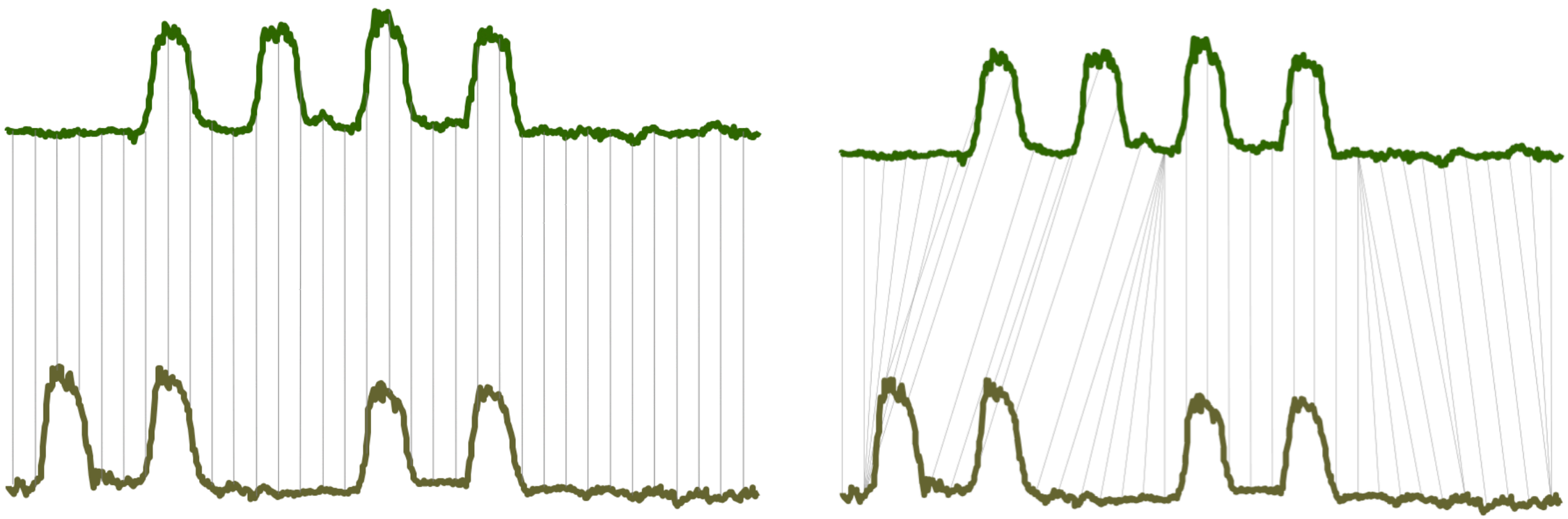
Lab6

CS562

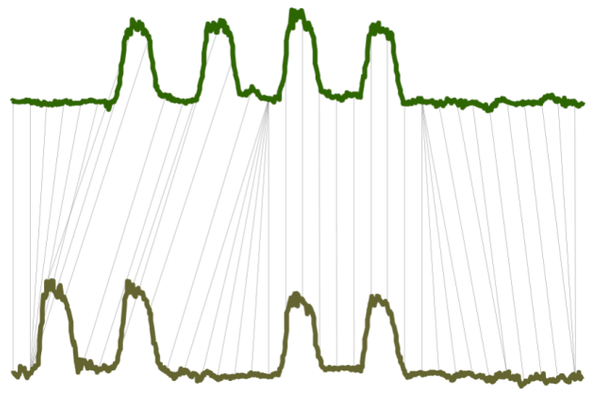
Time Series

ED vs DTW



Euclidean Distance vs Dynamic Type Warping

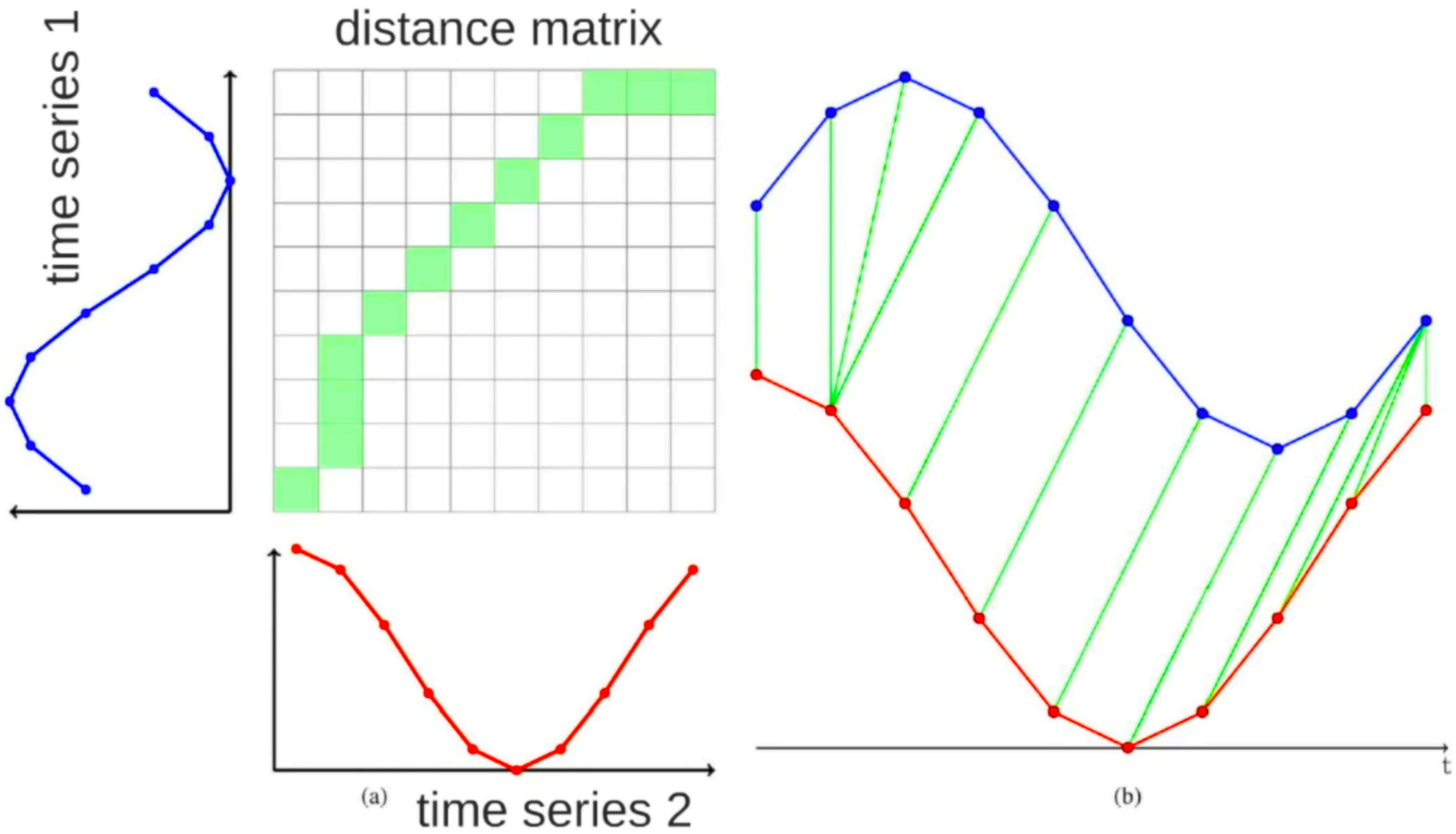
DTW



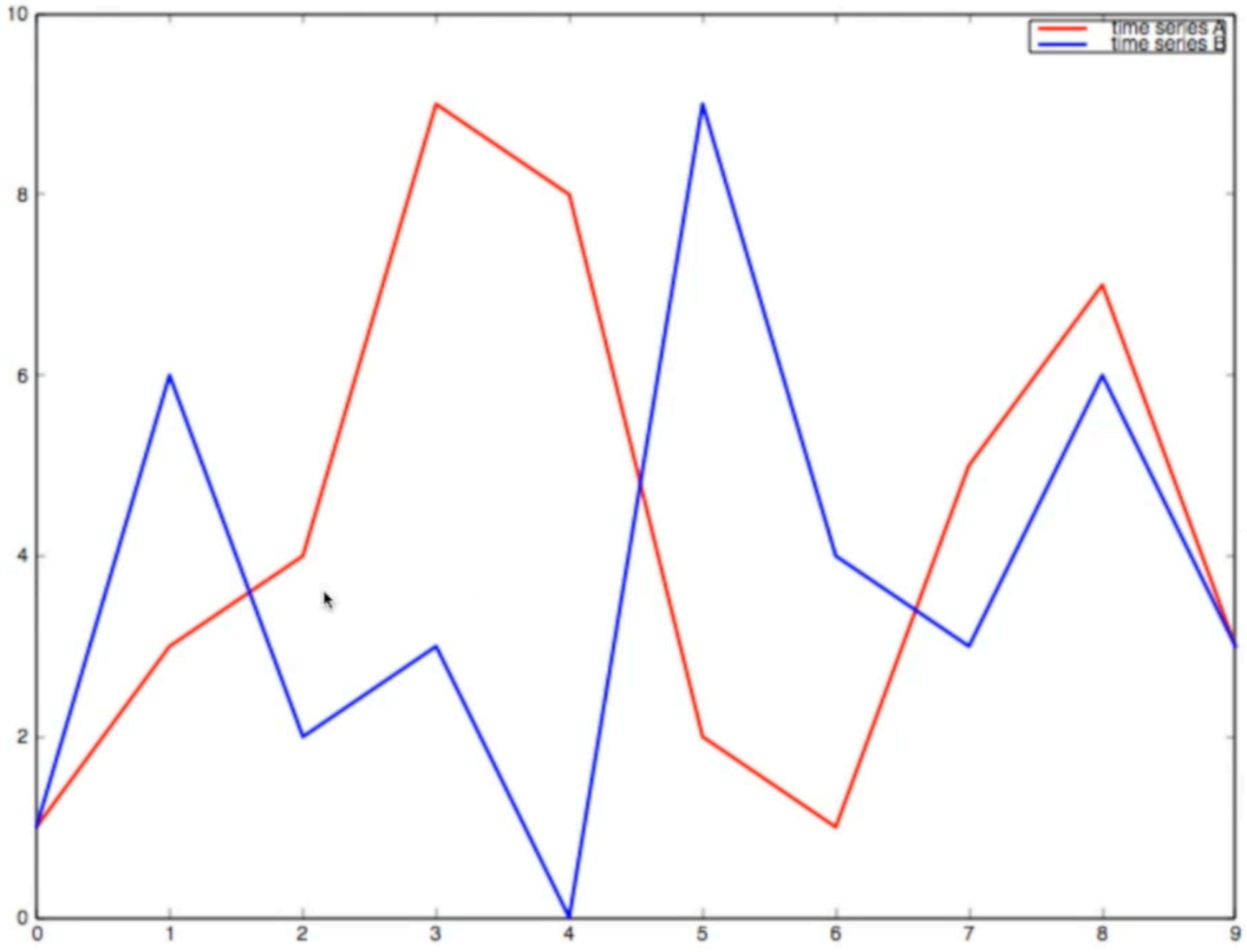
Dynamic time warping (DTW) finds an optimal alignment between two given (time-dependent) sequences.

Originally, DTW has been used to compare different speech patterns in automatic speech recognition. It is very commonly used in fields such as data mining and information retrieval.

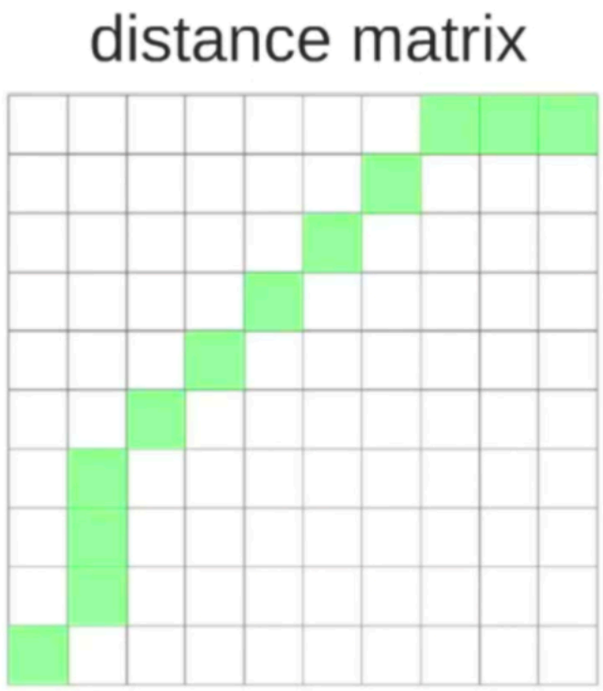
Compute DTW



DTW Example



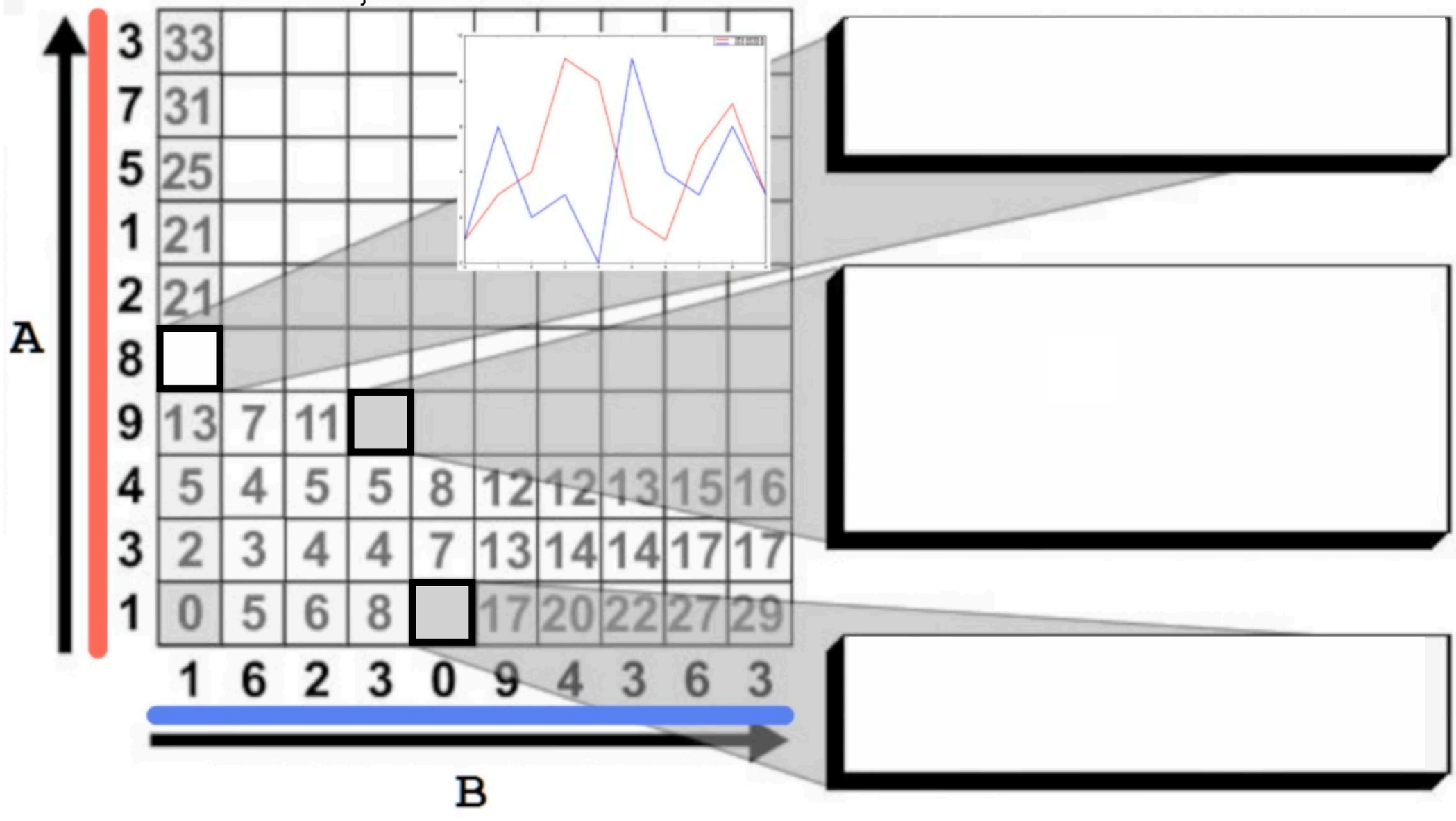
Distance Matrix



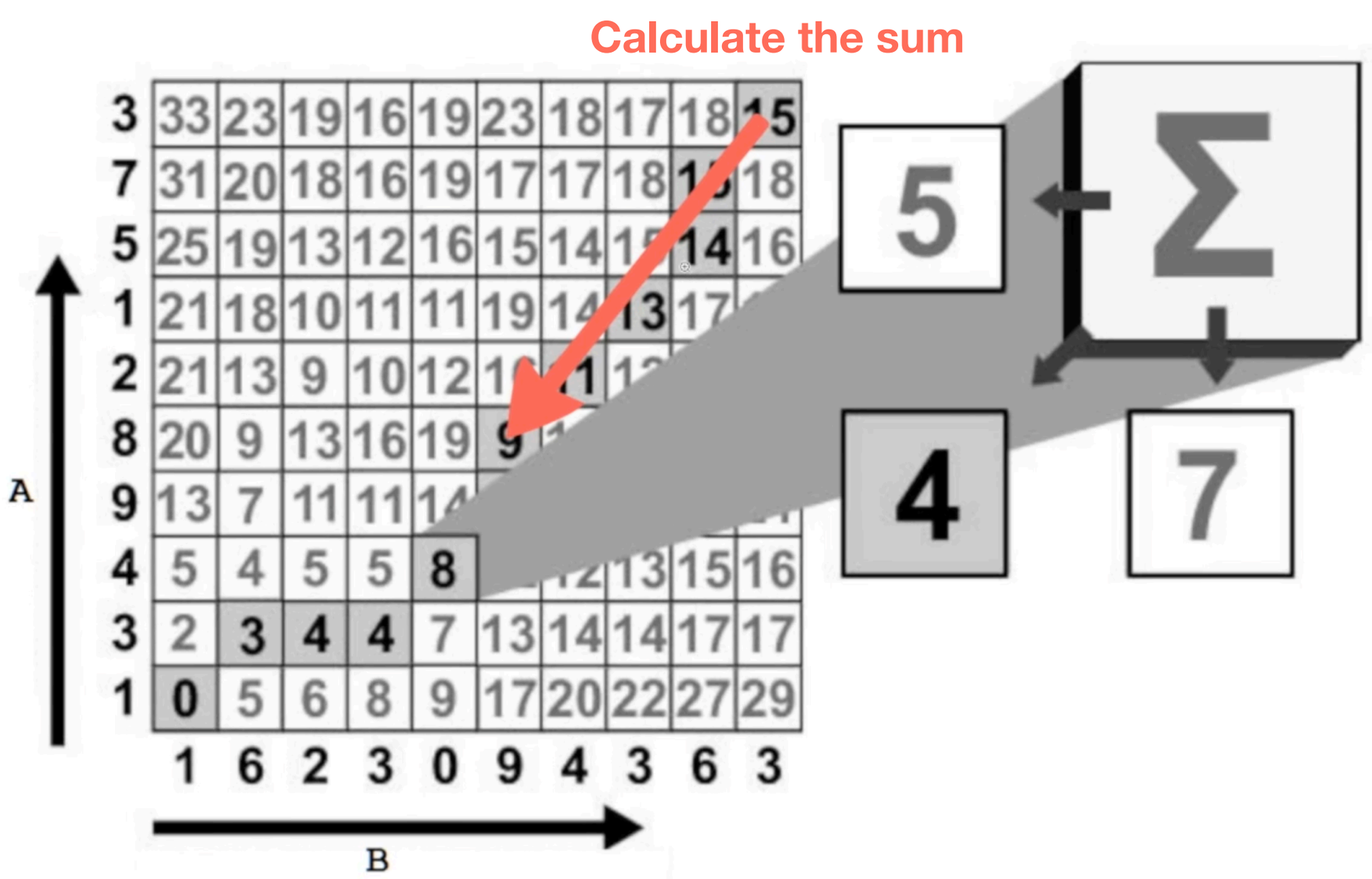
$$D(i, j) = | x_i - y_j | + \min \{ D(i - 1, j), D(i - 1, j - 1), D(i, j - 1) \}$$

Distance Matrix

$$D(i, j) = | x_i - y_j | + \min \{ D(i-1, j),\ D(i-1, j-1),\ D(i, j-1) \}$$



Distance Matrix



Text Indexing

Boolean Search

- Queries that use Boolean Search reply that documents either match or don’t

- Boolean queries often result in either too few (=0) or too many (>1000) results

- **Query 1**: “standard user dlink 650” → 200,000 hits
- **Query 2**: “standard user dlink 650 no card found”: → 0 hits

- It takes skill to come up with a query that produces a manageable number of hits

We need a way of assigning a score to a query/document pair

TF-IDF

- How many times the word occurs in the document
- Where the word occurs
- How important is the word – for example, “the” vs. “motorcycle”

TF-IDF Example

- D1 = "If it walks like a duck and quacks like a duck, it must be a duck.”
- D2 = "Beijing Duck is mostly prized for the thin, crispy duck skin with authentic versions of the dish serving mostly the skin.”
- D3 = "Bugs' ascension to stardom also prompted the Warner animators to recast Daffy Duck as the rabbit's rival, intensely jealous and determined to steal back the spotlight while Bugs remained indifferent to the duck's jealousy, or used it to his advantage. This turned out to be the recipe for the success of the duo.”
- D4 = "6:25 PM 1/7/2007 blog entry: I found this great recipe for Rabbit Braised in Wine on cookingforengineers.com.”
- D5 = "Last week Li has shown you how to make the Sechuan duck. Today we'll be making Chinese dumplings (Jiaozi), a popular dish that I had a chance to try last summer in Beijing. There are many recipies for Jiaozi."

Computing TF

After stemming: "*recipes = recipe*"

term	TF				
	D1	D2	D3	D4	D5
beijing		1			1
dish		1			1
duck	3	2	2		1
rabbit			1	1	
recipe			1	1	1
roast					

Normalizing TF

term	TF				
	D1	D2	D3	D4	D5
beijing		1			1
dish		1			1
duck	3	2	2		1
rabbit			1	1	
recipe			1	1	1
roast					

Normalizing: Divide by the maximum value of each column

term	TF normalized				
	D1	D2	D3	D4	D5
beijing	0	0.5	0	0	1
dish	0	0.5	0	0	1
duck	1	1	1	0	1
rabbit	0	0	0.5	1	0
recipe	0	0	0.5	1	1
roast	0	0	0	0	0

Computing IDF

Formula: idf(t) = log(N/(df + 1))

term	TF normalized					IDF
	D1	D2	D3	D4	D5	
beijing	0	0.5	0	0	1	0.398
dish	0	0.5	0	0	1	0.398
duck	1	1	1	0	1	0.097
rabbit	0	0	0.5	1	0	0.398
recipe	0	0	0.5	1	1	0.222
roast	0	0	0	0	0	0

Why log in the IDF formula?

N = 1.000.000

term	df_t	idf_t
calpurnia	1	1,000,000
animal	100	10,000
sunday	1,000	1,000
fly	10,000	100
under	100,000	10
the	1,000,000	1

More reasonable results

term	df_t	idf_t
calpurnia	1	6
animal	100	4
sunday	1,000	3
fly	10,000	2
under	100,000	1
the	1,000,000	0

Compute TF-IDF

term	TF normalized					IDF
	D1	D2	D3	D4	D5	
beijing	0	0.5	0	0	1	0.398
dish	0	0.5	0	0	1	0.398
duck	1	1	1	0	1	0.097
rabbit	0	0	0.5	1	0	0.398
recipe	0	0	0.5	1	1	0.222
roast	0	0	0	0	0	0

term	TFIDF				
	D1	D2	D3	D4	D5
beijing	0	0.199	0	0	0.398
dish	0	0.199	0	0	0.398
duck	0.097	0.097	0.097	0	0.097
rabbit	0	0	0.199	0.398	0
recipe	0	0	0.111	0.222	0.222
roast	0	0	0	0	0

Query

“Recipes for Beijing duck”

term	Q	Qidf
beijing	1	0.398
dish	0	0
duck	1	0.097
rabbit	0	0
recipe	1	0.222
roast	0	0

Which documents are the most relevant?

Query similarity

term	TFIDF				
	D1	D2	D3	D4	D5
beijing	0	0.199	0	0	0.398
dish	0	0.199	0	0	0.398
duck	0.097	0.097	0.097	0	0.097
rabbit	0	0	0.199	0.398	0
recipe	0	0	0.111	0.222	0.222
roast	0	0	0	0	0

term	Q	Qidf
beijing	1	0.398
dish	0	0
duck	1	0.097
rabbit	0	0
recipe	1	0.222
roast	0	0

$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}},$$

similarity				
D1	D2	D3	D4	D5
0.208	0.639	0.295	0.232	0.76