D5: Databases and Information Systems
Data Mining and Matrices, SS 2015
Programming assignment #3: ICA and spectral clustering
Due: **26 July 2015** at 23:59 CEST

max planck institut
informatik

You must hand in machine-typed report in PDF format and a script file (e.g. `*.R` *file). The report must explain your approach to the problem, the results you obtained, and your interpretation of the results. Naturally, the report must also answer to any direct question presented in the problems. You can, and in many cases should, add plots and other illustrations to your answers. The script file must show every step you have taken to solve these tasks, that is to say, if we run the script file we must get the same results you reported and see the same figures you presented. You can discuss these problems with other students, and you are encouraged to discuss with the tutor, but everybody must hand in their own answers and own code. Return your answers by email to* `dmm15@mpi-inf.mpg.de`*. Remember to write your name and matriculation number to every answer sheet!*

Task 1: ICA for housing prices

Download the data and utility files from `http://resources.mpi-inf.mpg.de/departments/d5/teaching/ss15/dmm/assignments/assignment_3.zip`. That package contains file `assignment3.R`. You can fill your answers to that file and return it as a part of your solution.

In this task, we study the applications of independent component analysis to housing price data from the US. The data set `us_housing_prices.csv`[1] contains the monthly house price index for twenty metropolitan areas in the US from January 1987 to June 2014. As is common to time series data, the rows correspond to the locations and the columns to the time stamps.

Get yourself familiar with the data by studying which locations it covers and by plotting the 20 time series. This data contains missing values (denoted `NA` in R), and we have to impute some values before we can proceed with the analysis. For this first round, we simply replace every missing value with 0; we will get back to this later.

Our algorithms expect the rows of the data to be the observations while the columns are the variables, and hence we will transpose the data. The index values vary between different locations, so before we proceed, we will normalize the data to z-scores. Is this normalization sensible? Argue!

We start by computing the full ICA of the scaled data (i.e. we find 20 independent components) using R's `fastICA` package. Compute the ICA and explain what do the different matrices in the solution mean. Show how to reconstruct the housing price index of Los Angeles, CA, from the ICA.

Let us now study the independent components. Plot them as time series. Can you interpret them? Remember that we do not know the sign of the components, so you might want multiply some components with -1 for plotting to have the peaks and well go the right way. For this analysis, it's useful to compare the locations of the peaks and wells with the original time series and/or to your general knowledge of US economy in the studied era.

We can also plot the scatter plot of the first 2 independent components. Can you see any outliers? If yes, identify the dates the outliers correspond to. Can you interpret them (probably using some general knowledge of US economy or by looking at the original data)? Is there any reason we should look for the top-2 independent components instead of, say, 7th and 8th? Try scatter plots of different pairs of independent components. Can you see other outliers and can you interpret them?

The US housing bubble made the housing prices climb heavily from around year 2000 until 2006, when they started to decline. By late 2008, the decline had turned into a crisis with significant drops in the housing prices; the houses would not start to fully recover until 2012. Can you identify these events in the independent components, either as such or from the scatter plots?

We now revise some of our earlier decisions, starting with the imputation of missing values. Earlier we replaced all missing values with 0s. Do you think this was a sensible decision given the data? One alternative would be to replace the missing values with the average of the data. Do you think that would be a sensible decision? Argue! The last option we are going to consider is to notice that all missing values happen at the begin of the time series, so we can replace the missing values with the first observed value for that location. Would this be sensible? Argue! Choose from the aforementioned techniques the one you think is the best and use it to replace the missing values in the original data. Transpose and scale the data again.

[1] Data source `http://data.okfn.org/data/core/house-prices-us`.

We now study the effects of whitening. Compute the SVD of the new transposed and scaled data. The columns of $\boldsymbol{U}$ now correspond to the uncorrelated (whitened) time series. Plot (some of) them. Do they show any patterns? Can you interpret them? Study the singular values. Are all the 20 uncorrelated signals necessary? Use some method to select the rank (e.g. Guttman–Kaiser or scree test) and report the results. Which method and rank you choose? Why?

Re-compute ICA, but compute only k independent components (using as k the rank you chose above). Study the results. Did they change? Can you interpret them? Report you findings and compare them with what you found in the previous step.

Task 2: Spectral clustering

In this experiment, we try to cluster the digit data into 10 different clusters; the optimal clustering assigns the same digits to the same cluster, and different digits to different clusters. With this data, we know the "correct" clustering of the data; this is often not the case in practice.

We use Euclidean distance as the distance measure and Gaussian kernel to diminish dissimilar points. That is, we replace every pairwise distance $d_{ij} = \|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2$ with

$$\tilde{d}_{ij} = \exp\left\{\frac{-d_{ij}^2}{2\sigma^2}\right\} ,$$

where σ is a parameter for the kernel. The functions we provide take care of this, but you must provide the parameter σ.

Compute the full similarity graph using $\sigma = 50$. Study the distribution of the resulting similarities (e.g., using a histogram as described in the provided R script). Is $\sigma = 50$ a good choice? Try to find a good setting for σ by trying both smaller and larger values. Discuss!

Now use a large value of $\sigma = 50$ and find (roughly) the smallest ϵ such that the ϵ-neighborhood graph is connected. Note that you can use the magnitudes of the smallest eigenvalues of the Laplacian to judge whether or not the graph is connected (as before, R may return a very small value for 0 eigenvalues). Now find the smallest k such that the symmetric kNN graph is connected, and the smallest k such that the mutual kNN graph is connected. Plot the resulting similarity matrices. Are they different? If so, why? Discuss!

Try to find a parameter setting for spectral clustering that you expect to work well (without actually performing the clustering). Then cluster the data into 10 clusters using unnormalized spectral clustering with your parameter settings. Compute the normalized mutual information (NMI) between the ground truth clustering and the clustering you obtained. Compare your results with what you would get by applying k-means to the raw data.

Task 3: Better clustering

Compute the k-means clustering of the digits data after you have applied the Karhunen–Lóeve transformation (i.e. PCA) for top-10 principal components. Are the results any better than before? Now, use different parameters (and similarity graphs and Laplacians) to improve your results with spectral clustering. Implement your approach inside the provided boilerplate code and test it using the provided test function. Try to get your NMI close to 0.25.

Optional competition. Try to get as good clustering of the digits data as possible. You can use any of the matrix factorization methods we have seen so far for pre-processing (SVD/PCA, NMF, CX/CUR, ICA), and either k-means, hierarchical clustering, or spectral clustering. Implement your approach in the provided boilerplate code and test it using the provided test script. If you pass this assignment and are among the students with the lowest (best) score, you will automatically receive an **excellent** grade. Notice that in order to qualify, your algorithms cannot assume anything on the ground-truth clustering. Also, we must be able to repeat your results by executing the test function. *This part of this task is optional. You can pass this assignment even if you do not take part to this competition. You must, however, do the first part of the task in any case.*