Proximity Measures: Similarity, Dissimilarity, Distance

- Similarity vs. Dissimilarity vs. Distance
- Similarity: Cosine, Jaccard, Pearson, Spearman
- Distance: Euclidean, Minkowski, Hamming, Mahalanobis
- Proximity for mixed attributes

Proximity

Similarity
- Range: [-1; 1] or [0; 1]
- Highly similar: ~1
- Binary: Jaccard
- Binary: Simple Matching
- Continuous: Cosine
- Continuous: Pearson cor.
- Ordinal: Spearman cor.

Dissimilarity
- Range: 1-similarity
- Closeness: dissim. ~ 0

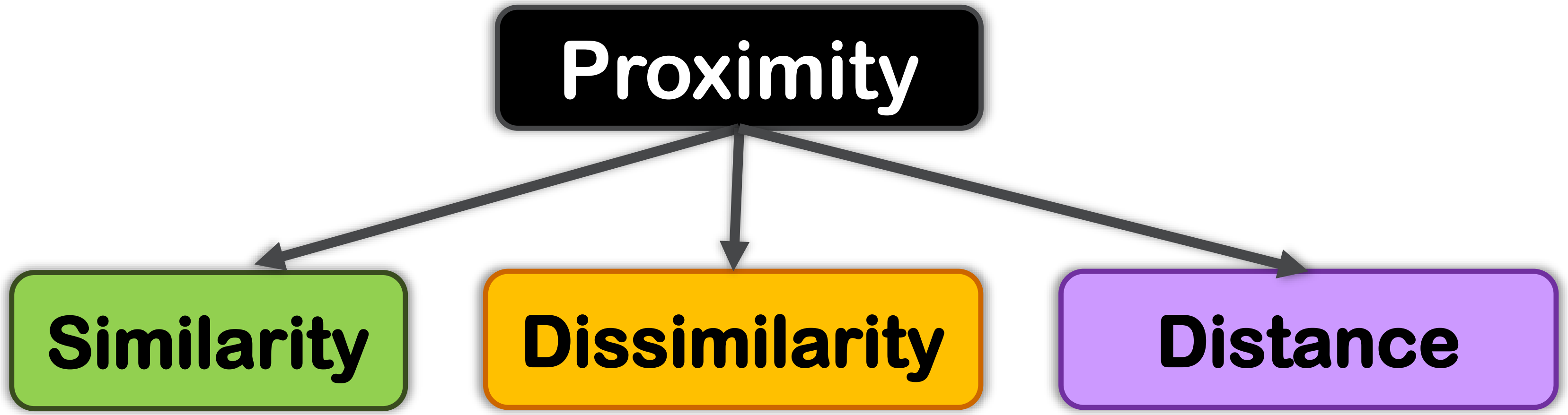
Distance
- [0; Infinity]
- Closeness: dist. ~ 0
- Attributes: Scaled
- Continuous: Euclidean
- Continuous: Minkowski
- Continuous: Mahalanobis
- Categorical: Hamming

Prof. Nagiza F. Samatova
samatova@csc.ncsu.edu
Department of Computer Science
North Carolina State University

NC STATE Executive Education

February 2019

Proximity Measures



- Range: [-1; 1] or [0; 1]
- Highly similar: ~1
- Binary: Jaccard
- Binary: Simple Matching
- Continuous: Cosine
- Continuous: Pearson cor.
- Ordinal: Spearman cor.

- Range: 1-similarity
- Closeness: dissim. ~ 0

- [0; Infinity]
- Closeness: dist. ~ 0
- Attributes: Scaled
- Continuous: Euclidean
- Continuous: Minkowski
- Continuous: Mahalanobis
- Categorical: Hamming

Proximity and Clustering

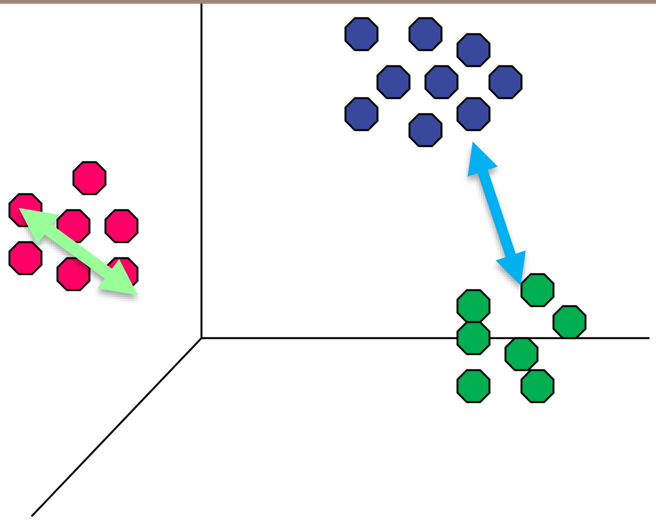
Task	Definition	Example question
Proximity **matching**	Attempts to identify *similar* individuals based on data known about them.	What are the companies that are similar to IBM's best business customers?
Clustering	Attempts to *group* individuals in a population together *by* their *proximity*, but *not driven by any specific purpose*.	Do our customers form natural groups or segments?

Proximity is at the Core of Clustering

- **Given a set of data objects, each having a set of attributes, and a proximity measure among them, find clusters such that**
  - Data points in one cluster are "more similar" to one another.
  - Data points in separate clusters are "less similar" to one another.
- **Proximity Measures:**
  - Euclidean distance if attributes are continuous.
  - Cosine similarity.

INTRA-cluster distances are minimized

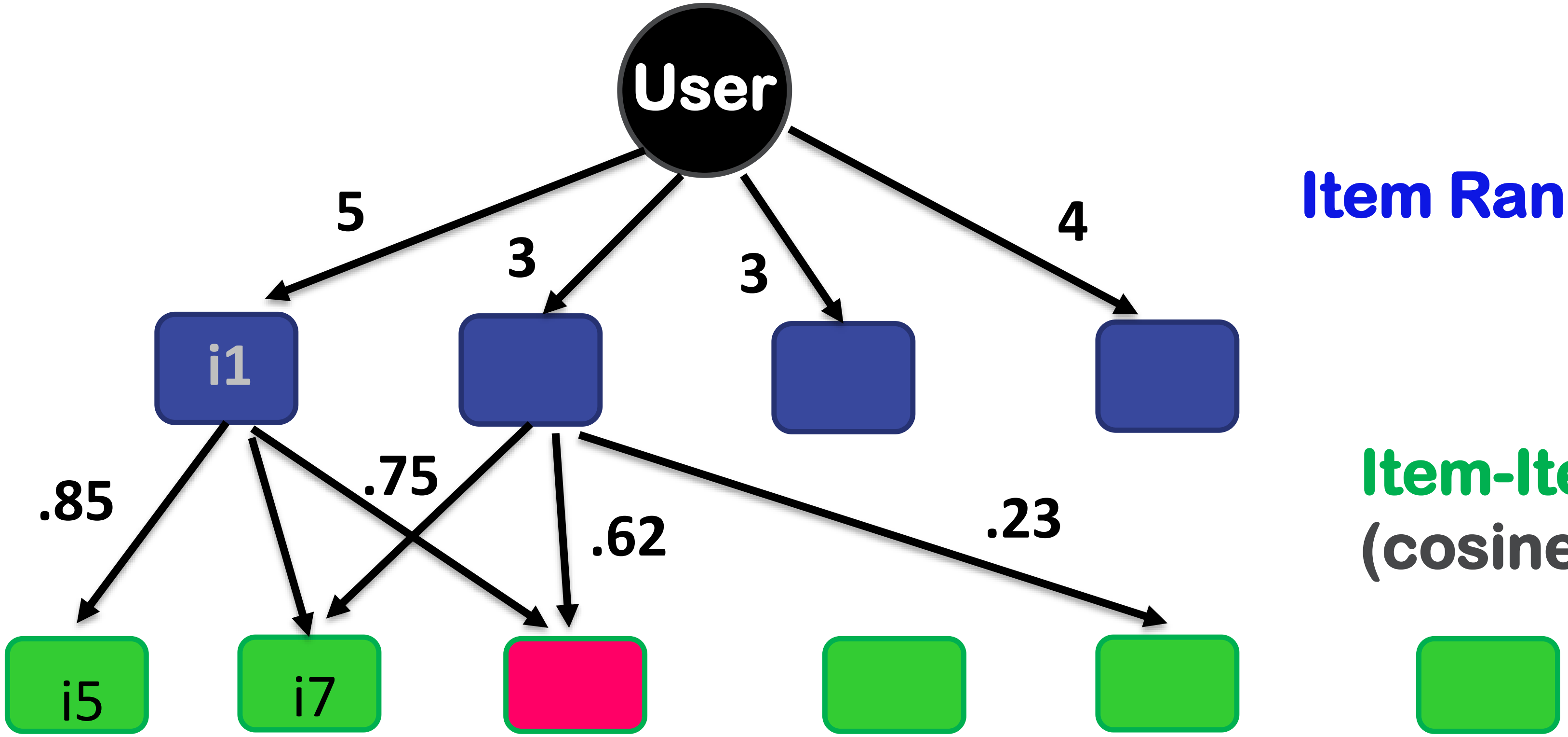
INTER-cluster distances are maximized



Euclidean Distance-based Clustering in 3-D space

Proximity is at the Core of Recommendation Systems

Item-Based Collaborative Filtering



Item Ranking

Item-Item Similarity:
(cosine, Pearson)

Item_Rank = Average Weighted Sum of Item-Item Similarities

Proximity Depends on Data Types

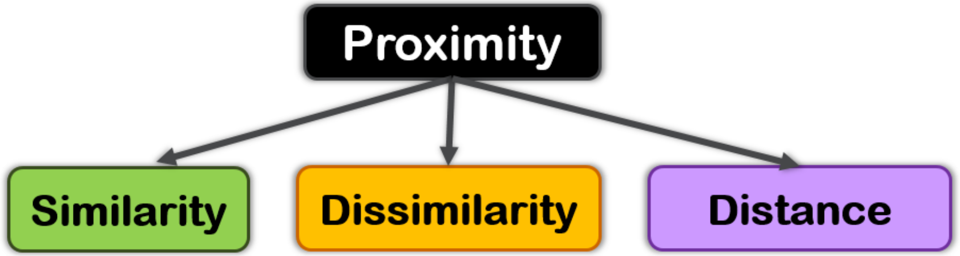
Features	Description	Example	Statistical Operation	Discrete vs. Continuous
Nominal	values are different names: provide enough info to distinguish one object from another (=, ≠)	zip codes, employee ID, eye colors, sex:{male, female}	**mode**, contingency, entropy, χ^2-test	discrete
Ordinal	values provide enough info to order objects (<, >)	grades (A, A-, B, B+) size (small, medium, large)	**median**, percentiles, rank correlation, run tests, sign tests	discrete
Interval	the differences between values are meaningful: allow ordering and subtraction but not other arithmetic operations	calendar dates, time, temperature in Celsius	**median, mean**, standard deviation, Pearson's correlation, t- and F-tests	both
Ratio	both differences and ratios are meaningful (*, /)	monetary quantities, counts, age, length, temperature	**mean, median, geometric mean, harmonic mean**, percent variation	continuous

Proximity depends on Discrete vs. Continuous Features

- **Discrete:**
  - Have only a finite or countably infinite set of values
  - Often represented as *integer* variables
  - Examples: zip codes, set of words in document collection, sex
  - Categorical:
    - Binary: two values: sex:{F, M})
    - Polytomous: a finite set of values:
      - Ordinal: can be compared (<,>): poor, good, excellent
      - Nominal: cannot be compared: zip codes, country names
    - Count: countably infinite set: number of traffic violations
- **Continuous:**
  - Have real numbers as values
  - Often represented as *floating point* variables
  - Examples: temperature, height, weight

Proximity: Similarity and Dissimilarity

- **Similarity**
  - Numerical measure of how **alike** two data objects are
  - Is higher when objects are more alike.
  - Often falls in the range [0,1]:
  - Examples: Cosine, Jaccard, Tanimoto,
- **Dissimilarity**
  - Numerical measure of how **different** two data objects are
  - Lower when objects are more alike
  - Minimum dissimilarity is often 0
  - Upper limit varies
- **Proximity refers to a similarity or dissimilarity**



Similarity/Dissimilarity for Simple Attributes

p and q are the attribute values for two data records

Features	Description	Dissimilarity	Similarity
Nominal	values are different names: provide enough info to distinguish one object from another (=, ≠)	$d = \begin{cases} 0, p = q \\ 1, p \neq q \end{cases}$	$s = \begin{cases} 1, p = q \\ 0, p \neq q \end{cases}$
Ordinal*	values provide enough info to order objects (<, >)	$d = \frac{\lvert p - q \rvert}{n - 1}$	$s = 1 - \frac{\lvert p - q \rvert}{n - 1}$
Interval or Ratio	Interval: the differences between values are meaningful: allow ordering and subtraction but not other arithmetic operations Ratio: both differences and ratios are meaningful (*, /)	$d = \lvert p - q \rvert$	$s = -d$ $s = \frac{1}{1 + d}$ $s = 1 - \frac{d - min_d}{max_d - min_d}$

* values mapped to integers $0, 1, \ldots, n-1$, where n is the number of unique values

Common Properties of a Similarity

- **Similarities, also have some well-known properties:**
  - sim(p, q) = 1 (or maximum similarity) only if p = q
  - sim(p, q) = sim(q, p)   for all p and q (Symmetry)

where sim(p, q) is the similarity between points (data objects), p and q

Similarity for Binary Attributes

- Suppose p and q have only binary attributes
- Compute similarities using the following quantities
  - M01 = the number of attributes where p was 0 and q was 1 (0-1 Mismatch)
  - M10 = the number of attributes where p was 1 and q was 0 (1-0 Mismatch)
  - M00 = the number of attributes where p was 0 and q was 0 (0-0 Match)
  - M11 = the number of attributes where p was 1 and q was 1 (1-1 Match)

- Simple Matching and Jaccard Coefficients:

  SMC = number of matches / number of attributes
  = (M11 + M00) / (M01 + M10 + M11 + M00)

  Jaccard = number of 11 matches / number of not-both-zero attributes values
  = (M11) / (M01 + M10 + M11)

Example: SMC vs. Jaccard

p = 1 0 0 0 0 0 0 0 0 0

q = 0 0 0 0 0 0 1 0 0 1

$M_{01} = 2$ (the number of attributes where p was 0 and q was 1)

$M_{10} = 1$ (the number of attributes where p was 1 and q was 0)

$M_{00} = 7$ (the number of attributes where p was 0 and q was 0)

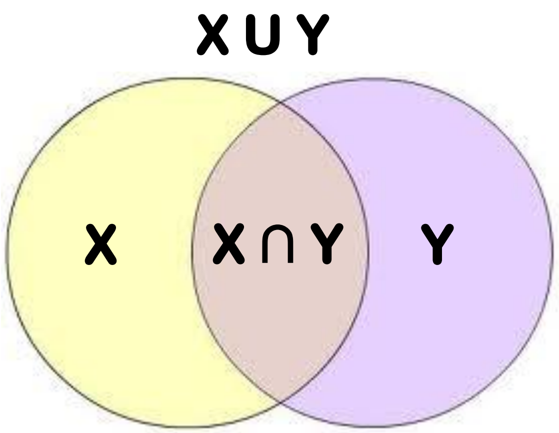
$M_{11} = 0$ (the number of attributes where p was 1 and q was 1)

SMC = $(M_{11} + M_{00})/(M_{01} + M_{10} + M_{11} + M_{00})$

$= (0+7) / (2+1+0+7) = 0.7$

Jaccard = $(M_{11}) / (M_{01} + M_{10} + M_{11}) = 0 / (2 + 1 + 0) = 0$

$$\boxed{Jaccard(X,Y) = \frac{X \cap Y}{X \cup Y}}$$



Jaccard Similarity: Binary Attributes

	U = 0	U = 1	
X = 0	a	b	a + b
X = 1	c	d	c + d
	a + c	b + d	p

Jaccard's Coefficient: d / (b + c + d)

Ignores zero matches (a):

- Desirable when we do not want two records to be similar simply because a large number of characteristics are absent in both
  - Document-document similarity: matching Words used
  - User-User similarity: matching Items purchased

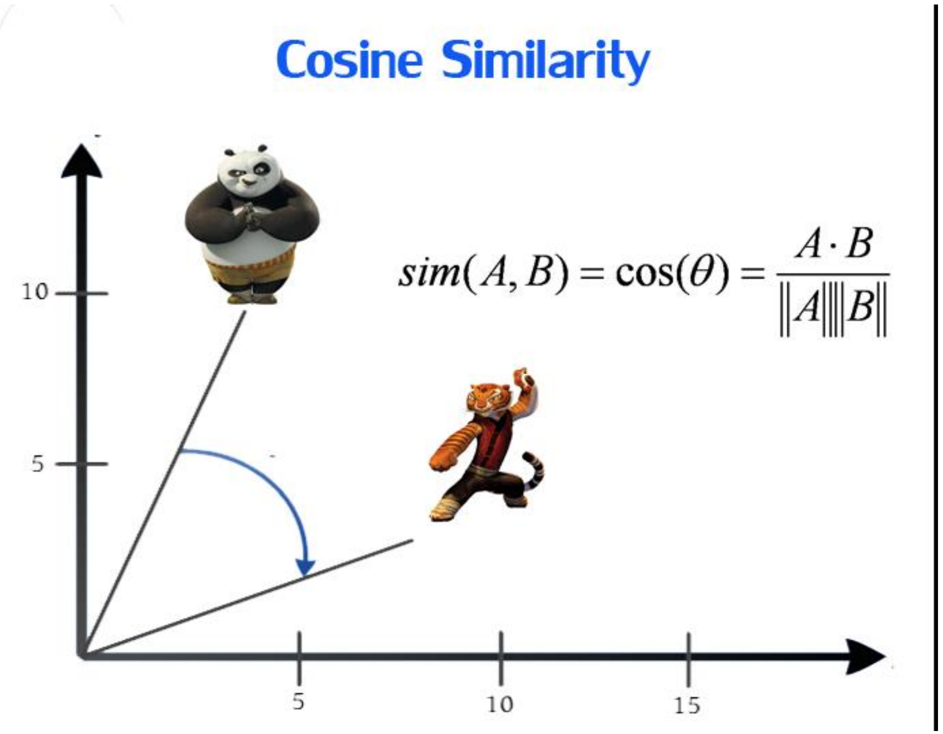
Non-binary Attributes: Cosine Similarity

- If d_1 and d_2 are two document vectors, then

  $\cos(d_1, d_2) = (d_1 \bullet d_2) / ||d_1|| \, ||d_2||$, where:

  $\bullet$ indicates vector dot product and

  $|| d ||$ is the length of vector d.



- Example:

  $d_1 =$ 3 2 0 5 0 0 0 2 0 0

  $d_2 =$ 1 0 0 0 0 0 0 1 0 2

  $\cos(d_1, d_2) = .3150$

$d_1 \bullet d_2 = 3*1 + 2*0 + 0*0 + 5*0 + 0*0 + 0*0 + 0*0 + 2*1 + 0*0 + 0*2 = 5$

$||d_1|| = (3*3+2*2+0*0+5*5+0*0+0*0+0*0+2*2+0*0+0*0)^{0.5} = (42)^{0.5} = 6.481$

$||d_2|| = (1*1+0*0+0*0+0*0+0*0+0*0+0*0+1*1+0*0+2*2)^{0.5} = (6)^{0.5} = 2.245$

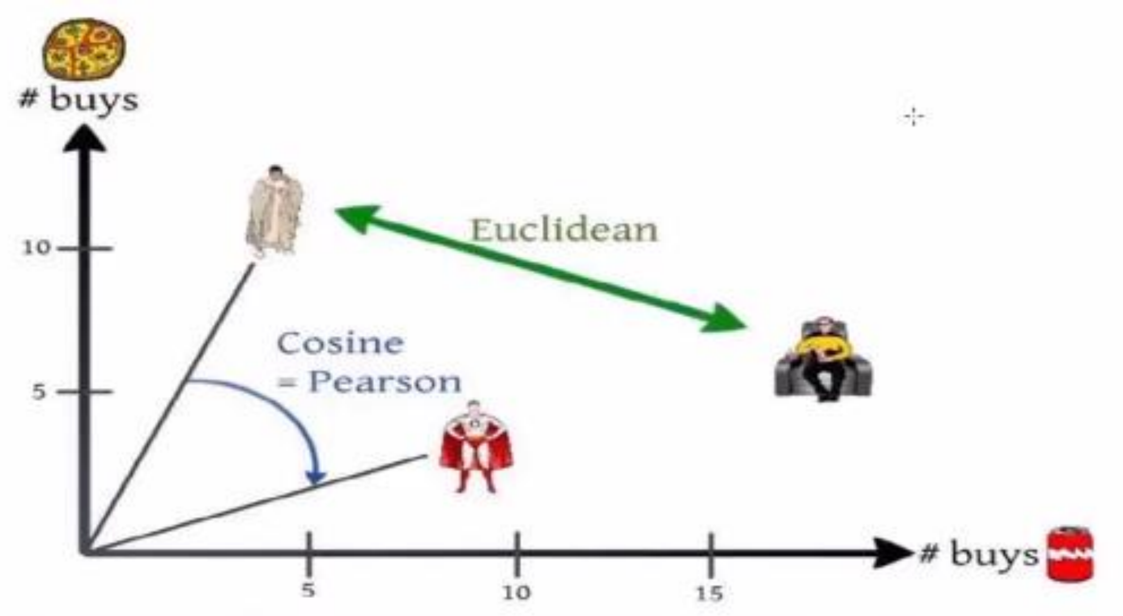
Correlation (Pearson Correlation)

- Correlation measures the *linear relationship* between objects
- To compute correlation, we standardize data objects, p and q, and then take their dot product

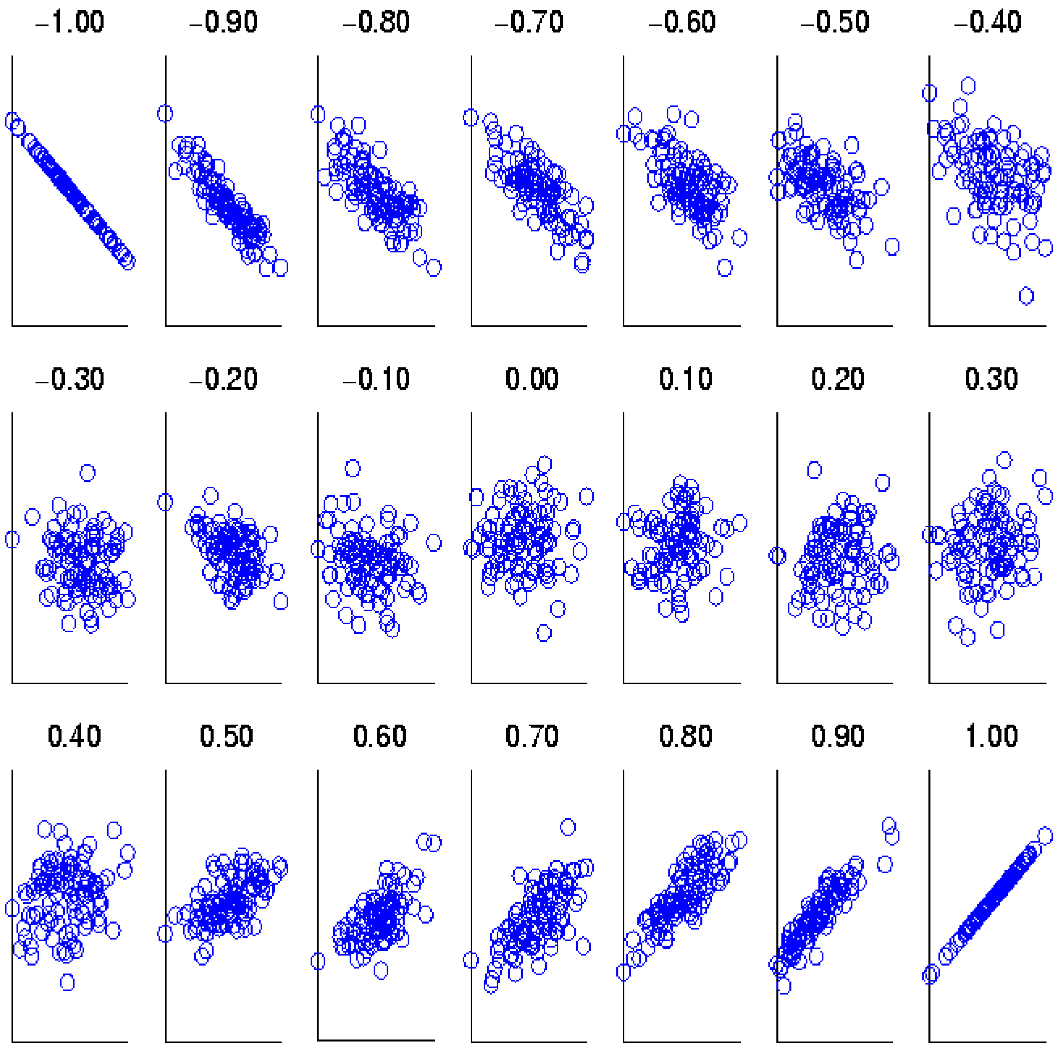
$$p'_k = (p_k - mean(p)) / std(p)$$

$$q'_k = (q_k - mean(q)) / std(q)$$

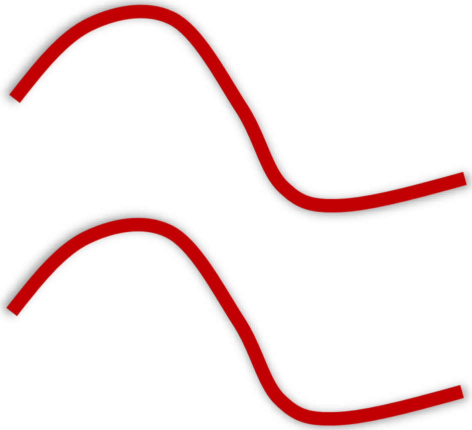
$$correlation(p, q) = p' \bullet q'$$



Visually Evaluating Correlation



Scatter plots showing the similarity from –1 to 1



Proximity Measures

Proximity

Similarity | Dissimilarity | Distance

Similarity

- Range: [-1; 1] or [0; 1]
- Highly similar: ~1
- Binary: Jaccard
- Binary: Simple Matching
- Continuous: Cosine
- Continuous: Pearson cor.
- Ordinal: Spearman cor.

Dissimilarity

- Range: 1-similarity
- Closeness: dissim. ~ 0

Distance

- [0; Infinity]
- Closeness: dist. ~ 0
- Attributes: Scaled
- Continuous: Euclidean
- Continuous: Minkowski
- Continuous: Mahalanobis
- Categorical: Hamming

Proximity: Distance Metric

- **Distance** $d(p, q)$ **between two points** p **and** q **is a dissimilarity measure if it satisfies:**

  1. **Positive definiteness**:
     $d(p, q) \geq 0$ for all p and q and
     $d(p, q) = 0$ only if $p = q$.
  2. **Symmetry**: $d(p, q) = d(q, p)$ for all p and q.
  3. **Triangle Inequality**:
     $d(p, r) \leq d(p, q) + d(q, r)$ for all points p, q, and r.

- **Examples:**
  - Euclidean distance
  - Minkowski distance
    - Manhattan (city block) distance
  - Mahalanobis distance
  - Hamming distance

Exercise: Is this a distance metric?

$$\boxed{p = (p_1, p_2, ..., p_d) \in R^d \quad \text{and} \quad q = (q_1, q_2, ..., q_d) \in R^d}$$

$$\boxed{d(p,q) = \max_{1 \le j \le d}(p_j, q_j)}$$

$$\boxed{d(p,q) = \sqrt{\sum_{j=1}^{d}(p_j - q_j)^2}}$$

$$\boxed{d(p,q) = \max_{1 \le j \le d}(p_j - q_j)}$$

$$\boxed{d(p,q) = \min_{1 \le j \le d} | p_j - q_j |}$$

Distance Metric

Not: Positive definite

Not: Symmetric

Not: Triangle Inequality

Distance: Euclidean, Minkowski, Mahalanobis

$$p = (p_1, p_2, \ldots, p_d) \in R^d \quad \text{and} \quad q = (q_1, q_2, \ldots, q_d) \in R^d$$

Euclidean

$$dist(p, q) = \sqrt{\sum_{j=1}^{d} (p_j - q_j)^2}$$

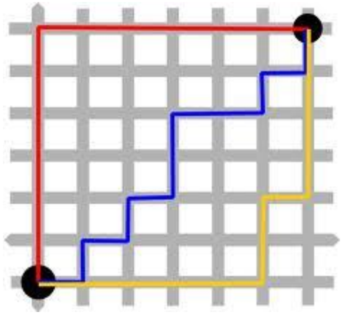
L_2-norm

Minkowski

$$dist_r(p, q) = \left(\sum_{j=1}^{d} |p_j - q_j|^r \right)^{\frac{1}{r}}$$

$$r = 1$$

City-block distance
Manhattan distance
L_1-norm



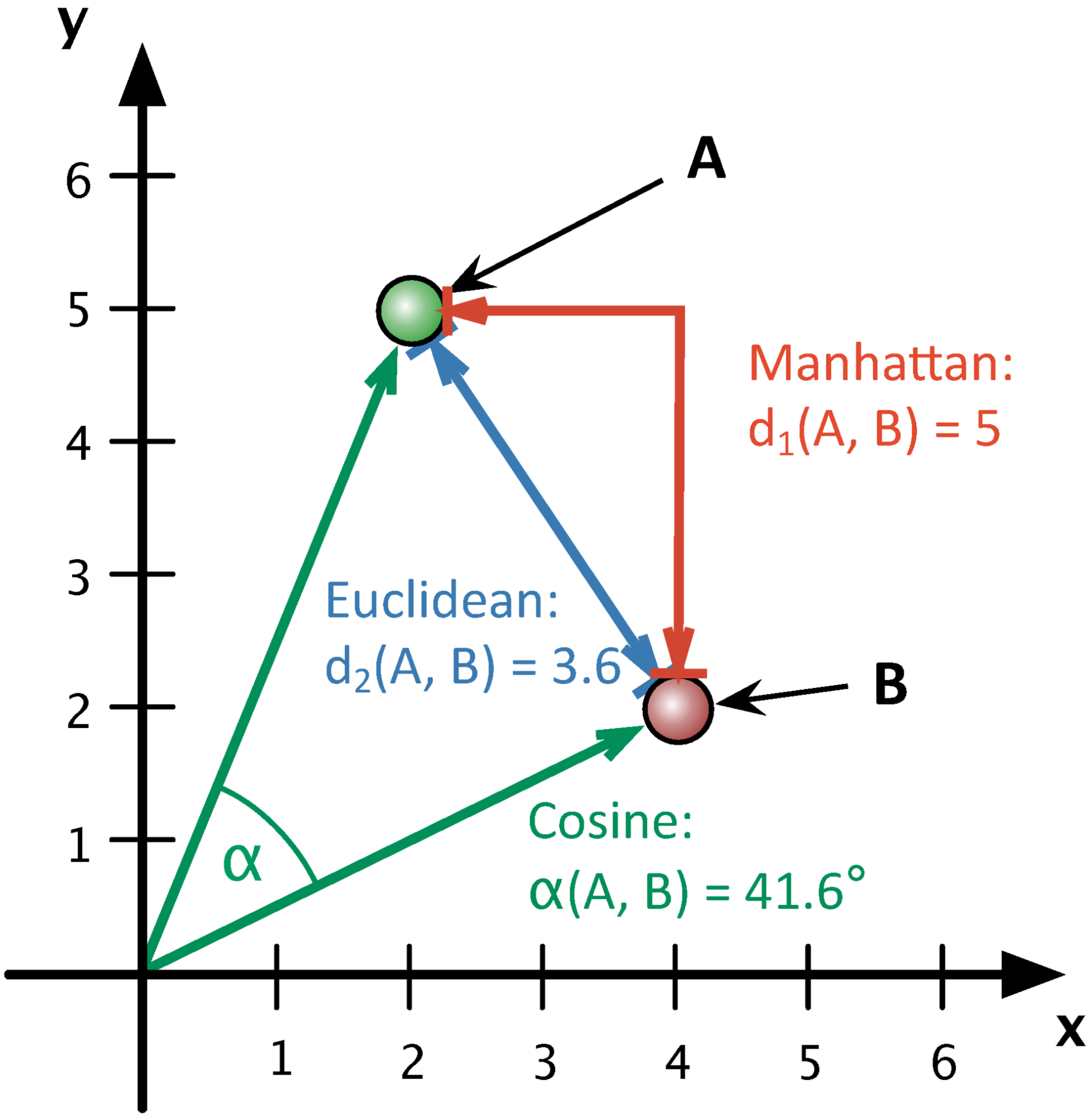
Mahalanobis

$$dist(p, q) = (p - q)\Sigma^{-1}(p - q)^{\mathrm{T}}$$

Σ^{-1}: **Empirical Covariance Matrix**

Features that are **highly correlated** with other features do not contribute as much to the distance

Examples



y
x
1
2
3
4
5
6
1
2
3
4
5
6
A
B
Manhattan:
$d_1(A, B) = 5$
Euclidean:
$d_2(A, B) = 3.6$
Cosine:
$\alpha(A, B) = 41.6°$
α

Euclidean Distance: Continuous Attributes

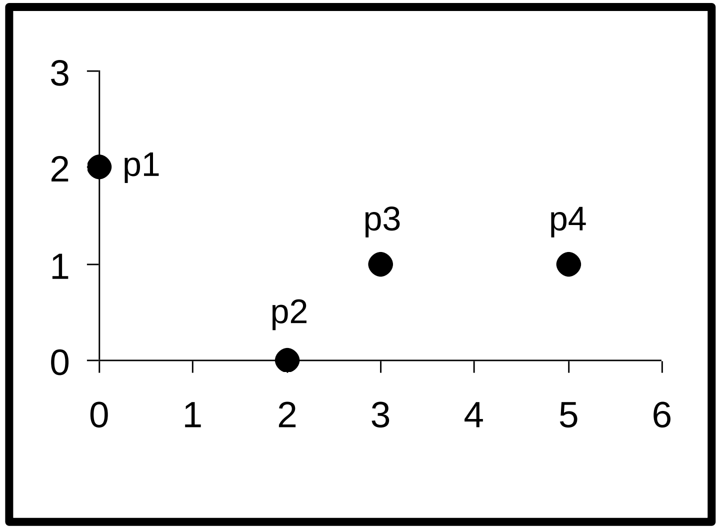
Standardization is necessary, if *scales* differ!

Ex: $p = (age, salary)$

$$dist(p,q) = \sqrt{\sum_{j=1}^{d} (p_j - q_j)^2}$$

Input Data Table: P

point	x	y
p1	0	2
p2	2	0
p3	3	1
p4	5	1



Output Distance Matrix: D

	p1	p2	p3	p4
p1	0	2.828	3.162	5.099
p2	2.828	0	1.414	3.162
p3	3.162	1.414	0	2
p4	5.099	3.162	2	0

Hamming Distance: Categorical Attributes

- **Hamming distance:**
  - The distance is 0 if the features are in the same category and 1, otherwise
  - Measures the number of bits that are different between two binary vectors

HD = number of mis-matches
= M01 + M10

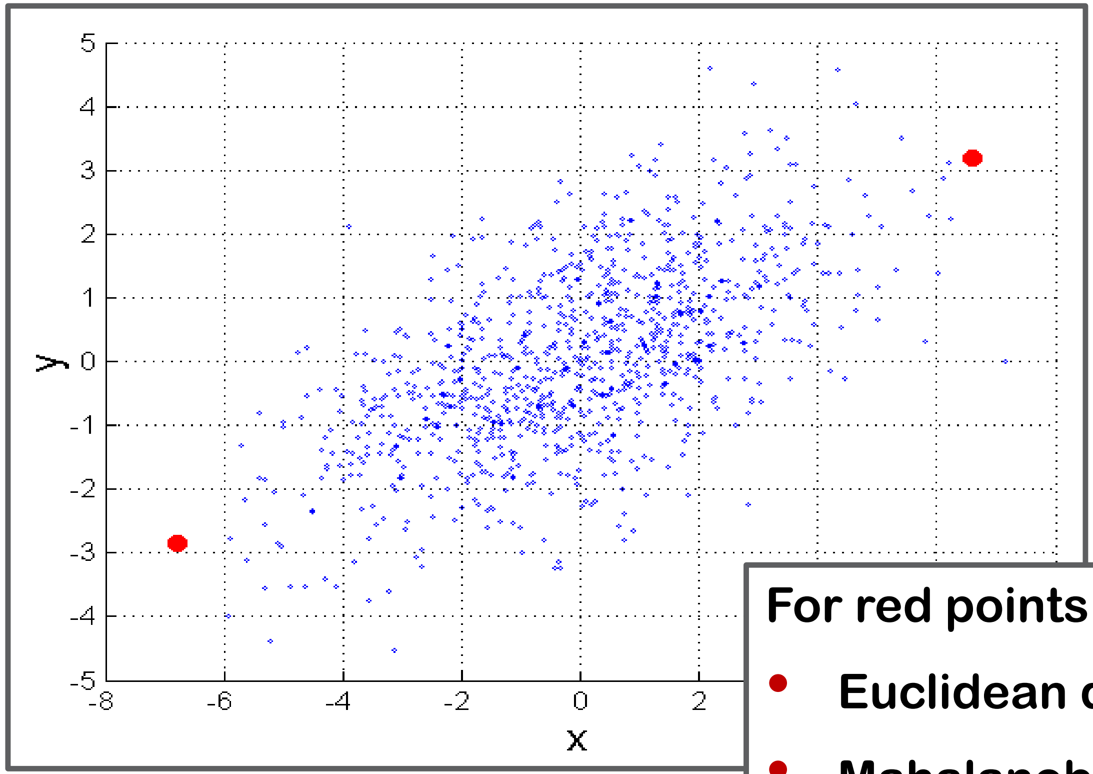
- **Matching coefficient:**
  - 1 if the features are in the same category and 0, otherwise
  - Similarity measure, inverse of Hamming distance
  - Often normalized by dividing by number of features

SMC = number of matches / number of attributes
= (M11 + M00) / (M01 + M10 + M11 + M00)

Mahalanobis Distance: Continuous Attributes

- If there is **correlation** between some attributes/dimensions/features
- Attributes have different ranges of values

$$d(p,q) = (p-q)\Sigma^{-1}(p-q)^T$$



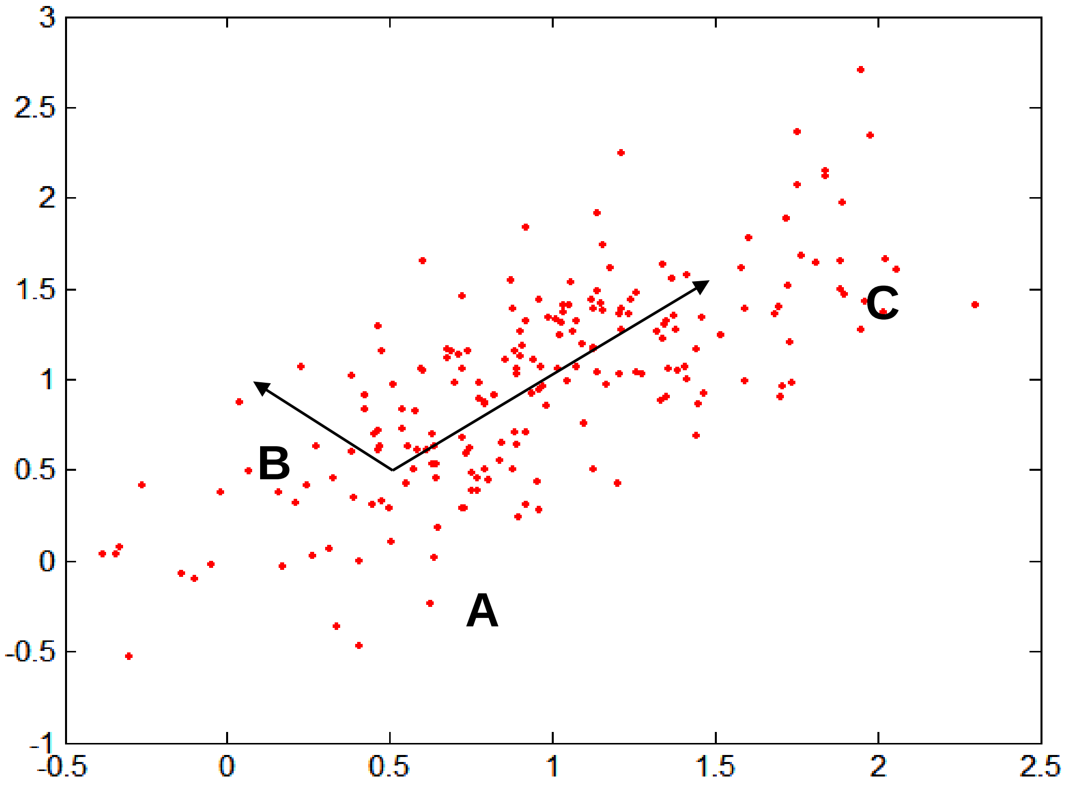
- Σ is the covariance matrix of the input data points
- Σ^{-1} is the inverse matrix

Features that are highly correlated with other features do not contribute as much to the distance

For red points:

- Euclidean distance is 14.7
- Mahalanobis distance is 6

Example: Mahalanobis Distance



Covariance Matrix:

$$\Sigma = \begin{bmatrix} 0.3 & 0.2 \\ 0.2 & 0.3 \end{bmatrix}$$

A: (0.5, 0.5)

B: (0, 1)

C: (1.5, 1.5)

<u>**Show that:**</u>

Mahal (A,B) = 5

Mahal (A,C) = 4

Proximity

MIXED ATTRIBUTES

Similarity for Mixed Attributes: Gower's Similarity

- **Gower's similarity measure:**
  - for distance, use 1 – Gower's similarity measure
- **Categorical feature or attribute:**
  - $s_i = 1$ if features are in the same category and 0, otherwise
- **Continuous feature or attribute:**
  - $$s_i = 1 - \frac{|p_i - q_i|}{range\ (i^{th}\ feature)}$$

$$\boxed{GowersSim = \frac{\sum_{i=1}^{d} s_i}{d}}$$

General Approach to Combining Similarities

- **Motivation**
  - Attributes are of many different types but an overall similarity is needed
  - Different groups of attributes require specialized similarity optimized for this group
- **Procedure to combine similarities**
  - For the k^{th} attribute (or a group of attributes), compute a similarity, s_k, in the range $[0, 1]$
  - Define an indicator variable, δ_k, for the k^{th} attribute (or groups of attributes) as follows

$$\boldsymbol{\delta_k} = \begin{cases} \mathbf{0}, & \text{if the } k^{th} \text{ attribute is a binary asymmetric attribute and both objects have a value of 0, or if one of the objects has missing values for the } k^{th} \text{ attribute} \\ \mathbf{1}, & \text{otherwise} \end{cases}$$

  - Compute the overall similarity between the two objects using the formula:

$$\boxed{\boldsymbol{sim(p,q) = \frac{\sum_{k=1}^{d} \delta_k s_k}{\sum_{k=1}^{d} \delta_k}}}$$

Using Weights to Combine Similarities

- **May not want to treat all attributes the same:**
  - Use weights w_k that are between 0 and 1 and sum to 1

Similarity:

$$sim(p,q) = \frac{\sum_{k=1}^{d} w_k \delta_k s_k}{\sum_{k=1}^{d} \delta_k}$$

Distance:

$$dist_r(p,q) = \left(\sum_{j=1}^{d} w_j \left| p_j - q_j \right|^r \right)^{\frac{1}{r}}$$

References

- https://docs.scipy.org/doc/scipy-0.7.x/reference/spatial.distance.html
- http://scikit-learn.org/stable/modules/classes.html#module-sklearn.metrics.pairwise