5-Distances-Timeseries

September 18, 2016

1 Distances and Timeseries

Today we will start building some tools for making comparisons of data objects, with particular attention to timeseries.

Working with data, we can encounter a wide variety of different data objects:

- Records of users
- Graphs
- Images
- Videos
- Text (webpages, books)
- Strings (DNA sequences)
- Timeseries
- …

How can we compare them?

1.1 Feature space representation

Usually a data object consists of a set of attributes.

These are also commonly called **features.**

- ("J. Smith", 25, $ 200,000)

If all d dimensions are real-valued then we can visualize each data object as a point in a vector space.

- (25, USD 200,000) $\rightarrow \begin{bmatrix} 25 \\ 200000 \end{bmatrix}$.

Likewise If all features are binary then we can think of each data object as a binary vector in vector space.

The space is called **feature space.**

We then are naturally interested in how **similar** or **dissimilar** two objects are.

A dissimilarity function takes two objects as input, and returns a large value when then two objects are not very similar.

Often we put restrictions on the dissimilarity function.

One of the most common is that it be a **metric.**

The dissimilarity $d(x, y)$ between two objects x and y is a **metric** if

- $d(i, j) = 0 \leftrightarrow i == j$. (identity of indiscernables)
- $d(i, j) = d(j, i)$ (symmetry)
- $d(i, j) \leq d(i, h) + d(h, j)$ (triangle inequality)

A metric is also commonly called a **distance**.

Sometimes we will use "distance" informally, ie, to refer to a dissimilarity function even if we are not sure it is a metric. We'll try to say "dissimilarity" in those cases though.

Definitions of distance or dissimilarity functions are usually diferent for real, boolean, categorical, and ordinal variables.

Weights may be associated with diferent variables based on applications and data semantics.

1.2 Matrix representation

Very often we will manage data conveniently in matrix form.

The standard way of doing this is:

$$
m \text{ data objects} \left\{
\begin{array}{c}
\overbrace{\begin{bmatrix}
x_{11} & \cdots & x_{1j} & \cdots & x_{1n} \\
\vdots & \ddots & \vdots & \ddots & \vdots \\
x_{i1} & \cdots & x_{ij} & \cdots & x_{in} \\
\vdots & \ddots & \vdots & \ddots & \vdots \\
x_{m1} & \cdots & x_{mj} & \cdots & x_{mn}
\end{bmatrix}}^{n \text{ features}}
\end{array}
\right.
$$

Where we typically use symbols m for number of rows (objects) and n for number of columns (features).

When we are working with distances, the matrix representation is:

$$
m \text{ data objects} \left\{
\overbrace{\begin{bmatrix}
0 & & & & \\
d(1,2) & 0 & & & \\
d(1,3) & d(2,3) & \ddots & & \\
\vdots & \vdots & \vdots & & \\
d(1,m) & d(2,m) & \cdots & 0 &
\end{bmatrix}}^{m \text{ data objects}}
\right.
$$

1.3 Norms

Assume some function $p(\mathbf{v})$ which measures the "size" of the vector $\mathbf{v}$.

$p()$ is called a **norm** if:

- $p(a\mathbf{v}) = |a|\, p(\mathbf{v})$ (absolute scaling)
- $p(\mathbf{u} + \mathbf{v}) \leq p(\mathbf{u}) + p(\mathbf{v})$ (subadditivity)
- $p(\mathbf{v}) = 0 \leftrightarrow \mathbf{v}$ is the zero vector (separates points)

Norms are important for this reason:
Every norm defines a corresponding metric.
That is, if $p()$ is a norm, then $d(\mathbf{x}, \mathbf{y}) = p(\mathbf{x} - \mathbf{y})$ is a metric.
ℓ_p Norm.

Defines the *Minkowski distance*.

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

A special case of this isthe ℓ_2 norm that we've already studied, which defines the **Euclidean** norm.

It is equal to the euclidean distance between the vectors.

$$L_1(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^{d} (x_i - y_i)^2}$$

Another important special case is the ℓ_1 norm.

This defines the **Manhattan** distance, or (for binary vectors), the **Hamming** distance:

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