
Distance & Similarity

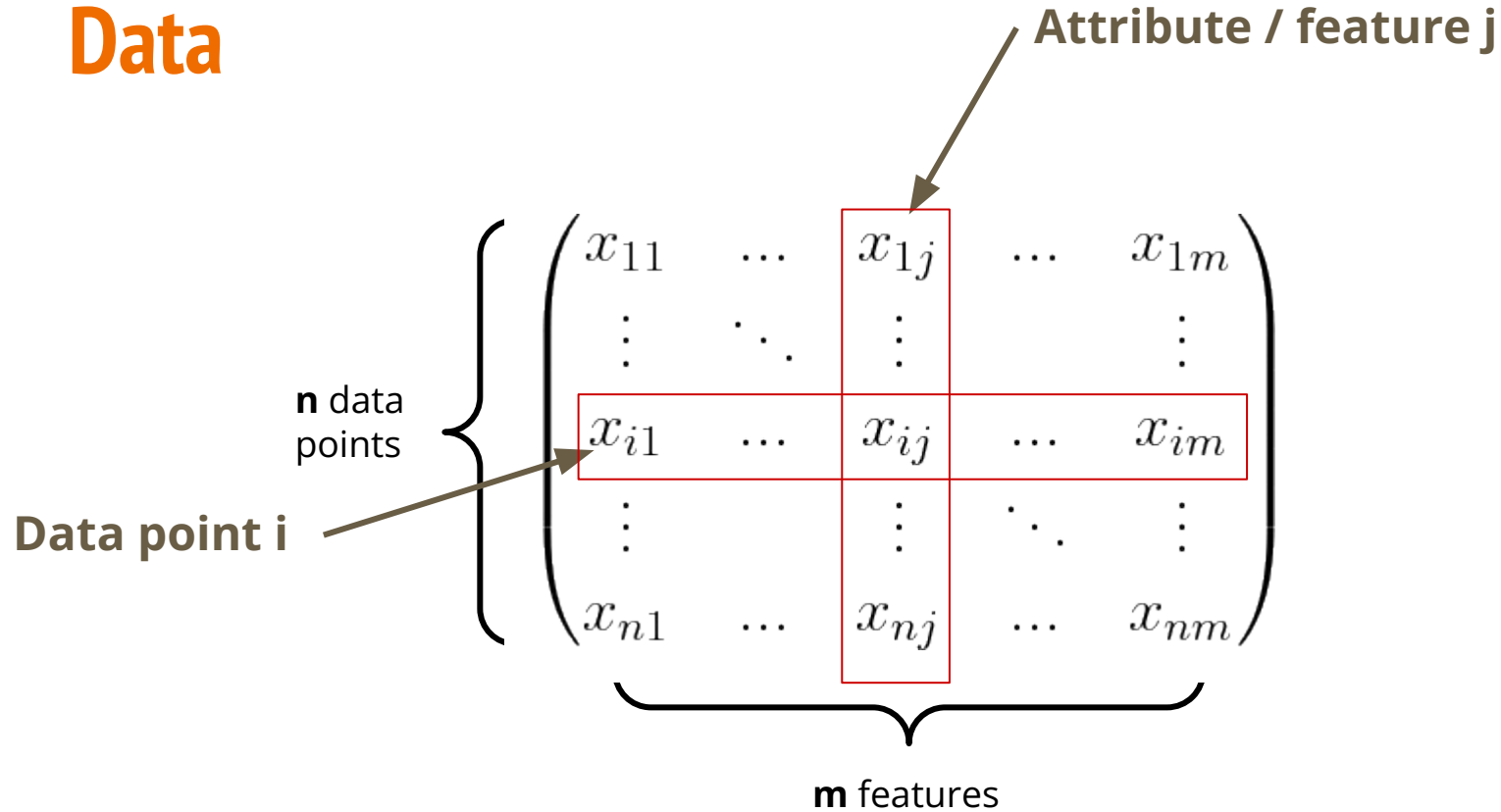
— Boston University CS 506 - Lance Galletti —

Data

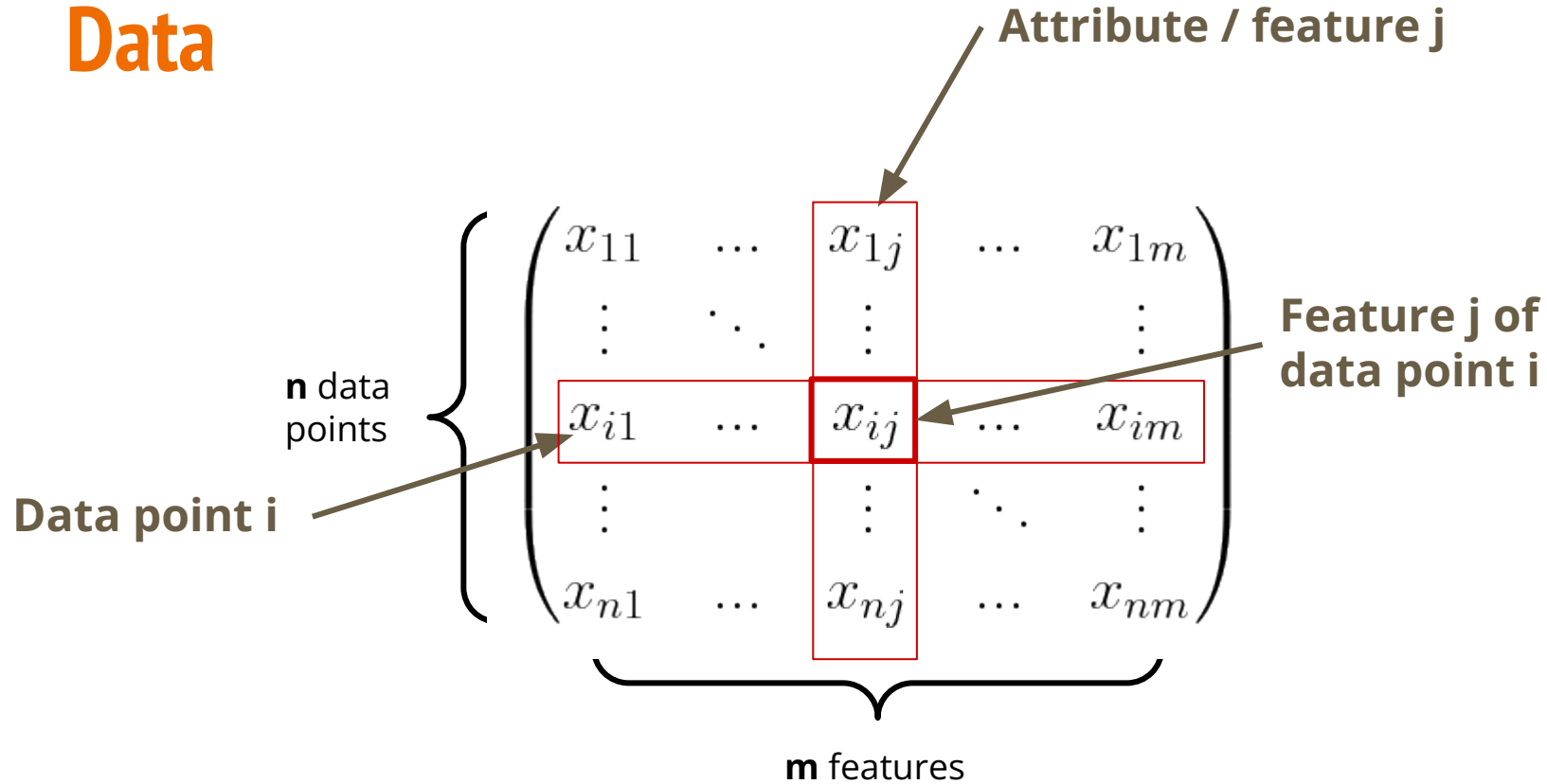
$$\begin{array}{c} \text{n data} \\ \text{points} \end{array} \left\{ \begin{pmatrix} x_{11} & \dots & x_{1j} & \dots & x_{1m} \\ \vdots & \ddots & \vdots & & \vdots \\ x_{i1} & \dots & x_{ij} & \dots & x_{im} \\ \vdots & & \vdots & \ddots & \vdots \\ x_{n1} & \dots & x_{nj} & \dots & x_{nm} \end{pmatrix} \right.$$

$\underbrace{\hspace{10em}}$
m features

Data



Data



Refund	Marital Status	Income	Age
1	Single	125k	25
0	Married	100k	27
0	Single	70k	22
1	Married	120k	30
0	Divorced	90k	28
0	Married	60k	37
1	Divorced	220k	24
0	Single	85k	23
0	Married	75k	23
0	Single	90k	26

Feature Space

From our data we can generate a **feature space** of all possible values for the set of features in our data.

name	age	balance
Jane	25	150
John	30	100

Feature Space

From our data we can generate a **feature space** of all possible values for the set of features in our data.

name	age	balance
Jane	25	150
John	30	100



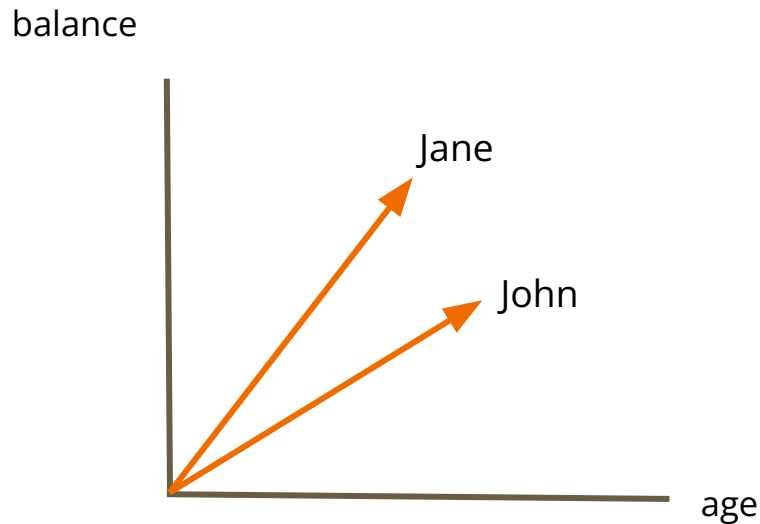
balance



Feature Space

From our data we can generate a **feature space** of all possible values for the set of features in our data.

name	age	balance
Jane	25	150
John	30	100



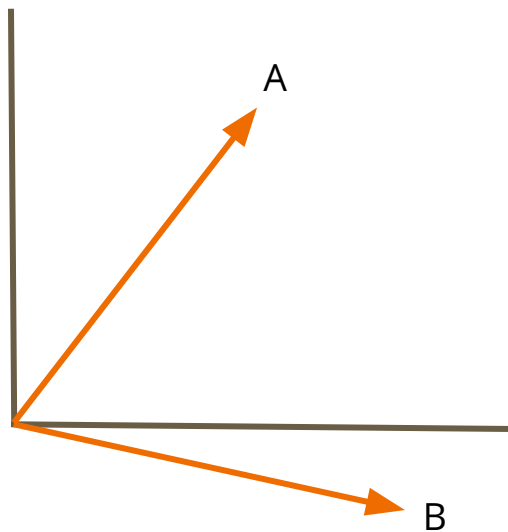
Our feature space is the Euclidean plane

Dissimilarity

In order to uncover interesting structure from our data, we need a way to **compare** data points.

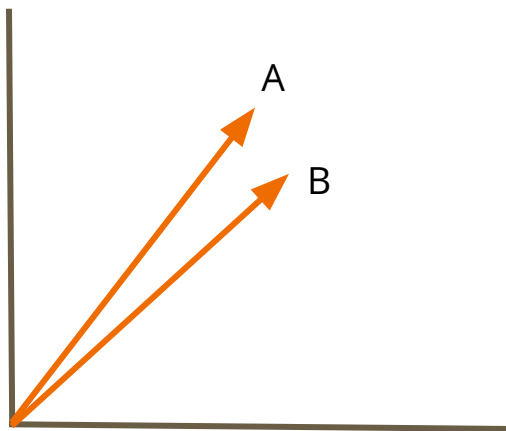
A **dissimilarity function** is a function that takes two objects (data points) and returns a **large value** if these objects are **dissimilar**.

Dissimilarity



$\text{dissim}(A, B)$ is large

Dissimilarity



$\text{dissim}(A, B)$ is small

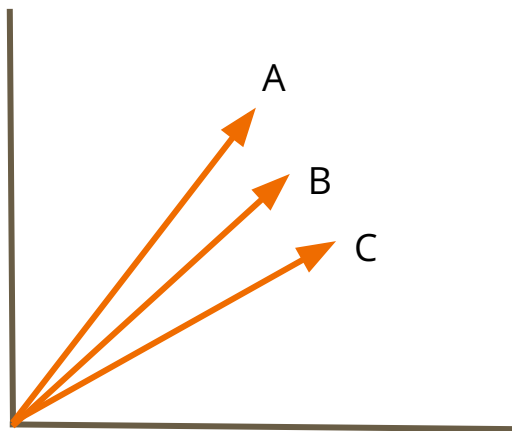
Distance

A special type of dissimilarity function is a **distance** function

d is a distance function if and only if:

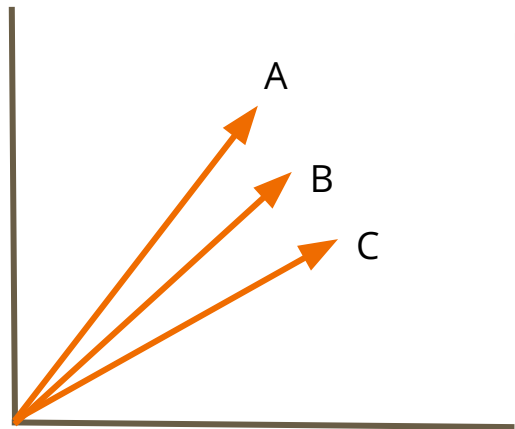
- $d(i, j) = 0$ if and only if $i = j$
- $d(i, j) = d(j, i)$
- $d(i, j) \leq d(i, k) + d(k, j)$

We don't **need** a distance function to compare data points, but why would we prefer using a distance function?



$\text{dissim}(A, B)$ is small

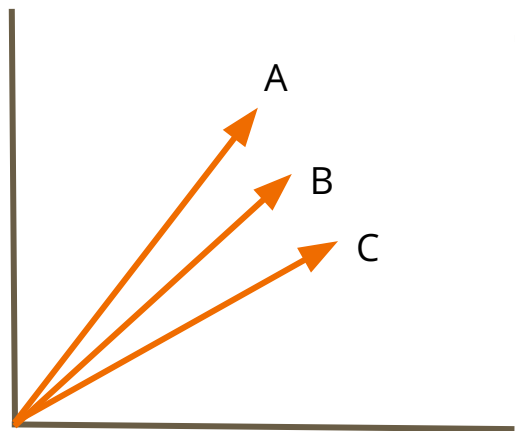
$\text{dissim}(B, C)$ is small



$\text{dissim}(A, B)$ is small

$\text{dissim}(B, C)$ is small

**$\text{dissim}(A, C)$ not
necessarily small**



$d(A, B)$ is small

$d(B, C)$ is small

**Triangle inequality
guarantees $d(A, C)$ small**

Minkowski Distance

For \mathbf{x}, \mathbf{y} points in \mathbf{d} -dimensional real space

i.e. $\mathbf{x} = [\mathbf{x}_1, \dots, \mathbf{x}_d]$ and $\mathbf{y} = [\mathbf{y}_1, \dots, \mathbf{y}_d]$

$\mathbf{p} \geq 1$

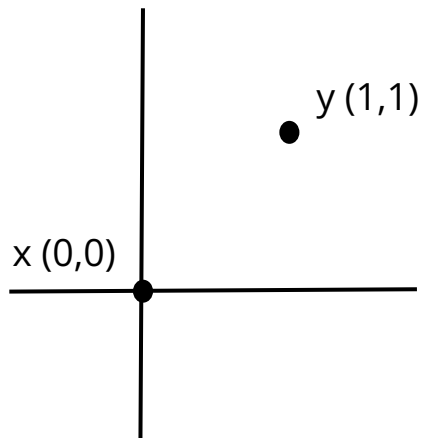
$$L_p(x, y) = \left(\sum_{i=1}^d |x_i - y_i|^p \right)^{\frac{1}{p}}$$

When $\mathbf{p} = 2$ -> Euclidean Distance

When $\mathbf{p} = 1$ -> Manhattan Distance

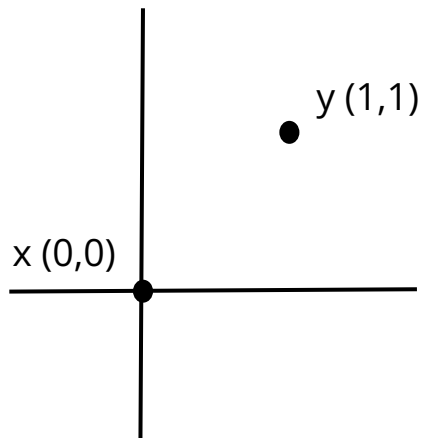
Example

$d = 2$



Example

$d = 2$

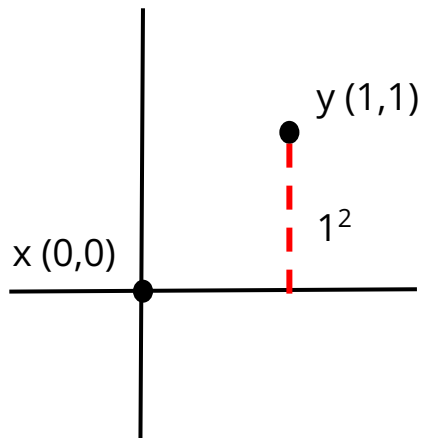


$p = 2$

$$L_p(x, y) = \left(\sum_{i=1}^d |x_i - y_i|^p \right)^{\frac{1}{p}}$$

Example

$d = 2$

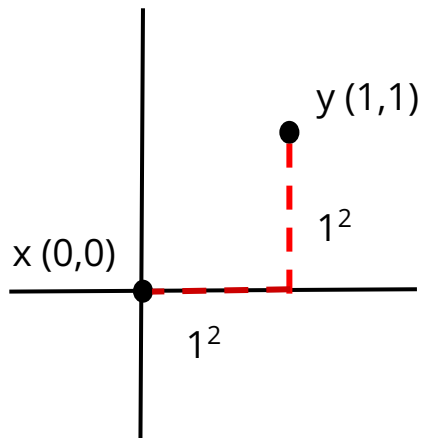


$p = 2$

$$L_p(x, y) = \left(\sum_{i=1}^d |x_i - y_i|^p \right)^{\frac{1}{p}}$$

Example

$d = 2$

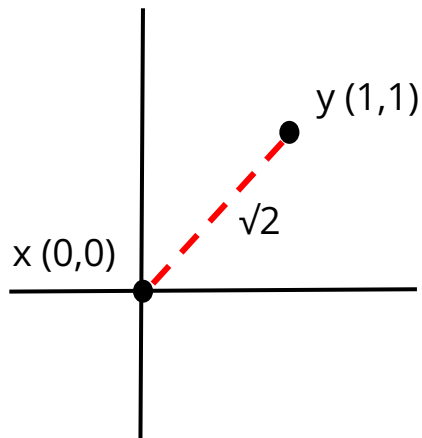


$p = 2$

$$L_p(x, y) = \left(\sum_{i=1}^d |x_i - y_i|^p \right)^{\frac{1}{p}}$$

Example

d = 2

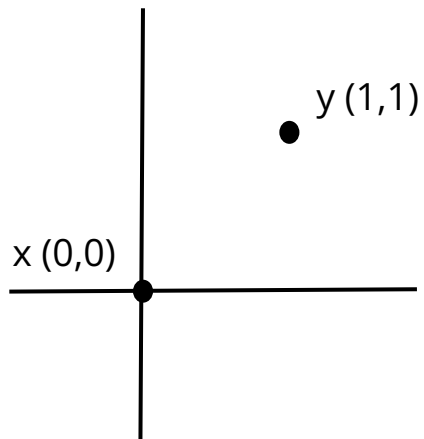


p = 2

$$L_p(x, y) = \left(\sum_{i=1}^d |x_i - y_i|^p \right)^{\frac{1}{p}}$$

Example

$d = 2$

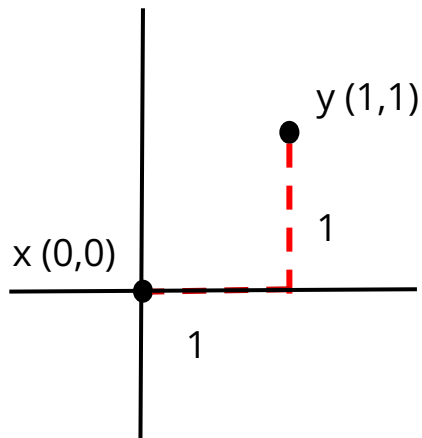


$p = 1$

$$L_p(x, y) = \left(\sum_{i=1}^d |x_i - y_i|^p \right)^{\frac{1}{p}}$$

Example

$d = 2$

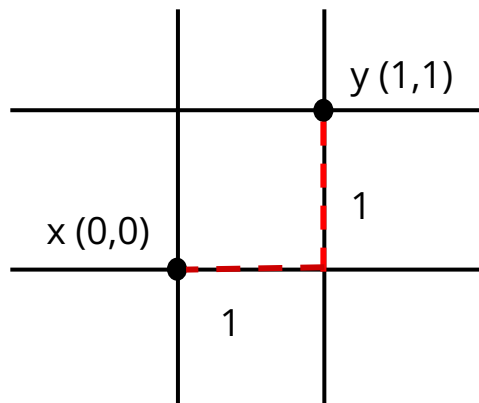


$p = 1$

$$L_p(x, y) = \left(\sum_{i=1}^d |x_i - y_i|^p \right)^{\frac{1}{p}}$$

Example

$d = 2$

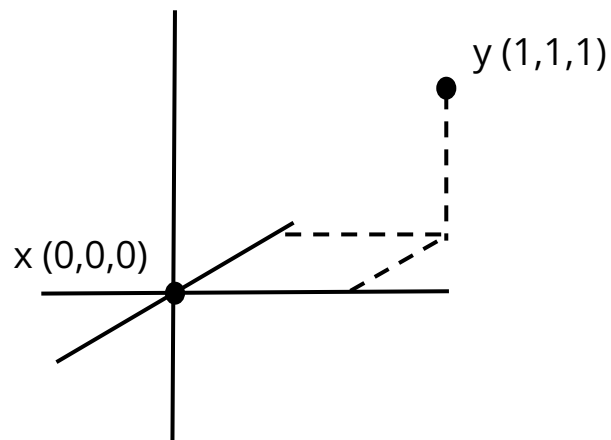


$p = 1$

$$L_p(x, y) = \left(\sum_{i=1}^d |x_i - y_i|^p \right)^{\frac{1}{p}}$$

Example

$d = 3$

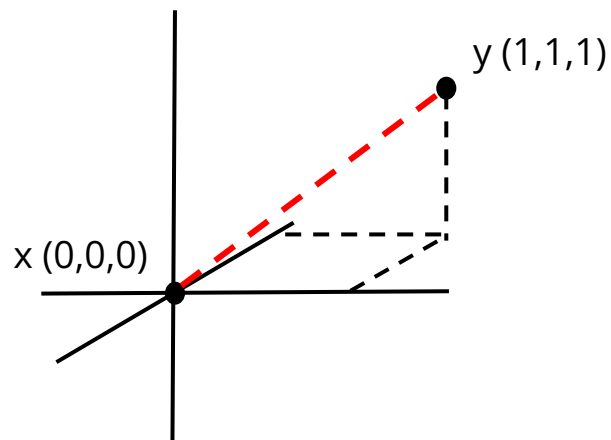


$p = 2$

$$L_p(x, y) = \left(\sum_{i=1}^d |x_i - y_i|^p \right)^{\frac{1}{p}}$$

Example

$d = 3$

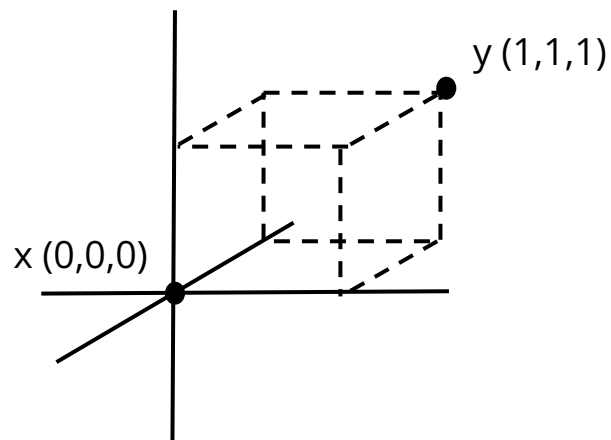


$p = 2$

$$L_p(x, y) = \left(\sum_{i=1}^d |x_i - y_i|^p \right)^{\frac{1}{p}}$$

Example

$d = 3$

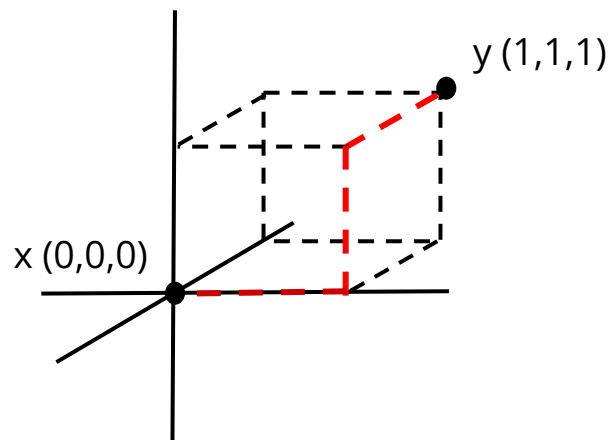


$p = 1$

$$L_p(x, y) = \left(\sum_{i=1}^d |x_i - y_i|^p \right)^{\frac{1}{p}}$$

Example

$d = 3$



$p = 1$

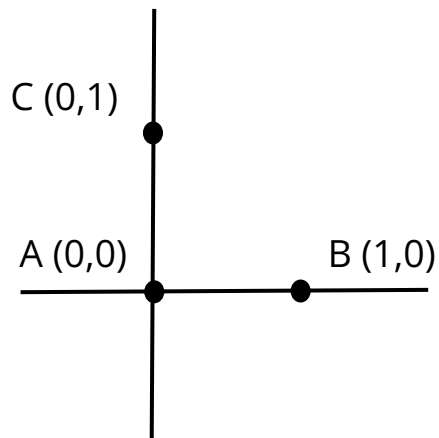
$$L_p(x, y) = \left(\sum_{i=1}^d |x_i - y_i|^p \right)^{\frac{1}{p}}$$

Minkowski Distance

Is L_p a distance function when $0 < p < 1$?

Minkowski Distance

Is L_p a distance function when $0 < p < 1$?

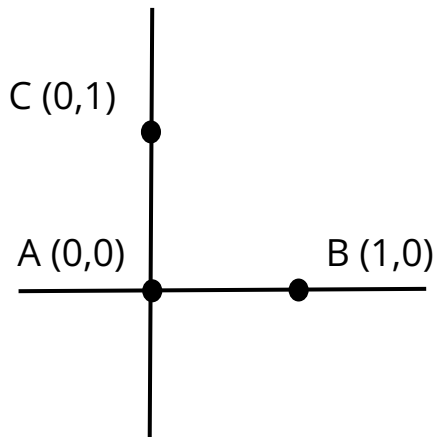


Minkowski Distance

Is L_p a distance function when $0 < p < 1$?

$$D(B,A) = D(A, C) = 1$$

$$D(B, C) = 2^{1/p}$$



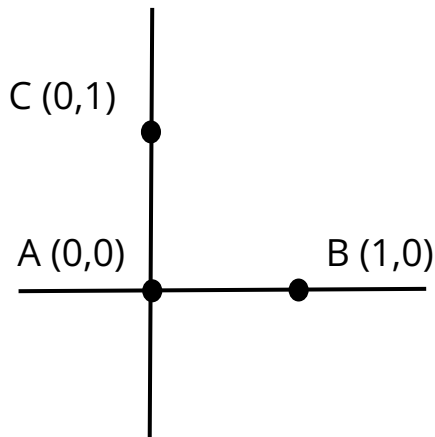
Minkowski Distance

Is L_p a distance function when $0 < p < 1$?

$$D(B,A) + D(A, C) = 2$$

$$D(B, C) = 2^{1/p}$$

But... if $p < 1$ then $1/p > 1$

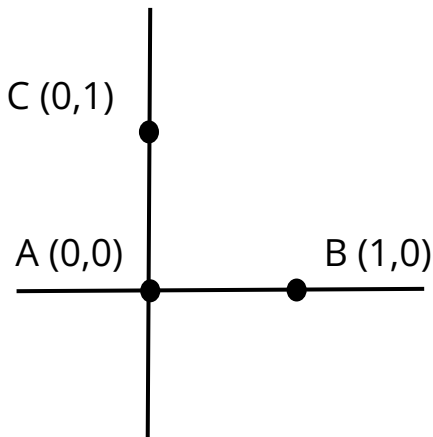


Minkowski Distance

Is L_p a distance function when $0 < p < 1$?

$$D(B,A) + D(A, C) = 2$$

$$D(B, C) = 2^{1/p}$$



So $D(B, C) > D(B, A) + D(A, C)$ which violates the triangle inequality

Jaccard Similarity

How similar are the following documents?

	w_1	w_2	...	w_d
x	1	0	...	1
y	1	1	...	0

Jaccard Similarity

One way is to use the Manhattan distance which will return the size of the set difference

	w_1	w_2	...	w_d
x	1	0	...	1
y	1	1	...	0

$$L_1(x, y) = \sum_{i=1}^d |x_i - y_i|$$

Jaccard Similarity

One way is to use the Manhattan distance which will return the size of the set difference

	w_1	w_2	...	w_d
x	1	0	...	1
y	1	1	...	0

$$L_1(x, y) = \sum_{i=1}^d |x_i - y_i|$$

Will only be 1 when $x_i \neq y_i$

Jaccard Similarity

But how can we distinguish between these two cases?

	w_1	w_2	...	w_{d-1}	w_d
x	1	1	1	0	1
y	1	1	1	1	0

Only differ on the last two words

	w_1	w_2
x	0	1
y	1	0

Completely different

Jaccard Similarity

But how can we distinguish between these two cases?

	w_1	w_2	...	w_{d-1}	w_d
x	1	1	1	0	1
y	1	1	1	1	0

Only differ on the last two words

	w_1	w_2
x	0	1
y	1	0

Completely different

Both have Manhattan distance of 2

Jaccard Similarity

We need to account for the size of the intersection!

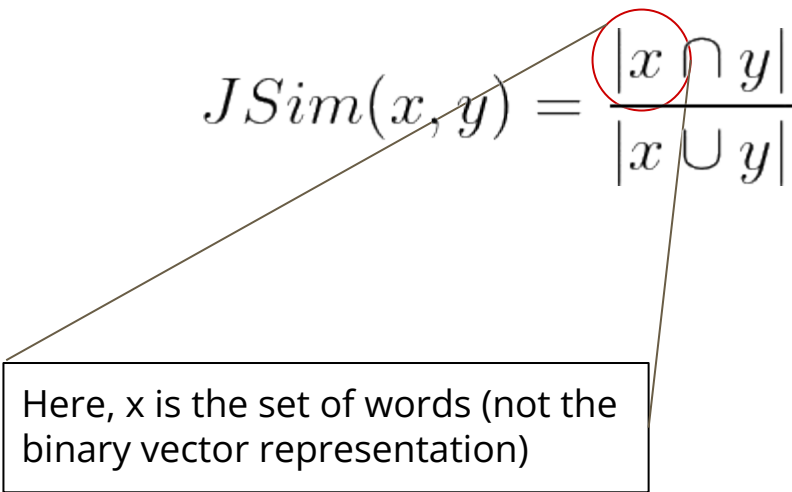
Given two documents x and y :

$$JSim(x, y) = \frac{|x \cap y|}{|x \cup y|}$$

Jaccard Similarity

We need to account for the size of the intersection!

Given two documents x and y :

$$JSim(x, y) = \frac{|x \cap y|}{|x \cup y|}$$
A diagram consisting of two thin black lines. One line starts from the top-left corner of a rectangular callout box and points to the numerator of the Jaccard Similarity formula. The other line starts from the bottom-right corner of the same box and points to the denominator. This indicates that the variables x and y in the formula refer to the sets of words described in the box.

Here, x is the set of words (not the binary vector representation)

$$JDist(x, y) = 1 - \frac{|x \cap y|}{|x \cup y|}$$

Jaccard Similarity

$$JDist(x, y) = 1 - \frac{|x \cap y|}{|x \cup y|}$$

	w_1	w_2	...	w_{d-1}	w_d
x	1	1	1	0	1
y	1	1	1	1	0

Only differ on the last two words

	w_1	w_2
x	0	1
y	1	0

Completely different

What is the jaccard distance in each?

Jaccard Similarity

$$JDist(x, y) = 1 - \frac{|x \cap y|}{|x \cup y|}$$

Here, x is the set of words (not the binary vector representation)

Cosine Similarity

A **similarity** function is a function that takes two objects (data points) and returns a **large value** if these objects are **similar**.

$$s(\mathbf{x}, \mathbf{y}) = \cos(\theta)$$

where θ is the angle between \mathbf{x} and \mathbf{y}

Cosine Similarity

A **similarity** function is a function that takes two objects (data points) and returns a **large value** if these objects are **similar**.

$$s(\mathbf{x}, \mathbf{y}) = \cos(\theta)$$

where θ is the angle between \mathbf{x} and \mathbf{y}

two proportional vectors have a cosine similarity of:

Cosine Similarity

A **similarity** function is a function that takes two objects (data points) and returns a **large value** if these objects are **similar**.

$$s(\mathbf{x}, \mathbf{y}) = \cos(\theta)$$

where θ is the angle between \mathbf{x} and \mathbf{y}

two proportional vectors have a cosine similarity of: 1

Cosine Similarity

A **similarity** function is a function that takes two objects (data points) and returns a **large value** if these objects are **similar**.

$$s(\mathbf{x}, \mathbf{y}) = \cos(\theta)$$

where θ is the angle between \mathbf{x} and \mathbf{y}

two proportional vectors have a cosine similarity of: 1

two orthogonal vectors have a similarity of:

Cosine Similarity

A **similarity** function is a function that takes two objects (data points) and returns a **large value** if these objects are **similar**.

$$s(\mathbf{x}, \mathbf{y}) = \cos(\theta)$$

where θ is the angle between \mathbf{x} and \mathbf{y}

two proportional vectors have a cosine similarity of: 1

two orthogonal vectors have a similarity of: 0

Cosine Similarity

A **similarity** function is a function that takes two objects (data points) and returns a **large value** if these objects are **similar**.

$$s(\mathbf{x}, \mathbf{y}) = \cos(\theta)$$

where θ is the angle between \mathbf{x} and \mathbf{y}

two proportional vectors have a cosine similarity of: 1

two orthogonal vectors have a similarity of: 0

two opposite vectors have a similarity of:

Cosine Similarity

A **similarity** function is a function that takes two objects (data points) and returns a **large value** if these objects are **similar**.

$$s(\mathbf{x}, \mathbf{y}) = \cos(\theta)$$

where θ is the angle between \mathbf{x} and \mathbf{y}

two proportional vectors have a cosine similarity of: 1

two orthogonal vectors have a similarity of: 0

two opposite vectors have a similarity of: -1

Cosine Similarity

To get a corresponding **dissimilarity** function, we can usually try

$$d(x, y) = 1 / s(x, y)$$

or

$$d(x, y) = k - s(x, y) \text{ for some } k$$

Here, we can use

$$d(x, y) = 1 - s(x, y)$$

Cosine Similarity

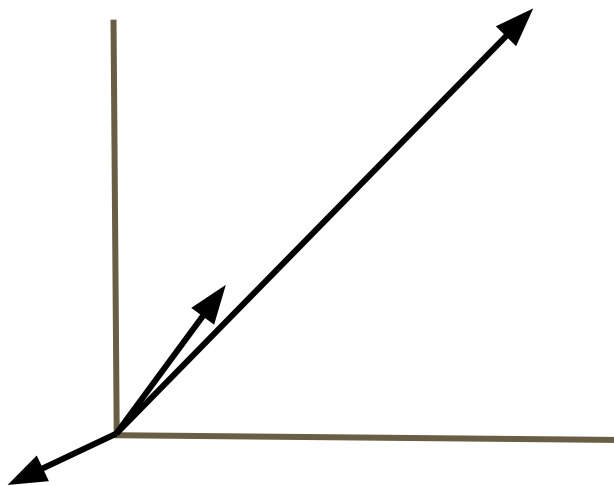
When should you use **cosine (dis)similarity** over **euclidean distance**?

When **direction** matters more than **magnitude**

Cosine Similarity

When should you use **cosine (dis)similarity** over **euclidean distance**?

When **direction** matters more than **magnitude**

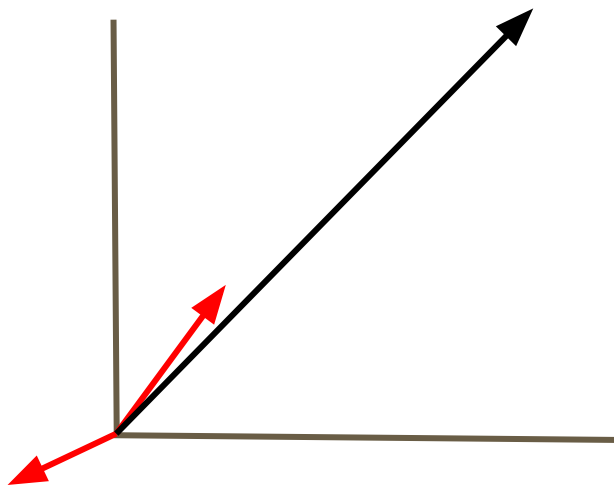


Cosine Similarity

When should you use **cosine (dis)similarity** over **euclidean distance**?

When **direction** matters more than **magnitude**

Close under
Euclidean distance

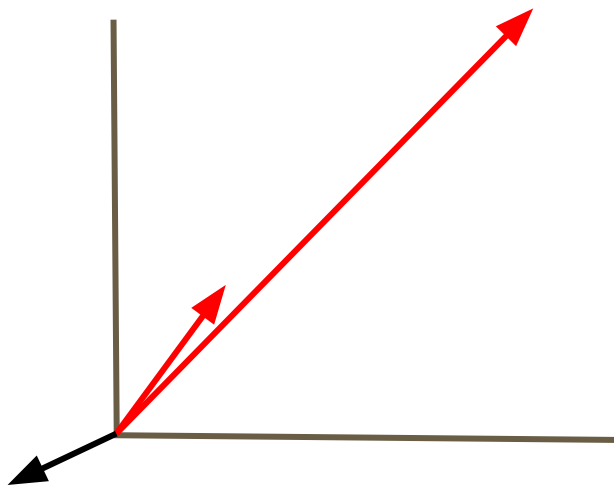


Cosine Similarity

When should you use **cosine (dis)similarity** over **euclidean distance**?

When **direction** matters more than **magnitude**

Close under Cosine
Similarity



A quick Note on Norms

$$d(A, B) = \|A - B\|$$

Size = Distance from the origin

$$d(0, X) = \|X\|$$

- Minkowski Distance \Leftrightarrow Lp Norm
- Not all distances can create a Norm