

---

度量 (metric) 是对一个集合里面的两个元素而言的, 是两点间距离的抽象。

范数 (norm) 是对一个元素而言的, 是实数的绝对值或复数的模这样的长度的抽象。

# Machine Learning

## Topic 4: Measuring Distance

# Why measure distance?

---

- Clustering requires distance measures.
- Local methods require a measure of “locality”
- Search engines require a measure of similarity

# What is a “metric”?

---

- A function of two values with these four qualities.

$d(x, y) = 0 \text{ iff } x = y$	( <sup>自反性</sup> reflexivity)
$d(x, y) \geq 0$	(non - negative)
$d(x, y) = d(y, x)$	(symmetry)
$d(x, y) + d(y, z) \geq d(x, z)$	(triangle inequality)

# What is a “norm”?

---

- Loosely, it is a function that applies a positive value to all vectors (except the 0 vector) in a vector space.
- 3 properties:

For all  $a \in F$  and  $u, v \in V$ , a function  $p : V \rightarrow F$

$$p(av) = |a| p(v) \quad (\text{positive scalability})$$

$$p(u) = 0 \quad \text{iff } u \text{ is the zero vector}$$

$$p(u) + p(v) \geq p(u + v) \quad (\text{triangle inequality})$$

## 2 definitions (AKA why this is confusing)

---

- A vector norm

A function that assigns a strictly positive value to all vectors in a vector space....except the 0 vector, which has a 0 assigned to it. (see previous slide)

- A normal vector

A vector is called a **normal** to another object if they are perpendicular to each other. So, a **normal vector** is perpendicular to (the tangent plane of) a surface at some point  $P$ .

# Metric == Norm??

---

- Every norm determines a metric.

Given a normed vector space, we can make a metric by saying

$$d(x, y) \equiv \|x - y\|$$

- Some metrics determine a norm.

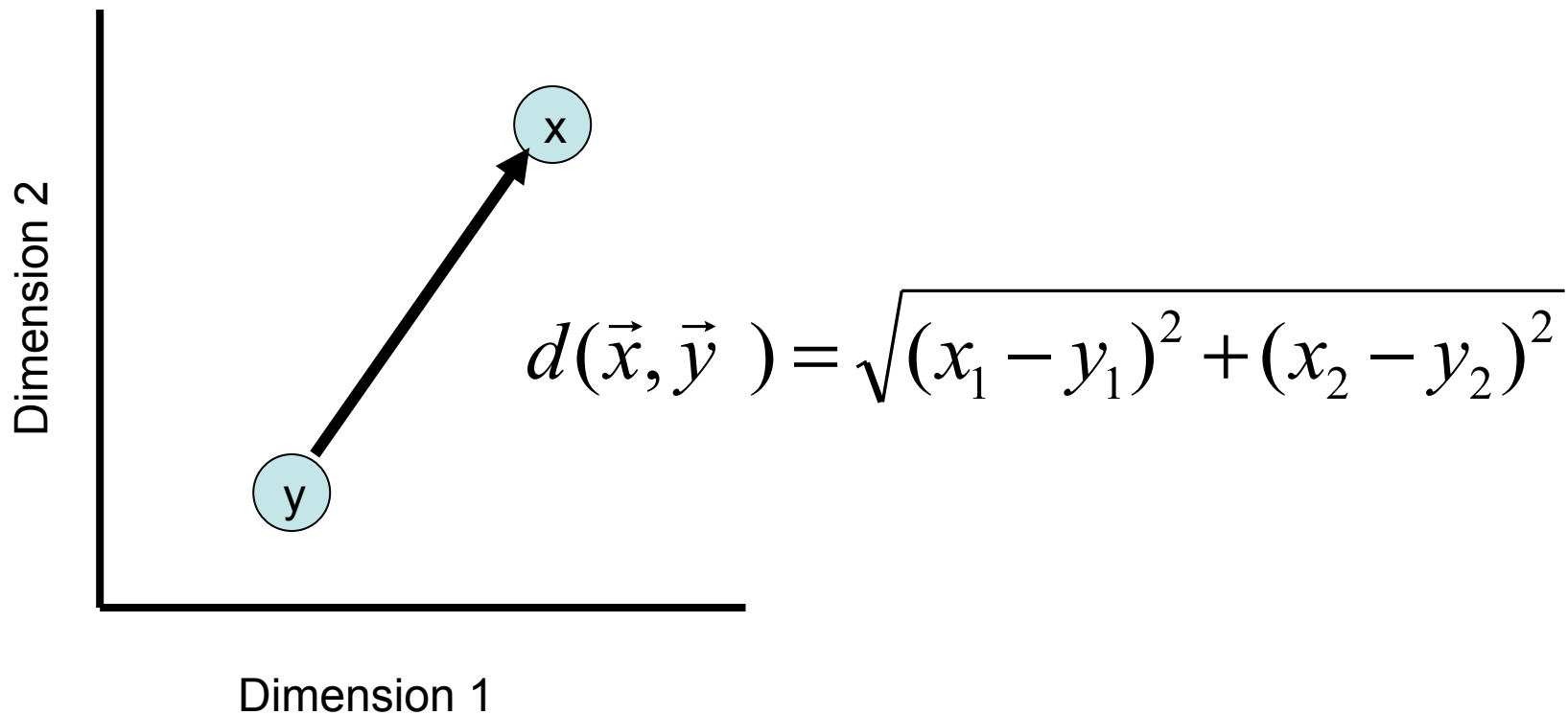
**If** the metric is on a vector space, you can define a norm by saying...

$$\|x\| \equiv d(x, 0)$$

# Euclidean Distance

---


- What people intuitively think of as “distance”
- Is it a metric?
- Is it a norm?



# Generalized Euclidean Distance

---

$n$  = the number of dimensions


$$d(\vec{x}, \vec{y}) = \left[ \sum_{i=1}^n |x_i - y_i|^2 \right]^{1/2}$$

where  $\vec{x} = \langle x_1, x_2, \dots, x_n \rangle$ ,

$$\vec{y} = \langle y_1, y_2, \dots, y_n \rangle$$

and  $\forall i (x_i, y_i \in \mathbb{R})$

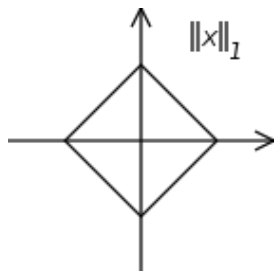


# $L^p$ norms

- $L^p$  norms are all special cases of this:

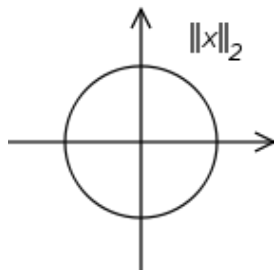
$$d(\vec{x}, \vec{y}) = \left[ \sum_{i=1}^n |x_i - y_i|^p \right]^{1/p}$$

$p$  changes the norm



$p = 1$  两点距离 = 两直角边之和

$$\|x\|_1 = L^1 \text{ norm} = \text{Manhattan Distance} : p = 1$$



两点距离 = 直线距离

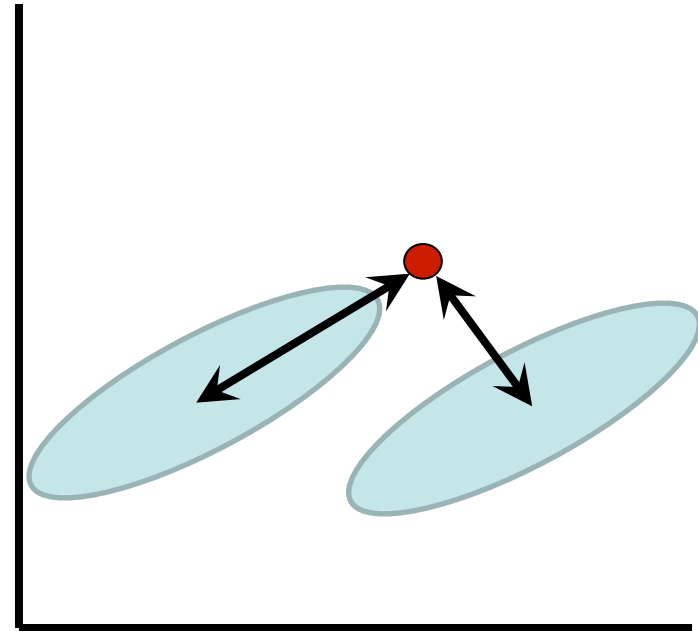
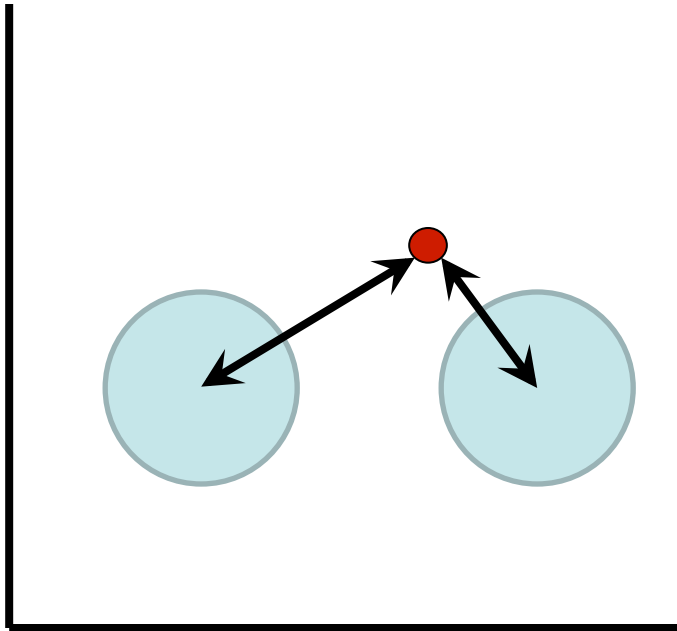
$$\|x\|_2 = L^2 \text{ norm} = \text{Euclidean Distance} : p = 2$$

不同位数的个数

$$\text{Hamming Distance} : p = 1 \text{ and } x_i, y_i \in \{0, 1\}$$

# Weighting Dimensions

---



- Put point in the cluster with the closest center of gravity
- Which cluster should the red point go in?
- How do I measure distance in a way that gives the “right” answer for both situations?

# Weighted Norms

---

- You can compensate by weighting your dimensions....

$$d(\vec{x}, \vec{y}) = \left[ \sum_{i=1}^n w_i |x_i - y_i|^p \right]^{1/p}$$

weighted norms只能表示轴平行于坐标轴的椭圆

This lets you turn your circle of equal-distance into an ellipse with axes parallel to the dimensions of the vectors.

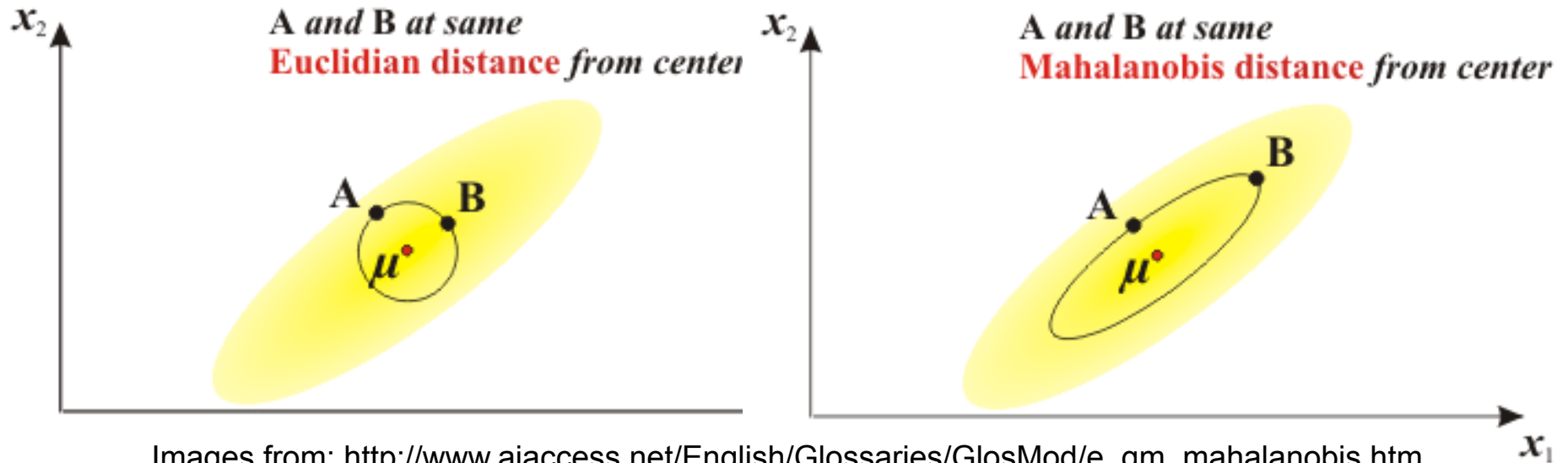
# Mahalanobis distance

The region of constant Mahalanobis distance around the mean of a distribution forms an ellipsoid.

在平均值周围以固定马氏距离分布的形成的是椭圆

这个椭圆的轴不必与坐标轴平行

The axes of this ellipsoid don't have to be parallel to the dimensions describing the vector



Images from: [http://www.aiaccess.net/English/Glossaries/GlosMod/e\\_gm\\_mahalanobis.htm](http://www.aiaccess.net/English/Glossaries/GlosMod/e_gm_mahalanobis.htm)

# Calculating Mahalanobis

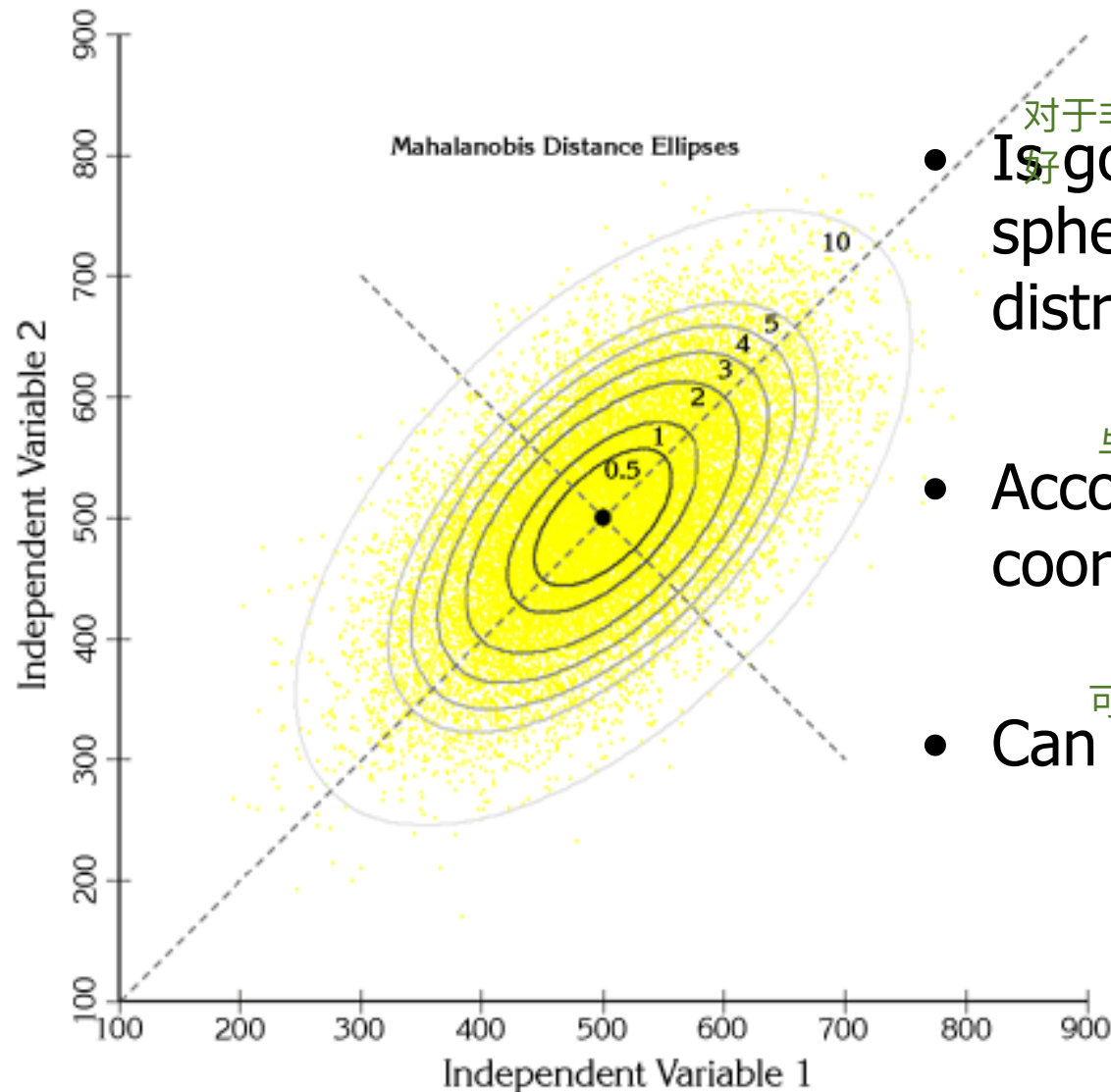
---

$$d(\vec{x}, \vec{y}) = \sqrt{(\vec{x} - \vec{y})^T S^{-1} (\vec{x} - \vec{y})}$$

- This matrix  $S^{-1}$  is called the “covariance” matrix and is calculated from the data distribution
- Let’s look at the demo here:

[http://www.aiaccess.net/English/Glossaries/GlosMod/e\\_gm\\_mahalanobis.htm#Animation%20Mahalanobis](http://www.aiaccess.net/English/Glossaries/GlosMod/e_gm_mahalanobis.htm#Animation%20Mahalanobis)

# Take-away on Mahalanobis



- 对于非球状分布比较好  
Is good for non-spherically symmetric distributions.
- 与坐标轴缩放有关  
Accounts for scaling of coordinate axes
- 可以退化成Euclidean  
Can reduce to Euclidean

# Metric, or not?

---

- Driving distance with 1-way streets



- Categorical Stuff :
  - Is distance (Jazz to Blues to Rock) no less than distance (Jazz to Rock)?

# Categorical Variables

---

- Consider feature vectors for genre & vocals:
  - Genre: {Blues, Jazz, Rock, Zydeco}
  - Vocals: {vocals, no vocals}

$s1 = \{\text{rock, vocals}\}$

$s2 = \{\text{jazz, no vocals}\}$

$s3 = \{\text{rock, no vocals}\}$

- Which two songs are more similar?



# One Solution: Hamming distance

Blues	Jazz	Rock	Zydeco	Vocals
0	0	1	0	1
0	1	0	0	0
0	0	1	0	0

$s1 = \{\text{rock, vocals}\}$

$s2 = \{\text{jazz, no\_vocals}\}$

$s3 = \{\text{rock, no\_vocals}\}$

Hamming Distance = number of bits different  
between binary vectors

# Hamming Distance

---

$$d(\vec{x}, \vec{y}) = \sum_{i=1}^n |x_i - y_i|$$

where  $\vec{x} = \langle x_1, x_2, \dots, x_n \rangle$ ,

$$\vec{y} = \langle y_1, y_2, \dots, y_n \rangle$$

and  $\forall i (x_i, y_i \in \{0, 1\})$

# Defining your own distance (an example)

---

How often does artist  $x$  quote artist  $y$ ?

## Quote Frequency

	Beethoven	Beatles	Liz Phair
Beethoven	7	0	0
Beatles	4	5	0
Liz Phair	?	1	2

Let's build a distance measure!

# Defining your own distance (an example)

---

	Beethoven	Beatles	Liz Phair
Beethoven	7	0	0
Beatles	4	5	0
Liz Phair	?	1	2

Quote frequency  $Q_f(x, y) = \text{value in table}$

$$\text{Distance } d(x, y) = 1 - \frac{Q_f(x, y)}{\sum_{z \in \text{Artists}} Q_f(x, z)}$$

# Missing data

---

- What if, for some category, on some examples, there is no value given?
- Approaches:
  - Discard all examples missing the category
  - Fill in the blanks with the mean value
  - Only use a category in the distance measure if both examples give a value

# Dealing with missing data

---

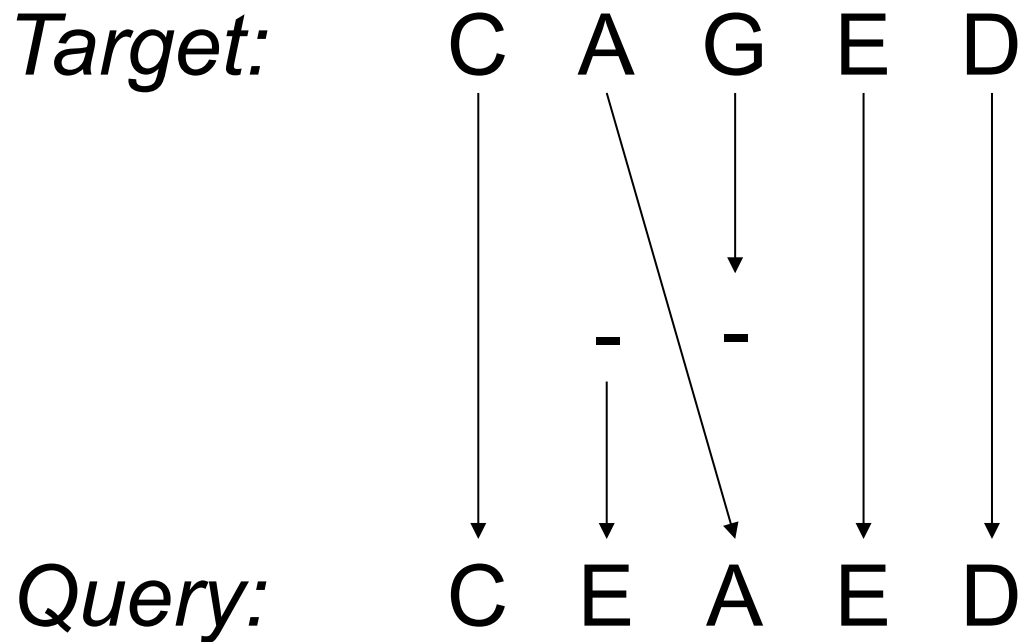
$$w_i = \begin{cases} 0, & \text{if both } x_i \text{ and } y_i \text{ are defined} \\ 1, & \text{else} \end{cases}$$

$$d(\vec{x}, \vec{y}) = \frac{n}{n - \sum_{i=1}^n w_i} \left[ \sum_{i=1}^n \phi(x_i, y_i) \right]$$

# Edit Distance

---

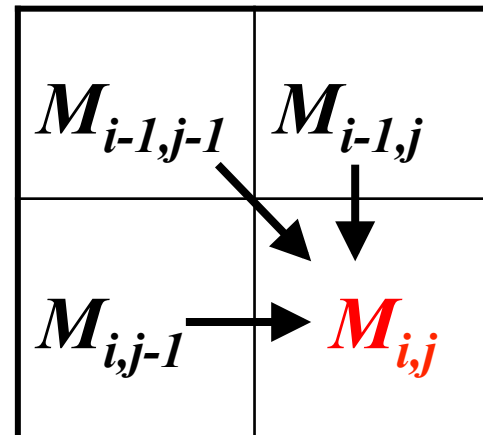
- Query = string from finite alphabet
- Target = string from finite alphabet
- Cost of Edits = Distance



# Levenshtein distance

$$M_{i,j} = \min \begin{cases} M_{i-1,j} + 1 & \text{Insert} \\ M_{i,j-1} + 1 & \text{Delete} \\ M_{i-1,j-1} + \mu(s_i, q_j) & \text{Match} \end{cases}$$

$$\mu(s_i, q_j) = \begin{cases} 0 & \text{if } s_i = q_j \\ 1 & \text{otherwise} \end{cases}$$





# Example Levenshtein Distance

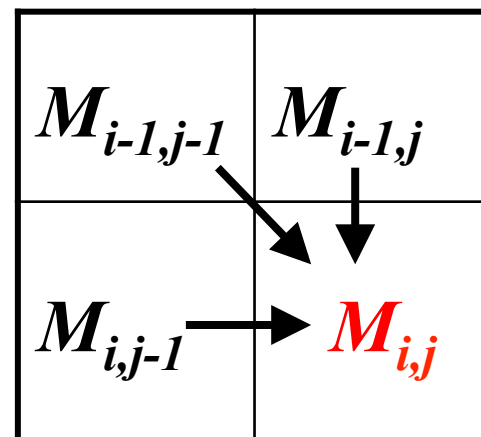
---

**DISTANCE**      $d(S, Q) = (M_{I,J})$

		<b>F</b>	<b>R</b>	<b>O</b>	<b>G</b>
	0	1	2	3	4
<b>D</b>	1	1	2	3	4
<b>O</b>	2	2	2	2	3
<b>G</b>	3	3	3	3	2

# (Somewhat more) General Edit Distance

$$M_{i,j} = \min \begin{cases} M_{i-1,j} + \mu(-, q_j) & \text{Insert} \\ M_{i,j-1} + \mu(s_i, -) & \text{Delete} \\ M_{i-1,j-1} + \mu(s_i, q_j) & \text{Match} \end{cases}$$



# One more distance measure

---

- Kullback–Leibler divergence
  - a non-symmetric measure of the difference between two probability distributions
  - not a metric, since it is not symmetric
  - Here's the definition of KL divergence for discrete probability distributions  $P$  and  $Q$

$$D_{KL}(P \parallel Q) = \sum_i \ln \left( \frac{P(i)}{Q(i)} \right) P(i)$$

# KL Divergence as Cross Entropy

---

$$\begin{aligned} D_{KL}(P \parallel Q) &= \sum_i \ln \left( \frac{P(i)}{Q(i)} \right) P(i) \\ &= \sum_i (\ln(P(i)) - \ln(Q(i))) P(i) \\ &= \sum_i P(i) \ln P(i) - \sum_i P(i) \ln Q(i) \end{aligned}$$

# Some take-away thoughts

---

- Many machine learning methods are helped by having a distance measure
- Some methods require metrics
- Not all measures are metrics
- Some common distance measures:
  - “P-norms”: Euclidean, Manhattan
  - “Edit distance”: Levenshtein
  - KL Divergence
  - Mahalanobis