Advanced Data Visualization

CS 6965

Fall 2019

Prof. Bei Wang Phillips

University of Utah

Project 01

Overview

- Project 01 contains two parts, a regular and a bonus part
- Part 1: Mapper (15 pts)
- Part 2: UMAP (10 pts, bonus)
- You have a few options
  - Option 1: complete Part 1 only (15 pts)
  - Option 2: compute the last question of Part 1 (your own data set) together with Part 2 (5 plus 10 = 15 pts)
  - Option 3: complete both Part 1 and Part 2 (25 pts)
- Please ignore the percentage calculation from Canvas; due to bonus points, these calculations by Canvas are meaningless.

Project Submission: Report + Code

- Report should contain answers to Q1 to Q5. It should also contain description and explanation associated with your own dataset.
- Source code: A ZIP file that contains the Python programs together with data files and screen shots (PDF). Each program is expected to run properly. There is no partial credit if the program does not run or does not give the expected result.
- Report (PDF) and source code (ZIP) should be submitted in a single ZIP file via Canvas.

Part 1:Mapper

Getting started

- Follow the instruction to install Kepler Mapper:
  - https://github.com/scikit-tda/kepler-mapper
- Note:
  - Depending on your version of Python, you might use pip3 (for Python3, recommended) instead of pip (for Python2.7)
  - You will also need to install scikit-learn, scipy, matplotlib

```
beis-mbp-2018:software beiphillips$ git clone https://github.com/MLWave/kepler-mapper
Cloning into 'kepler-mapper'...
remote: Enumerating objects: 3635, done.
remote: Total 3635 (delta 0), reused 0 (delta 0), pack-reused 3635
Receiving objects: 100% (3635/3635), 17.59 MiB | 10.28 MiB/s, done.
Resolving deltas: 100% (2217/2217), done.
beis-mbp-2018:software beiphillips$ cd kepler-mapper
beis-mbp-2018:kepler-mapper beiphillips$ pip3 install -e .
```

```
pip3 install networkx
pip3 install -U scikit-learn scipy matplotlib
```

Cat Example (5 points)

- Go to the examples folder, delete files in output folder
- Try to get the cat example to run: if Kepler Mapper is installed successfully, you should see cat.html generated in the output folder

```
beis-mbp-2018:kepler-mapper beiphillips$ cd examples
beis-mbp-2018:examples beiphillips$ python3 plot_cat.py
```

Cat Example (continued)

- (2 Pts) Modify plot_cat.py and rename it as plot_cat_test1.py
- Change the mapper interval overlap parameter from 20% to 80%
- Answer the following question Q1: What is the effect of increasing interval overlap parameter on the final graph in the visualization?

- (3 Pts) Modify plot_cat.py and rename it as plot_cat_test2.py
- Change the mapper parameter that deals with the number of intervals (per dimension) from 15 to 30.
- Answer the following question Q2: What is the effect of increasing number of interval parameter on the final graph in the visualization?

Digits Example (5 points)

- Go to the examples folder
- Run the default digit example: if Kepler Mapper is installed successfully, you should see digits_custom_tooltips.html and digits_ylabel_tooltips.html generated in the output folder

```
beis-mbp-2018:examples beiphillips$ python plot_digits.py
```

Digits Example (continued)

- (1 Pt) Run plot_digits.py twice, and answer the following question Q3: Why are the results not necessarily identical?
- (2 Pts) Modify plot_digits.py and save it as plot_digits_test1.py such that it uses Spectral Embedding as part of the projection, with parameters, n_components = 2, random_state = 0, and eigen_solver equal to “arpack”. Observe the results and answer the following question Q4: What is the difference between the results using Spectral Embedding in comparison to the results using t-SNE?
- (2 Pts) Modify the parameters for Spectral Embedding in plot_digits_test1.py and save the file as plot_digits_test2.py so that the resulting clusters (digits) are better separated. Answer the following question Q8: What is your modification and its effect on the data? (Hint: consider modifying the dimension of the projected subspace).

Your own dataset (5 points)

- Apply the Mapper framework to a dataset of your own. You could work with a 3D point cloud (similar to the setting of Cat Examples); or you could work with a high-dimensional example (similar to the breast-cancer example).
- Your point cloud data should have at least 200 points (if you are not sure, please speak with the instructor).

Your own dataset (continued)

- (2 Pts) Prepare your data in csv format and save it as mydata.csv. The data should be cleaned and readily usable by KeplerMapper.
- (3 Pts) Apply KeplerMapper to your dataset and give a description as what insights one might obtain from the results. Your code should be named as mydata.py. Your code should be able to run properly without error and give meaningful results. (If you are unsure, ask the instructor). There is no partial credit for a program that does not run.

Possible datasets

- http://www.pointclouds.org/news/2013/01/07/point-cloud-data-sets/
- http://www.wolframalpha.com/ or https: //github.com/caesar0301/awesome-public-datasets.
- Mesh data set at (http://people.csail.mit.edu/sumner/research/deftransfer/data.html). In this case, you would need to convert the data to the cvs format.

Part 2:UMAP

UMAP: Set Up

- Uniform Manifold Approximation and Projection
- Follow the instruction to install UMAP:
- https://umap-learn.readthedocs.io/en/latest/
- https://github.com/lmcinnes/umap

- Again, using pip or pip3
- The installation is nontrivial
- You might need various packages: bokeh

```
pip uninstall umap

pip install umap-learn
```

UMAP: Set Up

- Uniform Manifold Approximation and Projection
- Follow the instruction to install UMAP:
- https://umap-learn.readthedocs.io/en/latest/
- Again, using pip or pip3
- The installation is nontrivial
- You might need various packages: bokeh
- If needed: replace import umap by import umap.umap_ as umap for each python program

```
wget https://github.com/lmcinnes/umap/archive/master.zip
unzip master.zip
rm master.zip
cd umap-master
```

Default Example (5 pts)

- Install UMAP source code
- Go to examples folder
- (2 pts) Get both digits.py and iris.py to run properly (under folders digits and iris). The default code might have to be modified.
- (2 pts) Change n_neighbors=50 to 30 for iris.py and answer the following question: how different are the results?
- (1 pts) What's the main difference between UMAP and other non-linear DR techniques, such as t-SNE?
- (5 pts) Modify digits.py or iris.py using your own dataset (dimension 3 or higher), name your file mydata.py, and visualize the result as HTML.

Thanks!

Any questions?

You can find me at: beiwang@sci.utah.edu

CREDITS

Special thanks to all people who made and share these awesome resources for free:

- Presentation template designed by Slidesmash
- Photographs by unsplash.com and pexels.com
- Vector Icons by Matthew Skiles

Presentation Design

This presentation uses the following typographies and colors:

Free Fonts used:

http://www.1001fonts.com/oswald-font.html

https://www.fontsquirrel.com/fonts/open-sans

Colors used