Measuring distance and similarity of data objects
Many different data

- documents (webpages, books)
- records of users
- graphs
- images
- videos
- Strings (DNA sequences)
- Timeseries

- How do we compare them?
Data Representation

dataset $\mathbf{X}$ as a collection of objects

write $x, y, z, \ldots$ for objects in $\mathbf{X}$

at this point no assumption about the representation of objects in $\mathbf{X}$

$x$ can be

- real-valued vectors
- binary vectors
- sets
- time series
- images
Distance function

want to define function

\[ d : X \times X \rightarrow \mathbb{R} \]

what properties should \( d \) have?
Distance functions

\[ d(x, y) \geq 0 \]  
non negativity

\[ d(x, y) = 0 \text{ iff } x = y \]  
isolation

\[ d(x, y) = d(y, x) \]  
symmetry

\[ d(x, y) \leq d(x, z) + d(z, y) \]  
triangle inequality
Metric distance functions and metric spaces

A distance function that satisfies all properties of non-negativity, isolation, symmetry, and triangle inequality is called a metric.

A data space equipped with a metric function is called a metric space.
Distance and similarity functions

distance function $d : X \times X \rightarrow \mathbb{R}$

large for dissimilar objects

similarity function $s : X \times X \rightarrow \mathbb{R}$

large for similar objects

often similarity $s$ is between 0 and 1

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

$s(x, y) \propto e^{-d(x,y)}$
Distance functions for real-valued vectors

- $L_p$ norms or Minkowski distance:

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

- $p = 1$, $L_1$, Manhattan (or city block) or Hamming distance:

$$L_1(x, y) = \left( \sum_{i=1}^{d} |x_i - y_i| \right)$$
Distance functions for real-valued vectors

- **$L_p$ norms or Minkowski distance:**

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

- **$p = 2$, $L_2$, Euclidean distance:**

\[
L_2(x, y) = \left( \sum_{i=1}^{d} (x_i - y_i)^2 \right)^{1/2}
\]
Data structures

**data matrix**

\[
\begin{pmatrix}
x_{11} & \cdots & x_{1m} \\
\vdots & & \vdots \\
x_{n1} & \cdots & x_{nm}
\end{pmatrix}
\]

**distance matrix**

\[
\begin{pmatrix}
0 & \cdots \\
d(2, 1) & 0 & \cdots \\
\vdots & & \\
d(n, 1) & d(n, 2) & \cdots & d(n, n-1) & 0
\end{pmatrix}
\]
Similarity functions for real-valued vectors

• Dot product or cosine similarity

\[ \cos(x, y) = \frac{x \cdot y}{\|x\| \|y\|} \]

• Can we construct a distance function out of this?

• When use the one and when the other?
Distance functions for 0/1 data

\[
L_1(x, y) = \left( \sum_{i=1}^{d} |x_i - y_i| \right)
\]
How good is Hamming distance for 0–1 vectors?

• Drawback

• Documents represented as sets (of words)
• Two cases

  – Two very large documents -- almost identical -- but for 5 terms
  – Two very small documents, with 5 terms each, disjoint
Distance functions for binary vectors or sets

- **Jaccard** similarity between binary vectors $x$ and $y$ (Range?)
  
  $$J\text{Sim}(x, y) = \frac{|x \cap y|}{|x \cup y|}$$

- **Jaccard** distance (Range?):
  
  $$J\text{Dist}(x, y) = 1 - \frac{|x \cap y|}{|x \cup y|}$$
The previous example

• Case 1 (very large almost identical documents)

  \[ J(x, y) \text{ almost } 1 \]

  \( x \)

  \( y \)

• Case 2 (small disjoint documents)

  \[ J(x, y) = 0 \]

  \( x \)

  \( y \)
Distance functions between strings

strings \( x \) and \( y \) of equal length

modification of the Hamming distance

add 1 for all positions that are different

\[
\begin{align*}
  x & : c\,g\,t\,a\,a\,c\,g \\
  y & : g\,a\,t\,t\,t\,a\,c\,a
\end{align*}
\]

string Hamming distance = 4

drawbacks?
Distance functions between strings

1. strings must have equal length

2. what about

\[
\begin{align*}
\text{x} & \quad a \quad g \quad a \quad t \quad t \quad t \\
\text{y} & \quad g \quad a \quad t \quad t \quad a \quad c \quad a
\end{align*}
\]

string Hamming distance = 6
String edit distance

consider two strings \( x \) and \( y \)

try to change one to another

only single-character edits are allowed

insert character

delete character

substitute character

edit distance is the minimum number of such operations

not necessary to have equal length!
String edit distance

example

\[ \text{x: a g a t t a c} \]
\[ \text{y: g a t t a c a} \]

string edit distance = 2
String edit distance

consider two strings $x$ and $y$ of lengths $n$ and $m$, respectively

how can I compute the string edit distance between $x$ and $y$?

how expensive is this computation?
Computing the edit distance

- Dynamic programming
- Form nxm distance matrix $D$ (x of length n, y of length m)

$$D$$

$x$

$y$

- $D(i,j)$ is the optimal distance between strings $x[1..i]$ and $y[1..j]$
Computing the edit distance

• How to compute $D(i,j)$?
• Either
  – match the last two characters (substitution)
  – match by deleting the last char in one string
  – match by deleting the last character in the other string
Computing edit distance

\[ D(i, j) = \min \{ D(i - 1, j) + \text{del}(X[i]), \\
D(i, j - 1) + \text{ins}(Y[j]), \\
D(i - 1, j - 1) + \text{sub}(X[i], Y[j]) \} \]

- Running time? Metric?
Distance function between time series

• time series can be seen as vectors
• apply existing distance metrics
• L–norms

• what can go wrong?
Distance functions between time series

• Euclidean distance between time series

figures from Eamonn Keogh [www.cs.ucr.edu/~eamonn/DTW_myths.ppt]
Dynamic time warping

- Alleviate the problems with Euclidean distance
Dynamic time warping

- Quite useful in practice

figures from Eamonn Keogh [www.cs.ucr.edu/~eamonn/DTW_myths.ppt]
Dynamic time warping

• how to compute it?

• Dynamic programming

figures from Eamonn Keogh [www.cs.ucr.edu/~eamonn/DTW_myths.ppt]
Dynamic time warping

• constraints for more efficient computation

figures from Eamonn Keogh www.cs.ucr.edu/~eamonn/DTW_myths.ppt