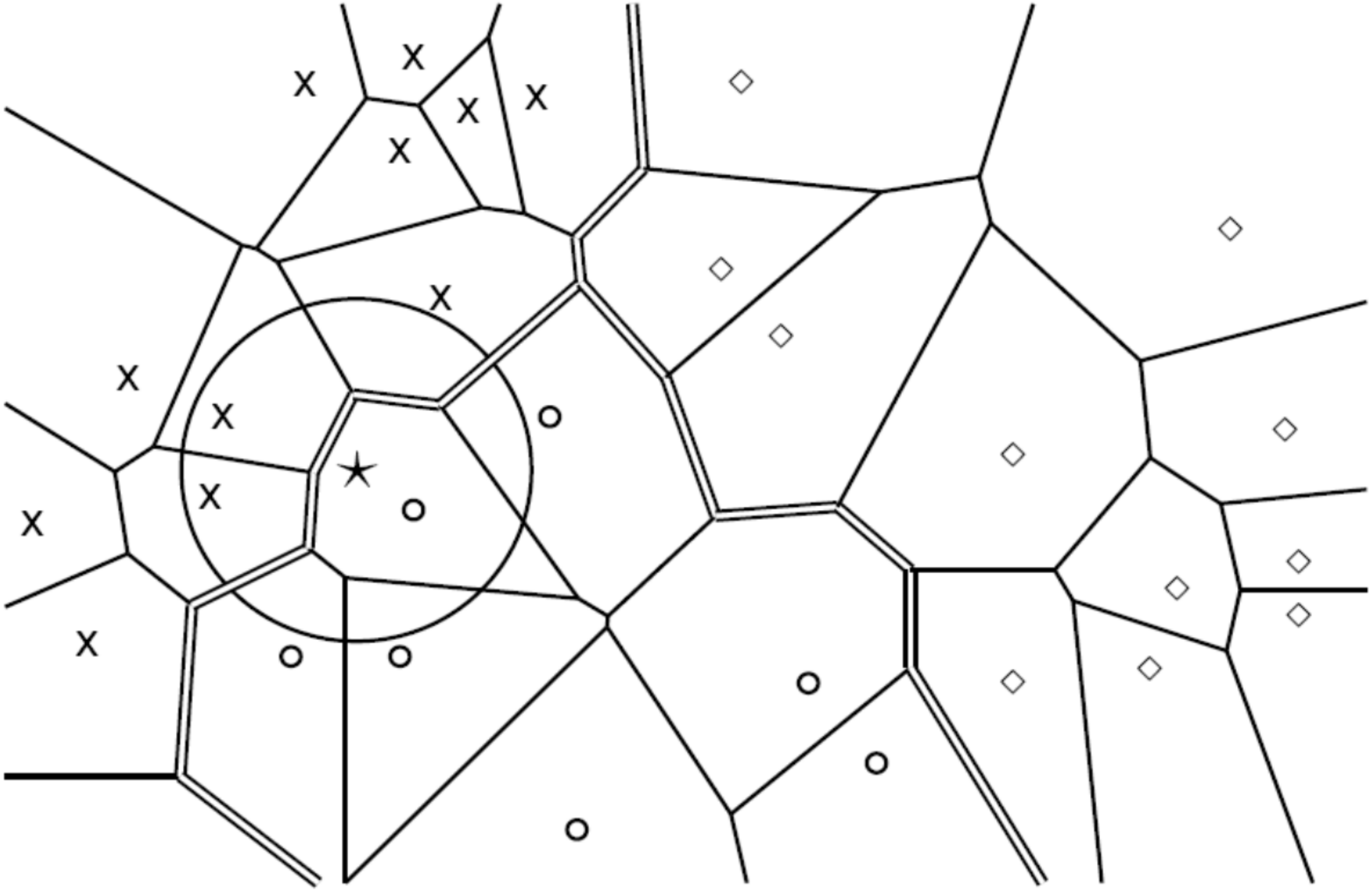
- For k = 1 (1 NN), assign each test document to the class of its nearest neighbor in the training set.

- For k > 1, assign each test document to the majority class of its *k* nearest neighbors in the training set.

- This amounts to locally defined decision boundaries between classes - far away points do not influence the classification decision.

- Probabilistic version of kNN: P(c|d) =fraction of k neighbors of d that are in c.

k nearest neighbor



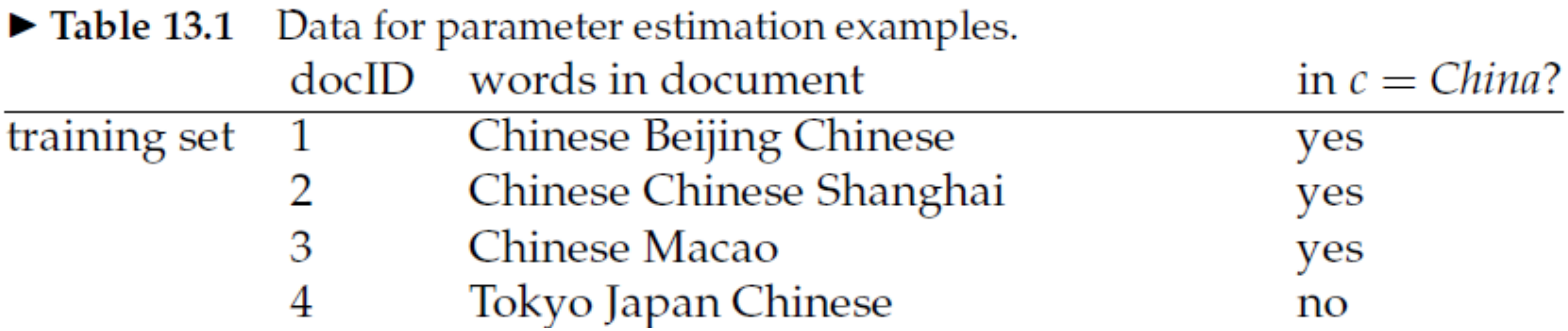
k nearest neighbor

- No feature selection necessary
- No training necessary
- Scales well with large number of classes
  - ⍰Don’t need to train *n* classifiers for *n* classes
- In most cases it’s more accurate than NB or Rocchio

Example

What are the three nearest neighbors to the query? What class does this document belong to if we consider 3NN classification? Assuming raw term frequency, no idf, and cosine similarity for distance calculation.

Consider the query document "Chinese Chinese Japan"



► **Table 13.1** Data for parameter estimation examples.

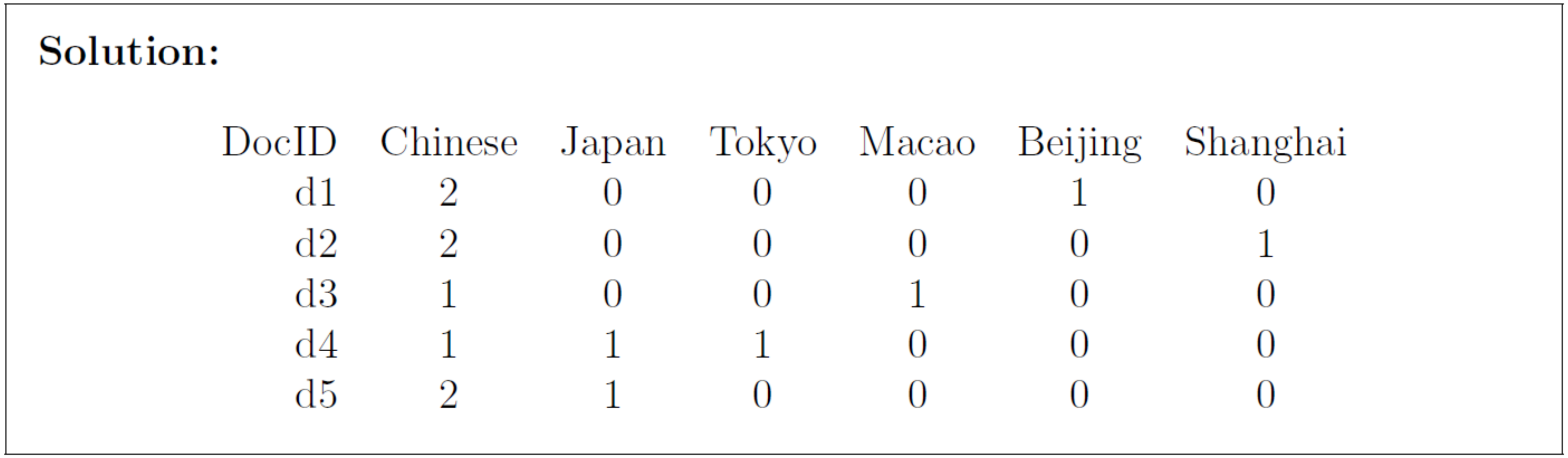
	docID	words in document	in $c = China$?
training set	1	Chinese Beijing Chinese	yes
	2	Chinese Chinese Shanghai	yes
	3	Chinese Macao	yes
	4	Tokyo Japan Chinese	no

Recap

Unit vectors

$$\cos(\vec{q},\vec{d}) = \frac{\vec{q} \bullet \vec{d}}{|\vec{q}||\vec{d}|} = \frac{\vec{q}}{|\vec{q}|} \bullet \frac{\vec{d}}{|\vec{d}|} = \frac{\sum_{i=1}^{|V|} q_i d_i}{\sqrt{\sum_{i=1}^{|V|} q_i^2}\sqrt{\sum_{i=1}^{|V|} d_i^2}}$$

solution



Solution:

DocID	Chinese	Japan	Tokyo	Macao	Beijing	Shanghai
d1	2	0	0	0	1	0
d2	2	0	0	0	0	1
d3	1	0	0	1	0	0
d4	1	1	1	0	0	0
d5	2	1	0	0	0	0

$\sqrt{3}=1.732$ $\sqrt{5}=2.236$

I. Rocchio classification

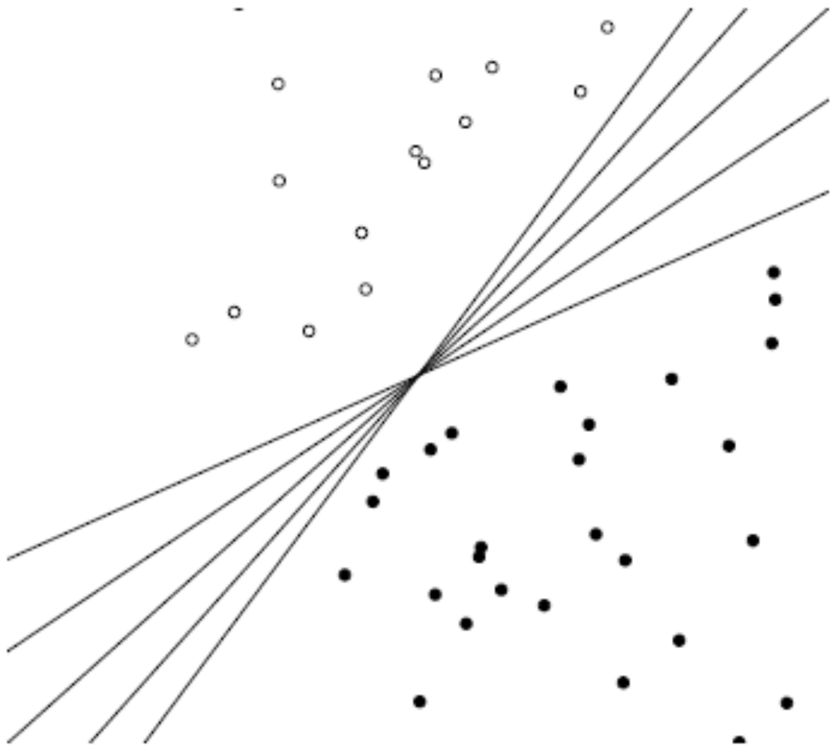
II. k nearest neighbor

III. Linear classifier:

Linear classifier:

The two learning methods **Naive Bayes** and **Rocchio** are instances of linear classifiers, perhaps most important group of text classifiers.

For a two-class classifier, the vector space is in two dimensions, a linear classifier is a line.



► **Figure 14.8** There are an infinite number of hyperplanes that separate two linearly separable classes.

Two-class Rocchio as a linear classifier

Lines or hyperplanes defined by:

$$\vec{w}^T\vec{x} = b$$

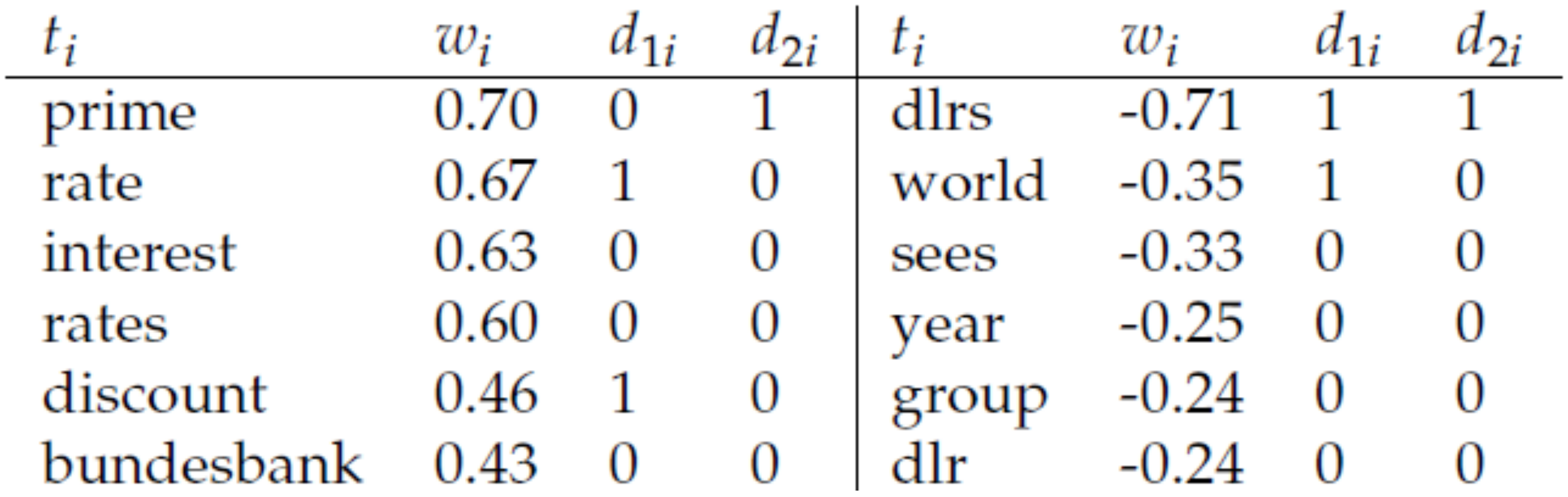
The classification rule of a linear classifier is to assign a document to c if $w_1x_1 + w_2x_2 > b$ and to $\bar{c}$ if $w_1x_1 + w_2x_2 \leq b$.

Two-class Rocchio as a linear classifier

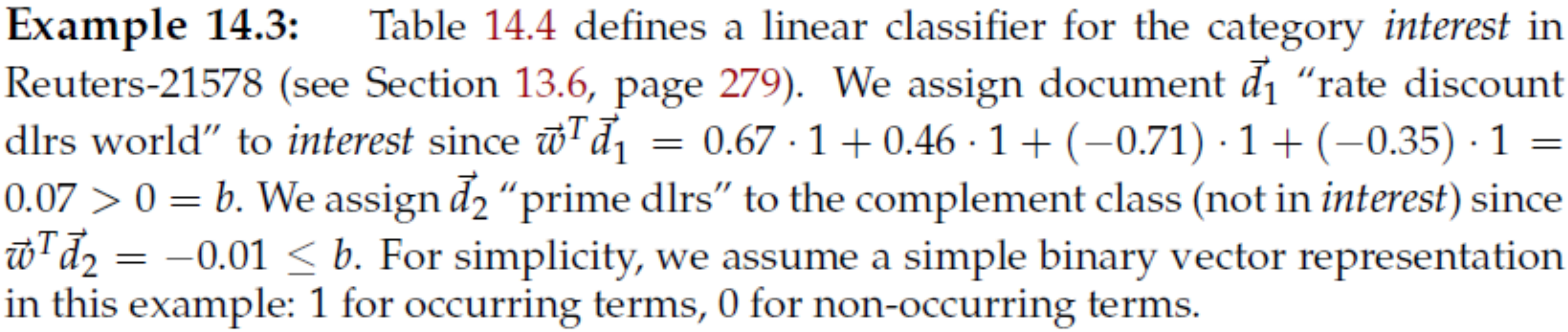
(x_1, x_2) is the 2D vector presentation of the document and (w_1, w_2) is the parameter vector that defines **the decision boundary**.

Example

A linear classifier for the class *interest* in Reuters-21578, The dimensions t_i and parameters w_i are listed below. The threshold $b = 0$. (negative weights are indicators for the competing class)



t_i	w_i	d_{1i}	d_{2i}	t_i	w_i	d_{1i}	d_{2i}
prime	0.70	0	1	dlrs	-0.71	1	1
rate	0.67	1	0	world	-0.35	1	0
interest	0.63	0	0	sees	-0.33	0	0
rates	0.60	0	0	year	-0.25	0	0
discount	0.46	1	0	group	-0.24	0	0
bundesbank	0.43	0	0	dlr	-0.24	0	0



Example 14.3: Table 14.4 defines a linear classifier for the category *interest* in Reuters-21578 (see Section 13.6, page 279). We assign document $\vec{d}_1$ "rate discount dlrs world" to *interest* since $\vec{w}^T\vec{d}_1 = 0.67 \cdot 1 + 0.46 \cdot 1 + (-0.71) \cdot 1 + (-0.35) \cdot 1 = 0.07 > 0 = b$. We assign $\vec{d}_2$ "prime dlrs" to the complement class (not in *interest*) since $\vec{w}^T\vec{d}_2 = -0.01 \leq b$. For simplicity, we assume a simple binary vector representation in this example: 1 for occurring terms, 0 for non-occurring terms.

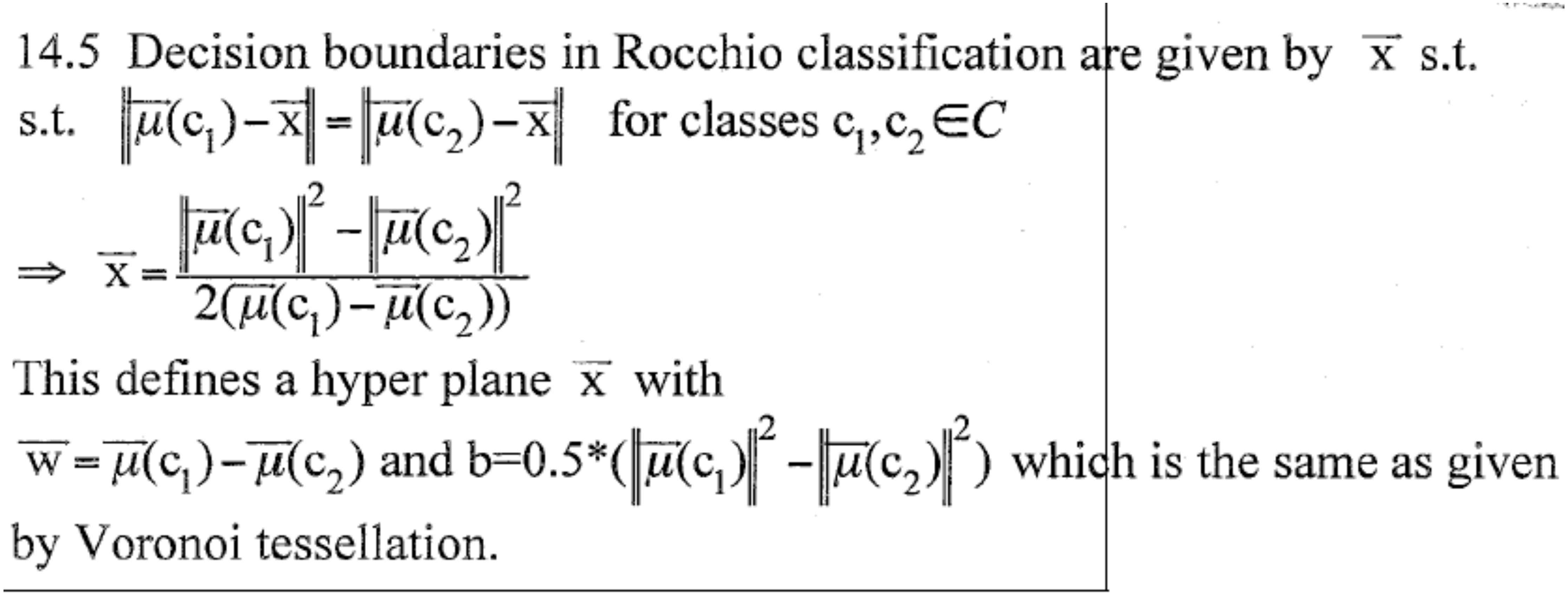
Rocchio is a linear classifier

Exercise

Show that Equation (14.4) defines a hyperplane $\vec{w}^T\vec{x} = b$ with $\vec{w} = \vec{\mu}(c_1) - \vec{\mu}(c_2)$ and $b = 0.5 * (|\vec{\mu}(c_1)|^2 - |\vec{\mu}(c_2)|^2)$

(14.4) $|\vec{\mu}(c_1) - \vec{x}| = |\vec{\mu}(c_2) - \vec{x}|$

Solution



14.5 Decision boundaries in Rocchio classification are given by $\vec{x}$ s.t.

s.t. $\|\vec{\mu}(c_1)-\vec{x}\| = \|\vec{\mu}(c_2)-\vec{x}\|$ for classes $c_1, c_2 \in C$

$$\Rightarrow \quad \vec{x} = \frac{\|\vec{\mu}(c_1)\|^2 - \|\vec{\mu}(c_2)\|^2}{2(\vec{\mu}(c_1)-\vec{\mu}(c_2))}$$

This defines a hyper plane $\vec{x}$ with

$\vec{w} = \vec{\mu}(c_1) - \vec{\mu}(c_2)$ and $b=0.5*(\|\vec{\mu}(c_1)\|^2 - \|\vec{\mu}(c_2)\|^2)$ which is the same as given by Voronoi tessellation.