Exercises 9: Manifold learning

Introduction to data science and Python programming for medical research (2025)

If you have not done so already you need to install the sklearn package. For windows users this would require you to open your conda terminal, found in the start menu, and then typing:

```
conda install sklearn
```

UMAP is not available in sklearn and needs to be loaded from a separate package.

```
conda install -c conda-forge umap-learn
```

1 Isomap matrix construction

In the code provided for this exercise you need to replace the line stating `B=np.eye(N)` with an expression that computes the matrix

$$B = -\frac{1}{2} C G_2 C$$

such that the isomap implementation will work.

No other lines should be changed.

Note: Make sure you use matrix multiplication and not regular array multiplication!

The result of running the script will be a figure with three subplots, one of the original 3D swiss roll data and two that shows the isomap unrolling of the data, of which one is computed with our own isomap implementation.

2 tSNE of cancer associated fibroblast data

Use the provided script to load the CAF data and apply tSNE on the data.

Look at the examples from the lecture and try different values for the perplexity parameter.

Your script should:

- Load the CAF data
- Perform tSNE to reduce it to 2 dimensions.
- Plot the transformed 2D data using a scatter plot.
- **Optional:**Measure how long it takes for the tSNE `fit` function to complete. Look in the slides from lecture 2 to see how time can be measured.

Note: Add a logarithmic transform to the data matrix before you apply tSNE! Look at the examples from the lecture!

https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html

3 UMAP of cancer associated fibroblast data

Use the provided script to load the CAF data and apply UMAP on the data.

Look at the examples from the lecture and try different values for the perplexity parameter.

Your script should:

- Load the CAF data
- Perform UMAP to reduce it to 2 dimensions.
- Plot the transformed 2D data using a scatter plot.
- **Optional:**Measure how long it takes for the UMAP `fit` function to complete. Look in the slides from lecture 2 to see how time can be measured.

Note: Add a logarithmic transform to the data matrix before you apply UMAP! Look at the examples from the lecture!

https://umap-learn.readthedocs.io/en/latest/

The prepared CAF data was originally downloaded from:
https://github.com/NikolayOskolkov/HowUMAPWorks