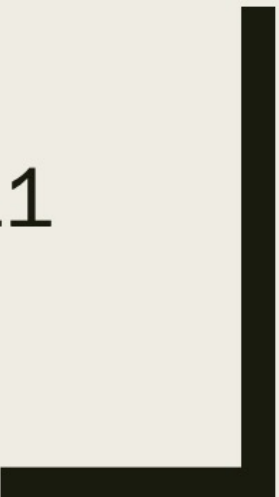




# LECTURE 11

UNIT II  
18CS54 DATA MINING



# Outline

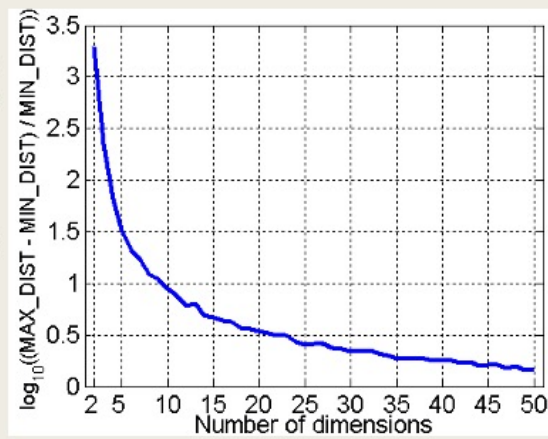
- Data Preprocessing
  - *Dimensionality Reduction*
  - *Feature subset selection*
  - *Feature creation*

# Data Preprocessing

- Discretization and Binarization
- Aggregation
- Sampling
- Attribute Transformation
- Dimensionality Reduction
- Feature subset selection
- Feature Weighing
- Feature creation

# Curse of Dimensionality

- When dimensionality increases, data becomes increasingly sparse in the space that it occupies
- Definitions of density and distance between points, which are critical for clustering and outlier detection, become less meaningful



- Randomly generate 500 points
- Compute difference between max and min distance between any pair of points

# Dimensionality Reduction

- Purpose:

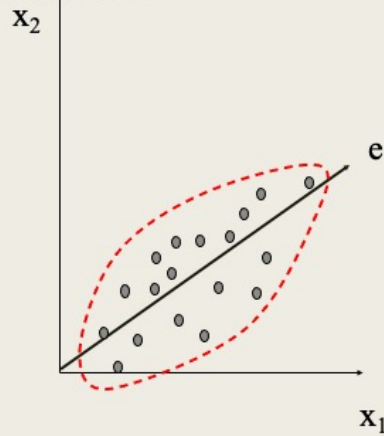
- *Avoid curse of dimensionality*
- *Reduce amount of time and memory required by data mining algorithms*
- *Allow data to be more easily visualized*
- *May help to eliminate irrelevant features or reduce noise*

- Techniques

- *Principal Components Analysis (PCA)*
- *Singular Value Decomposition*
- *Others: supervised and non-linear techniques*

# Dimensionality Reduction: PCA

- Goal is to find a projection that captures the largest amount of variation in data



# Dimensionality Reduction:

PCA

256



# Feature Subset Selection...

- Another way to reduce dimensionality of data
- Redundant features
  - *Duplicate much or all the information contained in one or more other attributes*
  - *Example: purchase price of a product and the amount of sales tax paid*
- Irrelevant features
  - *Contain no information that is useful for the data mining task at hand*
  - *Example: students' ID is often irrelevant to the task of predicting students' GPA*
- Many techniques developed, especially for classification



## Feature Subset Selection...

- While some irrelevant and redundant attributes can be eliminated immediately by using common sense or domain knowledge, **selecting the best subset of features frequently requires a systematic approach.**
- The **ideal approach** to feature selection is to try all possible subsets of features as input to the data mining algorithm of interest, and then take the subset that produces the best results. This method has the advantage of reflecting the objective and bias of the data mining algorithm that will eventually be used.
- But the **number of subsets involving  $n$  attributes is  $2^n$ , such an approach is impractical** in most situations and alternative strategies are needed.

# Feature Subset Selection...

- There are **three standard approaches** to feature selection: **embedded, filter, and wrapper**.
- **Embedded approaches** Feature selection occurs naturally as part of the data mining algorithm. Specifically, during the operation of the data mining algorithm, the algorithm itself decides which attributes to use and which to ignore.

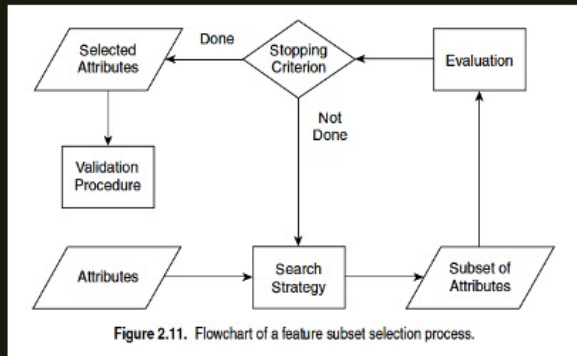
## Feature Subset Selection...

- **Filter approaches** Features are selected before the data mining algorithm is run, using some approach that is independent of the data mining task.
- For example, select sets of attributes whose pairwise correlation is as low as possible.

## Feature Subset Selection...

- **Wrapper approaches** These methods use the target data mining algorithm as a black box to find the best subset of attributes, in a way similar to that of the **ideal algorithm**, but typically without enumerating all possible subsets.

# FEATURE SUBSET SELECTION



# Feature Weighting

- Feature weighting is an alternative to keeping or eliminating features.
- More important features are assigned a higher weight, while less important features are given a lower weight.
- These weights are sometimes assigned based on domain knowledge about the relative importance of features. Alternatively, they may be determined automatically.
- For example, some classification schemes, such as **support vector machines produce classification models in which each feature is given a weight**. Features with larger weights play a more important role in the model.
- The normalization of objects that takes place when computing the **cosine similarity** can also be regarded as a type of feature weighting.

# Feature Creation

- Create new attributes that can capture the important information in a data set much more efficiently than the original attributes
- Three general methodologies:
  - *Feature extraction*
    - Example: extracting edges from images
  - *Feature construction*
    - Example: dividing mass by volume to get density
  - *Mapping data to new space*
    - Example: Fourier and wavelet analysis

Feature extraction : Eg: Presence or absence of human face

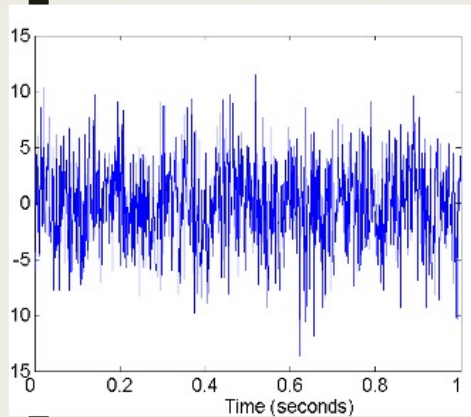
Feature construction: consider a data set consisting of information about historical artifacts, which, along with other information, contains the volume and mass of each artifact. For simplicity, assume that these artifacts are made of a small number of materials (wood, clay, bronze, gold) and that we want to classify the artifacts with respect to the material of which they are made. In this case, a density feature constructed from the mass and volume features,

i.e.,  $\text{density} = \text{mass}/\text{volume}$ , would most directly

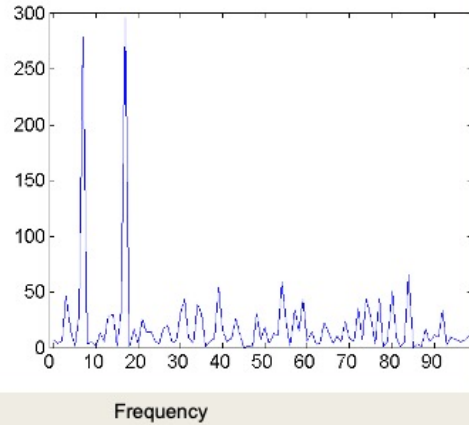
yield an accurate classification. Although there have been some attempts to automatically perform feature construction by exploring simple mathematical combinations of existing attributes, the most common approach is to construct features using domain expertise.

# Mapping Data to a New Space

## Fourier and wavelet transform



**Two Sine Waves + Noise**



**Frequency**

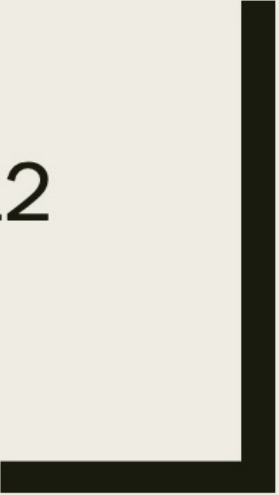
In spite of the noise, there are two peaks that correspond to the periods of the two original, non-noisy time series. Again, the main point is that better features can reveal important aspects of the data.





# LECTURE 12

18CS54 DATA MINING  
UNIT II



# Outline

- Similarity and Dissimilarity Measures
  - *Similarity and Dissimilarity for simple attributes*
  - *Euclidean Distance*
  - *Minkowski Distance*
  - *SMC*
  - *Jaccard*
  - *Cosine Similarity*
  - *Correlation*
- MSE 1
  - *GitHub link to checkout the lecture slides*  
[\(<https://github.com/vanivasudevan/Data-Mining>\)](https://github.com/vanivasudevan/Data-Mining)
  - *Question Paper Format*
  - *Topics to focus from Unit I & Unit II*

Measures such as correlation and Euclidean distance are useful for dense data such as time series or two-dimensional points, as well as the Jaccard and cosine similarity measures are useful for sparse data like documents.

# Similarity and Dissimilarity Measures

- Similarity measure
  - *Numerical measure of how alike two data objects are.*
  - *Is higher when objects are more alike.*
  - *Often falls in the range [0,1]*
- Dissimilarity measure
  - *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

Measures such as correlation and Euclidean distance are useful for dense data such as time series or two-dimensional points, as well as the Jaccard and cosine similarity measures are useful for sparse data like documents.

## Similarity/Dissimilarity for Simple Attributes

The following table shows the similarity and dissimilarity between two objects,  $x$  and  $y$ , with respect to a single, simple attribute.

Attribute Type	Dissimilarity	Similarity
Nominal	$d = \begin{cases} 0 & \text{if } x = y \\ 1 & \text{if } x \neq y \end{cases}$	$s = \begin{cases} 1 & \text{if } x = y \\ 0 & \text{if } x \neq y \end{cases}$
Ordinal	$d =  x - y  / (n - 1)$ (values mapped to integers 0 to $n - 1$ , where $n$ is the number of values)	$s = 1 - d$
Interval or Ratio	$d =  x - y $	$s = -d, s = \frac{1}{1+d}, s = e^{-d},$ $s = 1 - \frac{d - \min_d}{\max_d - \min_d}$

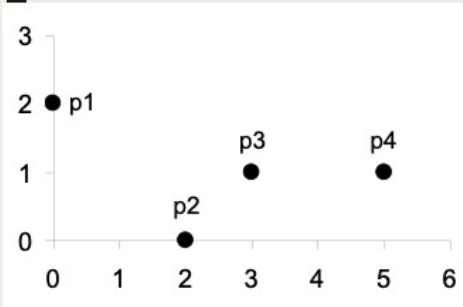
## Euclidean Distance

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{k=1}^n (x_k - y_k)^2}$$

where  $n$  is the number of dimensions (attributes) and  $x_k$  and  $y_k$  are, respectively, the  $k^{th}$  attributes (components) or data objects  $\mathbf{x}$  and  $\mathbf{y}$ .

- Standardization is necessary, if scales differ.

## Euclidean Distance



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

	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

**Distance Matrix**

$$(0-2)^2 + (2-0)^2 = 2.828$$

## Minkowski Distance

- Minkowski Distance is a generalization of Euclidean Distance

$$d(\mathbf{x}, \mathbf{y}) = \left( \sum_{k=1}^n |x_k - y_k|^r \right)^{1/r}$$

*Where  $r$  is a parameter,  $n$  is the number of dimensions (attributes) and  $x_k$  and  $y_k$  are, respectively, the  $k^{th}$  attributes (components) or data objects  $\mathbf{x}$  and  $\mathbf{y}$ .*

## Minkowski Distance: Examples

- $r = 1$ . City block (Manhattan, taxicab,  $L_1$  norm) distance.
  - *A common example of this for binary vectors is the Hamming distance, which is just the number of bits that are different between two binary vectors*
- $r = 2$ . Euclidean distance
- $r \rightarrow \infty$ . “supremum” ( $L_{\max}$  norm,  $L_{\infty}$  norm) distance.
  - *This is the maximum difference between any component of the vectors*
- Do not confuse  $r$  with  $n$ , i.e., all these distances are defined for all numbers of dimensions.



# Minkowski Distance

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

L1	p1	p2	p3	p4
p1	0	4	4	6
p2	4	0	2	4
p3	4	2	0	2
p4	6	4	2	0

L2	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

$L_{\infty}$	p1	p2	p3	p4
p1	0	2	3	5
p2	2	0	1	3
p3	3	1	0	2
p4	5	3	2	0

**Distance Matrix**

## Common Properties of a Distance

- Distances, such as the Euclidean distance, have some well-known properties.

1.  $d(\mathbf{x}, \mathbf{y}) \geq 0$  for all  $\mathbf{x}$  and  $\mathbf{y}$  and  $d(\mathbf{x}, \mathbf{y}) = 0$  if and only if  $\mathbf{x} = \mathbf{y}$ .
2.  $d(\mathbf{x}, \mathbf{y}) = d(\mathbf{y}, \mathbf{x})$  for all  $\mathbf{x}$  and  $\mathbf{y}$ . (Symmetry)
3.  $d(\mathbf{x}, \mathbf{z}) \leq d(\mathbf{x}, \mathbf{y}) + d(\mathbf{y}, \mathbf{z})$  for all points  $\mathbf{x}, \mathbf{y}$ , and  $\mathbf{z}$ . (Triangle Inequality)

where  $d(\mathbf{x}, \mathbf{y})$  is the distance (dissimilarity) between points (data objects),  $\mathbf{x}$  and  $\mathbf{y}$ .

- A distance that satisfies these properties is a **metric**

## Common Properties of a Similarity

- Similarities, also have some well-known properties.

1.  $s(\mathbf{x}, \mathbf{y}) = 1$  (or maximum similarity) only if  $\mathbf{x} = \mathbf{y}$ .  
(does not always hold, e.g., cosine)
2.  $s(\mathbf{x}, \mathbf{y}) = s(\mathbf{y}, \mathbf{x})$  for all  $\mathbf{x}$  and  $\mathbf{y}$ . (Symmetry)

where  $s(\mathbf{x}, \mathbf{y})$  is the similarity between points (data objects),  $\mathbf{x}$  and  $\mathbf{y}$ .

## Similarity Between Binary Vectors

- Common situation is that objects,  $\mathbf{x}$  and  $\mathbf{y}$ , have only binary attributes

- Compute similarities using the following quantities

$f_{01}$  = the number of attributes where  $\mathbf{x}$  was 0 and  $\mathbf{y}$  was 1

$f_{10}$  = the number of attributes where  $\mathbf{x}$  was 1 and  $\mathbf{y}$  was 0

$f_{00}$  = the number of attributes where  $\mathbf{x}$  was 0 and  $\mathbf{y}$  was 0

$f_{11}$  = the number of attributes where  $\mathbf{x}$  was 1 and  $\mathbf{y}$  was 1

- Simple Matching and Jaccard Coefficients

SMC = number of matches / number of attributes

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

J = number of 11 matches / number of non-zero attributes

$$= (f_{11}) / (f_{01} + f_{10} + f_{11})$$

## SMC versus Jaccard: Example

**x** = 1 0 0 0 0 0 0 0 0 0

**y** = 0 0 0 0 0 0 1 0 0 1

$f_{01} = 2$  (the number of attributes where **x** was 0 and **y** was 1)

$f_{10} = 1$  (the number of attributes where **x** was 1 and **y** was 0)

$f_{00} = 7$  (the number of attributes where **x** was 0 and **y** was 0)

$f_{11} = 0$  (the number of attributes where **x** was 1 and **y** was 1)

$$\begin{aligned}\text{SMC} &= (f_{11} + f_{00}) / (f_{01} + f_{10} + f_{11} + f_{00}) \\ &= (0 + 7) / (2 + 1 + 0 + 7) = 0.7\end{aligned}$$

$$J = (f_{11}) / (f_{01} + f_{10} + f_{11}) = 0 / (2 + 1 + 0) = 0$$

## Cosine Similarity

- If  $\mathbf{d}_1$  and  $\mathbf{d}_2$  are two document vectors, then

$$\cos(\mathbf{d}_1, \mathbf{d}_2) = \langle \mathbf{d}_1, \mathbf{d}_2 \rangle / \|\mathbf{d}_1\| \|\mathbf{d}_2\| ,$$

where  $\langle \mathbf{d}_1, \mathbf{d}_2 \rangle$  indicates inner product or vector dot product of vectors,  $\mathbf{d}_1$  and  $\mathbf{d}_2$ , and  $\|\mathbf{d}\|$  is the length of vector  $\mathbf{d}$ .

- Example:

$$\mathbf{d}_1 = 3 \ 2 \ 0 \ 5 \ 0 \ 0 \ 0 \ 2 \ 0 \ 0$$

$$\mathbf{d}_2 = 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 2$$

$$\langle \mathbf{d}_1, \mathbf{d}_2 \rangle = 3*1 + 2*0 + 0*0 + 5*0 + 0*0 + 0*0 + 0*0 + 2*1 + 0*0 + 0*2 = 5$$

$$\|\mathbf{d}_1\| = (3^2 + 2^2 + 0^2 + 5^2 + 0^2 + 0^2 + 0^2 + 2^2 + 0^2 + 0^2)^{0.5} = (42)^{0.5} = 6.481$$

$$\|\mathbf{d}_2\| = (1^2 + 0^2 + 0^2 + 0^2 + 0^2 + 0^2 + 0^2 + 1^2 + 0^2 + 2^2)^{0.5} = (6)^{0.5} = 2.449$$

$$\cos(\mathbf{d}_1, \mathbf{d}_2) = 0.3150$$

$X=(1,1,1,0,1,0)$  and  $Y=(1,0,1,0,1,0)$

J, CS, Euclidean

$$ED = 1$$

$f_{01} = 0$  (the number of attributes where  $x$  was 0 and  $y$  was 1)

$f_{10} = 1$  (the number of attributes where  $x$  was 1 and  $y$  was 0)

$f_{00} = 2$  (the number of attributes where  $x$  was 0 and  $y$  was 0)

$f_{11} = 3$  (the number of attributes where  $x$  was 1 and  $y$  was 1)

$$J = (f_{11}) / (f_{01} + f_{10} + f_{11}) = 3/0+1+3 = 3/4 = 0.75$$

Jaccard Similarity = number of 1-1 matches / ( number of bits - number 0-0 matches) =  $3/6-2 = 0.75$

$X=(1,1,1,0,1,0)$  and  $Y=(1,0,1,0,1,0)$

$$\cos(d1,d2) = \langle d1,d2 \rangle / \|\mathbf{d}_1\| \|\mathbf{d}_2\| = 1+1+1 = 3 / 2*1.73 = 0.87$$

## Correlation measures the linear relationship between objects

$$\text{corr}(\mathbf{x}, \mathbf{y}) = \frac{\text{covariance}(\mathbf{x}, \mathbf{y})}{\text{standard\_deviation}(\mathbf{x}) * \text{standard\_deviation}(\mathbf{y})} = \frac{s_{xy}}{s_x s_y}, \quad (2.11)$$

where we are using the following standard statistical notation and definitions

$$\text{covariance}(\mathbf{x}, \mathbf{y}) = s_{xy} = \frac{1}{n-1} \sum_{k=1}^n (x_k - \bar{x})(y_k - \bar{y}) \quad (2.12)$$

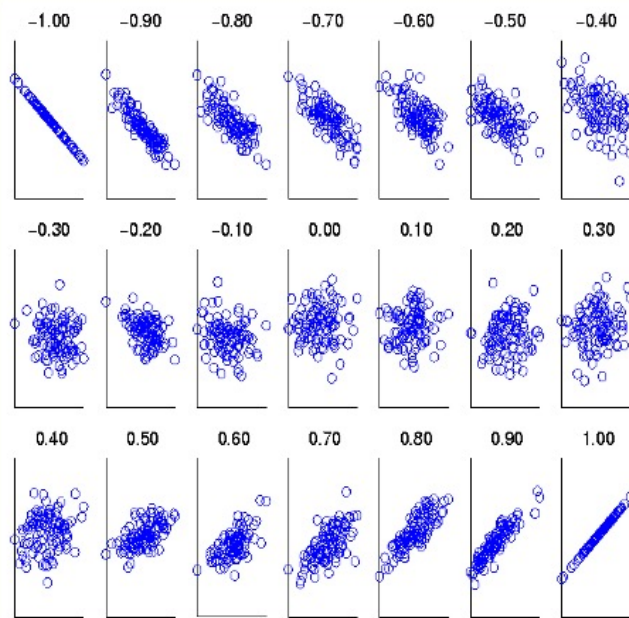
$$\text{standard\_deviation}(\mathbf{x}) = s_x = \sqrt{\frac{1}{n-1} \sum_{k=1}^n (x_k - \bar{x})^2}$$

$$\text{standard\_deviation}(\mathbf{y}) = s_y = \sqrt{\frac{1}{n-1} \sum_{k=1}^n (y_k - \bar{y})^2}$$

$$\bar{x} = \frac{1}{n} \sum_{k=1}^n x_k \text{ is the mean of } \mathbf{x}$$

$$\bar{y} = \frac{1}{n} \sum_{k=1}^n y_k \text{ is the mean of } \mathbf{y}$$

## Visually Evaluating Correlation



**Scatter plots  
showing the  
similarity from  
-1 to 1.**



## Drawback of Correlation

- $x = (-3, -2, -1, 0, 1, 2, 3)$

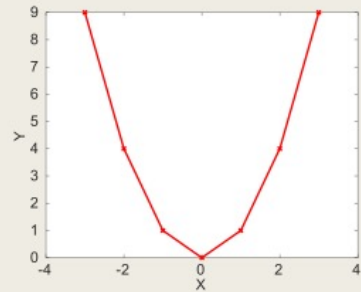
- $y = (9, 4, 1, 0, 1, 4, 9)$

$$y_i = x_i^2$$

- $\text{mean}(x) = 0, \text{mean}(y) = 4$

- $\text{std}(x) = 2.16, \text{std}(y) = 3.74$

- $$\text{corr} = \frac{(-3)(5) + (-2)(0) + (-1)(-3) + (0)(-4) + (1)(-3) + (2)(0) + 3(5)}{6 * 2.16 * 3.74} = 0$$



non-linear relationships may still exist

## Correlation vs Cosine vs Euclidean Distance

- Compare the three proximity measures according to their behavior under variable transformation

- *scaling: multiplication by a value*
- *translation: adding a constant*

Property	Cosine	Correlation	Euclidean Distance
Invariant to scaling (multiplication)	Yes	Yes	No
Invariant to translation (addition)	No	Yes	No

- Consider the example

- $x = (1, 2, 4, 3, 0, 0, 0)$ ,  $y = (1, 2, 3, 4, 0, 0, 0)$
- $y_s = y * 2$  (scaled version of  $y$ ),  $y_t = y + 5$  (translated version)

Measure	$(x, y)$	$(x, y_s)$	$(x, y_t)$
Cosine	0.9667	0.9667	0.7940
Correlation	0.9429	0.9429	0.9429
Euclidean Distance	1.4142	5.8310	14.2127

## Correlation vs cosine vs Euclidean distance

- Choice of the right proximity measure depends on the domain
- What is the correct choice of proximity measure for the following situations?
  - *Comparing documents using the frequencies of words*
    - Documents are considered similar if the word frequencies are similar
  - *Comparing the temperature in Celsius of two locations*
    - Two locations are considered similar if the temperatures are similar in magnitude
  - *Comparing two time series of temperature measured in Celsius*
    - Two time series are considered similar if their “shape” is similar, i.e., they vary in the same way over time, achieving minimums and maximums at similar times, etc.

## Comparison of Proximity Measures

- Domain of application
  - *Similarity measures tend to be specific to the type of attribute and data*
  - *Record data, images, graphs, sequences, 3D-protein structure, etc. tend to have different measures*
- However, one can talk about various properties that you would like a proximity measure to have
  - *Symmetry is a common one*
  - *Tolerance to noise and outliers is another*
  - *Ability to find more types of patterns?*
  - *Many others possible*
- The measure must be applicable to the data and produce results that agree with domain knowledge