Machine Learning Exercise Sheet 11

Dimensionality Reduction & Matrix Factorization

In-class Exercise

There is no in-class exercise this week.

Homework

t-SNE

Problem 1: Figure 1 shows a scatter plot of your two-dimensional data ($N = 13$ instances). You want to apply a non-linear dimensionality reduction technique based on neighbor graphs (e.g. T-SNE or UMAP). As a first step you compute the $N \times N$ weighted adjacency matrix representing the neighbor graph. Assume that the weights are computed as

$$p_{j|i} = \frac{\exp\left(-\left\|\mathbf{x}_i - \mathbf{x}_j\right\|^2 / 2\sigma^2\right)}{\sum_{k \neq i} \exp\left(-\left\|\mathbf{x}_i - \mathbf{x}_k\right\|^2 / 2\sigma^2\right)}$$

where $\boldsymbol{x}_i \in \mathbb{R}^2$ and you set $p_{i|i} = 0$. Finally, you obtain the similarity between instances i and j with $p_{ij} = \frac{p_{i|j} + p_{j|i}}{2}$.

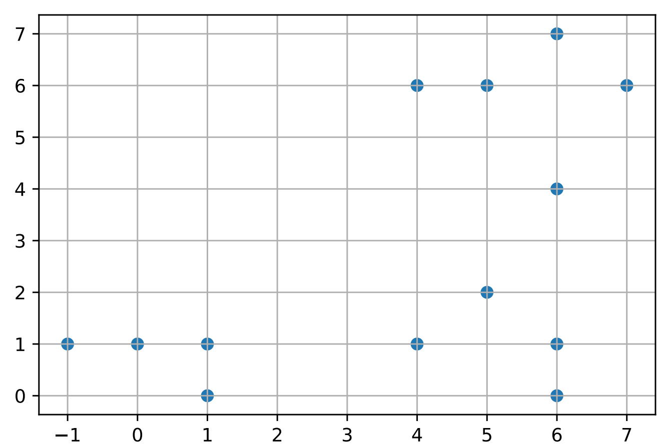
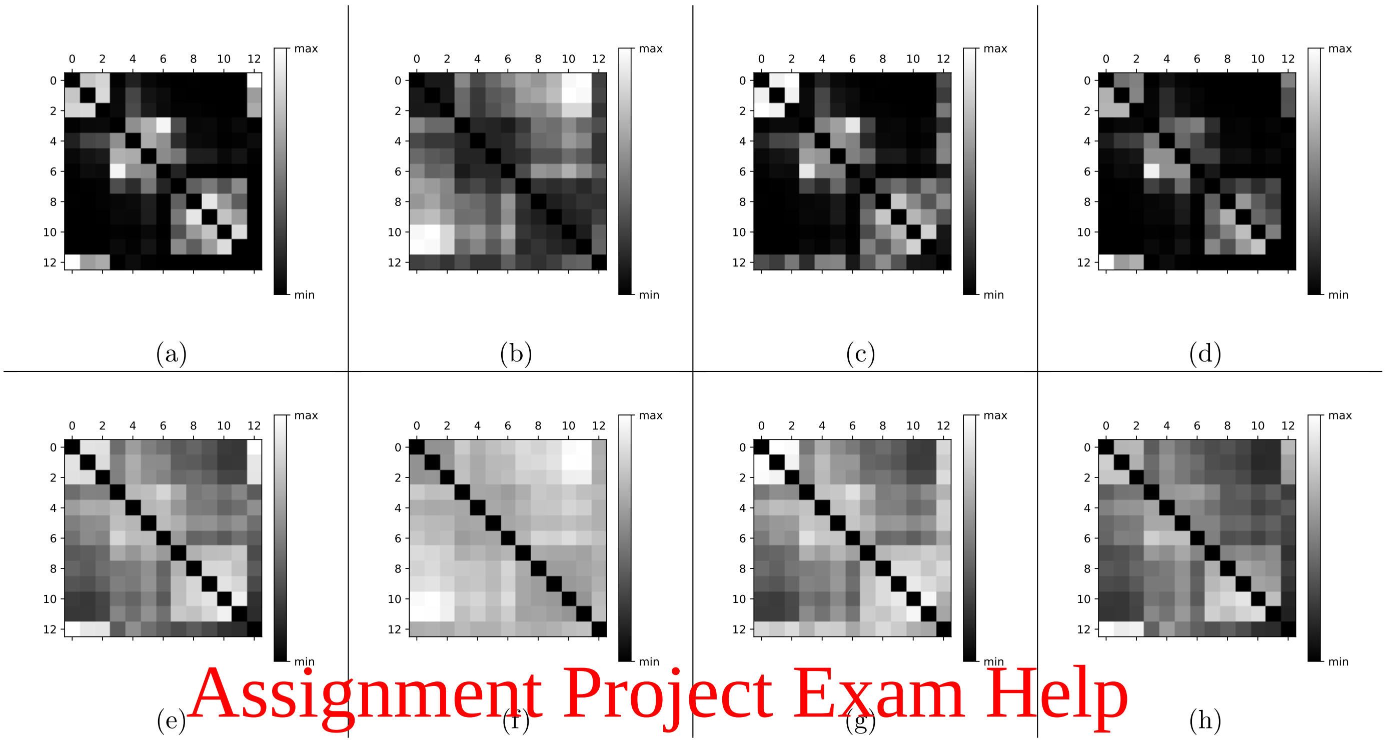


Figure 1: Scatter plot of the data

Which of the following neighbor graph plots (pixel in position i, j shows the value of p_{ij}) corresponds to the given dataset and the stated formula for $\sigma = 2$? What is your answer for $\sigma = 5$? *Justify your answers!*



(a) and (e) are correct for $\sigma = 2$ and $\sigma = 5$, respectively.

1. First column is correct.
2. Second column shows a distance instead of similarity.
3. Third column misses one instance in the lower left cluster and it is located at the center instead $(2.75, 3.5)$.
4. Fourth column shows an asymmetrical matrix.
5. Upper row $\sigma = 2$, lower row $\sigma = 5$.

Autoencoders

Problem 2: We train a linear autoencoder to D-dimensional data. The autoencoder has a single K-dimensional hidden layer, there are no biases, and all activation functions are identity ($\sigma(x) = x$).

- Why is it usually impossible to get zero reconstruction error in this setting if $K < D$?
- Under which conditions is this possible?

We have $f(\boldsymbol{x}) = \boldsymbol{X}\boldsymbol{W}_1\boldsymbol{W}_2$ where $\boldsymbol{X}$ is the data matrix and the dimensions of the weight matrices are $D \times K$ for $\boldsymbol{W}_1$ and $K \times D$ for $\boldsymbol{W}_2$.

The final multiplication $\boldsymbol{W}_2$ brings points from K-dimensions up into D-dimensions but the points will still all be in a K-dimensional linear subspace. Unless the data happen to lie exactly in a

K-dimensional linear subspace, they can't be exactly fitted.

Coding Exercise

Problem 3: Download the notebook `exercise_11_notebook.ipynb` and `exercise_11_matrix_factorization_ratings.npy` from Moodle. Fill in the missing code and run the notebook. Convert the evaluated notebook to PDF and append it to your other solutions before uploading.