Data Visualization Concepts

Prof. Dr. Renato Pajarola

Exercise and Homework Completion Requirements

1. Exercises and reading assignments are **mandatory.** They must be completed successfully to finish the class and get a sufficient passing final grade.
2. Exercises are graded coarsely into categories **pass** or **fail**.
   - *A **fail** is given to failed submissions and incomplete solutions, and no points are awarded.*
   - *A **pass** indicates that the exercise is sufficiently good to receive the corresponding points.*
   - *Late submissions (up to one day) will result in a reduction of one point. Submissions after more than one day will not be accepted or graded.*
3. The four exercises give ascending points in the following distribution: 2 – 3 – 5 – 5.
   - *A **minimum of 7 points** from all four exercises must be achieved to pass the module. Failure to achieve this minimum will result in a failing grade for the entire module.*
   - *Thus at least two exercises have to be correctly solved, and one has to be from the more advanced ones.*
4. We give **bonus points** to students who have completed more than 8 points from all the exercises.
   - *Thus **7 points** from the exercises are required, **8 points** are still a normal pass, and **9 and above** would give 1 or more extra bonus points.*
   - *Only the bonus points can and will be added directly to the final grade.*
5. Do not copy assignments, tools to detect copying and plagiarism will be used.
   - *The exercise results are an integral part of the final course grade, therefore the handed-in solutions to the exercises **must be your personal work**.*

Submission Rules

- *The deadline for submitting Exercise 3 is Sunday, 7 May 2023 at 23:59h.*
- Please submit your solution code via OLAT with file name **'dvc_ex3_MATRIKELNUMBER.py'**.
- If additional packages are used other than the ones in the 'environment.yml' file, please specify them in a **'readme.txt'** file.
- The code should run without errors and generate the bokeh app that contains the expected interactive visualizations.

Exercise 3

In this exercise, you will build an interactive visualization app for the analysis of a high-dimension dataset. The dataset contains information on 750 public companies in the tech industry with 5 categorical columns and 102 numeric columns. You will first process the dataset with the principal component analysis (PCA) technique to reduce the 102 numeric features to 2 principal components, based on which you will cluster the companies and assign cluster labels to them. Then, you will visualize the principal components in a scatter plot (the PCA plot) and apply a color map to the data points. The color map is generated from one of the original features. In addition, you will make a subplot that shows the statistic of an original feature, for both all the data points, and the selected points in the PCA plot by a lasso selection tool. You will use two Select widgets to choose features for the color map and for the subplot. Please read the comments and refer to the examples in the code skeleton for more details and instructions.

Task 1: Dimension Reduction

1.1 Reduce the 102 numeric features to 2 dimensions using PCA.

1.2 Cluster the data points based on the principal components and assign cluster labels to them.

Task 2: Visualization

2.1 Define a function to create a color map of the feature selected for the PCA plot.

2.2 Define a function to plot the principal components in a scatter plot.

2.3 Define a function to draw the histogram for a numeric feature.

2.4 (Optional) Define a function to draw a bar chart for a categorical feature.

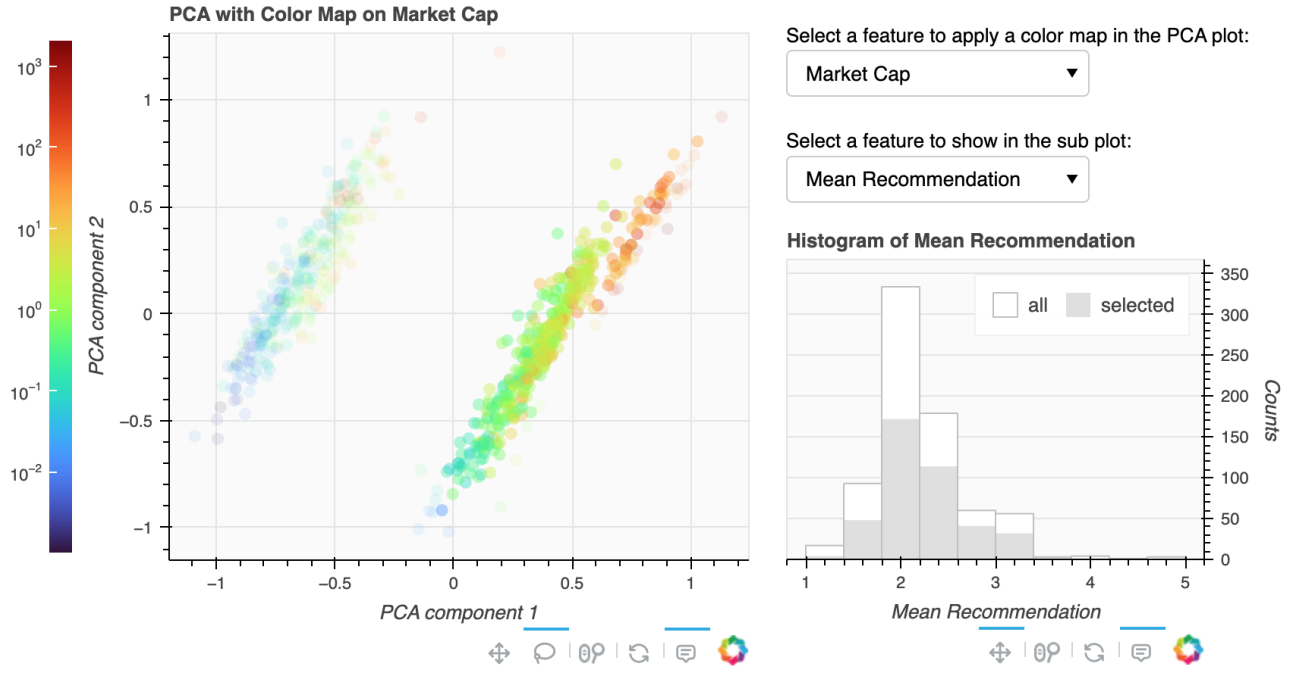
Task 3: Interaction

3.1 Add two Select widgets. One is to select a feature to generate the color map for the PCA plot. It should include at least one categorical feature in the options. Another is to select a feature to show in the subplot. It is optional to include any categorical feature in the options.

3.2 Define the callback functions for the Select widgets.

3.3 Define the callback function for the lasso selection tool in the PCA plot.

Run your app with the bokeh server. An example app looks like this demo (not necessarily the same):



Remarks:

- The code skeleton is structured into sections corresponding to the tasks. You are free to change the skeleton and rewrite the code in your own way.
- We recommend starting with the examples in **Bokeh Tutorial 11. Running Bokeh Applications - Bokeh Apps with bokeh serve**. Please note the difference between creating bokeh apps in a notebook and in a Python script.
- You can first try to implement the dimension reduction and visualization in a notebook, then integrate these parts into the Python script and add the interactions. Please note that the final submission of your code should be a .py file.
- The comments and references in the skeleton code are important for completing the tasks.
- Try Google first for any errors. Chances are good that someone else has solved the problem.
- If Google cannot help, please use the OLAT forum to post technical questions regarding the exercise.
- The Q&A session for Exercise 3 will be on Tuesday, 02 May 2023, from 17:00 to 18:00. The place will be announced in time.