MADR-2025-Project-Tasks-03

Multidimensional Scaling - distances between some European cities

The **eurodist** dataset in R gives the road distances (in km) between 21 cities in Europe. The data are taken from a table in The Cambridge Encyclopaedia.

- Implement the MDS algorithm yourself. Perform MDS on this data, and find a two-dimensional set of points which has interpoint distances approximately equal to the data.
- Plot the coordinates you've obtained along with the labels giving the city names. Is the plot similar to the map of Europe?
- Use the MDS implementation from the scikit-learn library (or any other trustworthy library or package) for the **eurodist** data - compare the results with the ones you've got using your implementation.
- Is the distance matrix from **eurodist** Euclidean?
- Now work in the opposite direction - create the Euclidean distance matrix from the set of two-dimensional points you've found using MDS. Compute the Frobienius norm of the difference of this matrix amd the original distance matrix.

A classical example - perception of colors in human vision

(From the book: A. Izenman (2008), *Modern Multivariate Statistical Techniques*, Springer Texts in Statistics, 2013) In an experiment designed to study the perceptions of color in human vision, 14 colors differing only in their hue (i.e., wavelengths from 434 μm to 674 μm) were projected two at a time onto a screen in an all-pairs design to 31 subjects. The colors correspond to the following wavelengths: 434=*indigo*, 445=*blue*, 472=*blue-green*, 504=*green*, 555=*yellow-green*, 600=*yellow*, 628=*orange-yellow*, 651=*orange*, 674=*red*.

The subjects rated each of the possible $\binom{14}{2} = 91$ pairs on a five-point scale from 0 ("no similarity at all") to 4 ("identical"). The rating for each pair of colors was averaged over all

subjects and the result divided by 4 to bring the similarity ratings into the interval $[0, 1]$. These mean similarity ratings were then collected into a (14×14) table (see file **color.stimuli.rda**).

- Use your implementation of MDS to reproduce the so called two-dimensional *color wheel* (find also information on *Newton disc* in Wikipedia) - remember to subtract the numbers from **color.stimuli.rda** from 1 (as the MDS is built on dissimilarities, not similarities).
- Would a one-dimensional solution work well for this problem?

Multidimensional scaling with non-Euclidean matrices

- Generate a number of random points on the plane. Compute the distance matrix for the points. Use your implementation of MDS and plot the coordinates you've obtained. Compare the plot with the plot of the original data.
- Modify the distance matrix you've got in the previous point to make it a non-Euclidean distance matrix. Use your implementation of MDS to find a set of points that have distances approximately given by the modified distance matrix. Plot the coordinates you've obtained. Try different modifications of the Euclidean distance matrix from the first point to see their effect on the final result.

Multidimensional scaling for binary data

- Given some numbers $p_{ij} \in [0, 1]$, $i, j = 1, 2, 3$, generate the synthethic data of 30 binary attributes $a_1 \ldots, a_{30}$ on 300 cases in the following way:
  - the values of the attributes $a_1, \ldots, a_{10}$ for the first one hundred cases should be generated independently from the Bernoulli distribution with the parameter p_{11}, for the next one hundred cases with the parameter p_{21}, and with the last one hundred cases with the parameter p_{31},
  - the values of the attributes $a_{11}, \ldots, a_{20}$ for the first one hundred cases should be generated independently from the Bernoulli distribution with the parameter p_{12}, for the next one hundred cases with the parameter p_{22}, and with the last one hundred cases with the parameter p_{32},
  - the values of the attributes $a_{21}, \ldots, a_{30}$ for the first one hundred cases should be generated independently from the Bernoulli distribution with the parameter p_{13}, for the next one hundred cases with the parameter p_{23}, and with the last one hundred cases with the parameter p_{33}.

(You should thus get a table with 300 rows and 30 columns, containing zeroes and ones.)

- Compute the Jaccard index and SMC similarity matrices for these data.

- Perform classical MDS for both similarity matrices, producing a plot of the coordinates in 2D. Color the points coming from the first 100 observations red, the next 100 green, and the last 100 blue. Are the results for both similarity matrices similar?
- Try different values of the parameters p_{ij} to see the effect that this has.