Prof. Dr. Thomas Schultz
Mohammad Khatami (khatami@cs.uni-bonn.de)
Ikram Jumakulyyev (ijumakulyyev@cs.uni-bonn.de)

University of Bonn
B-IT/Computer Science II
May 29, 2020

Summer term 2020

Visual Data Analysis

Assignment Sheet 7

Solution has to be uploaded by June 15, 2020, 8:00 a.m.
to `https://uni-bonn.sciebo.de/s/s8N8JMw0riz7cLX` with password `vda.2020`

Please bundle the results (as PDF) and scripts (*.py/*.ipynb files) in a single ZIP file. Submit each solution only once, but include names and email addresses of all team members in the PDF and each script. Name the file `vda-2020-xx-names.zip`, where `xx` is the assignment sheet number, and `names` are your last names.

If you have questions concerning the exercises, please write to our mailing list: vl-scivis@lists.iai.uni-bonn.de.

Exercise 1 (Interactive Visualization with Dash, *34 Points*)

In the lecture, we repeatedly emphasized the importance of interaction (such as brushing and linking) in computer-based data visualization. In this exercise, you will learn how to implement an interactive visualization in the framework Dash, which we introduced in last week's assignment. You should find the Dash tutorial at `https://dash.plotly.com/` useful for solving this exercise.

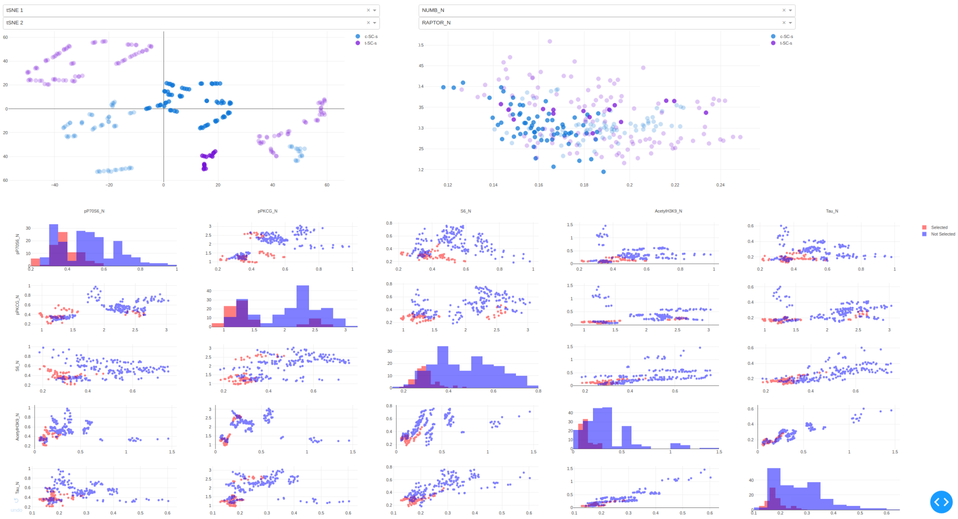

Figure 1: A screenshot of the interactive visualization you are asked to implement in this exercise.

a) Read the file `Data_Cortex_Nuclear.xls`, and restrict it to the classes t-CS-s and c-CS-s. Run PCA, ISOMAP and t-SNE as dimensionality reduction techniques. Within the Dash framework, create a single scatter plot that will show the output from one of the techniques, as selected by the user. Add a **dropdown** component from Dash to switch between the different techniques.

A **callback** function is a function which is triggered by an event such as a mouse click, double click, or selection. In Dash, you can use **@app.callback** to define your event callback, with the dropdown **value** as the input. (10P)

b) Add a second scatter plot with corresponding dropdown menus that should give you the opportunity to map any individual feature (i.e., protein) to any of the axes. (5P)

c) Implement brushing and linking: Modify the previously defined callbacks so that points belonging to a selection made in the first (dimensionality reduction) plot is highlighted in that same plot, and the corresponding samples are also highlighted in the second plot. You can use **selectedData** as the corresponding input for your callback functions. *Note:* Dash only permits a single callback per plot, so you need to modify the previously defined functions, you cannot add new ones. (5P)

d) In the PCA projection, you should notice a cluster of samples from the c-SC-s class that get mixed with samples from the t-SC-s class. Submit screenshots that show where these points project to when using ISOMAP and t-SNE. Do these nonlinear techniques manage to separate them from the other class? (2P)

e) Add a 5 × 5 scatterplot matrix, similar to the one you made in Sheet 6. This time, it should automatically display the five most relevant attributes, as judged by the F score (Sheet 3). When no selection has been made, the F score should be computed with respect to the classes t-CS-s and c-CS-s. Whenever the user makes a selection in the dimensionality reduction plot, F scores should be computed with respect to selected vs. unselected data, and the scatterplot matrix should be updated to show the corresponding top 5 highest-ranking features. Colors in the scatterplot matrix should reflect the classification that is currently used for feature ranking. (10P)

f) Select a cluster in the t-SNE embedding and report how samples within this cluster differ from the rest with respect to the expression levels of specific proteins. Submit a screenshot that illustrates your reasoning. (2P)

Exercise 2 (GAN Lab, *12 Points*)

In the lecture, we will discuss several visualization strategies that aid our understanding of how fully trained neural networks achieve their tasks. Another goal of visualization is to provide users with a general understanding of how specific types of neural networks work, and of challenges in training them. In this context, Kahng et al. recently presented GAN Lab, a browser-based interface for learning about Generative Adversarial Networks. Their system is available from https://poloclub.github.io/ganlab/, which also links to a corresponding scientific paper (at the bottom). Based on reading about and trying out this system, please answer the following questions in your own words.

a) As explained in the paper, GANs aim to find a mapping that transforms random noise into a distribution that is as similar as possible to the distribution of a given reference dataset. Which part of the interface allows you to check visually whether this goal has been met? Which number that is displayed in the interface quantifies this? (2P)

b) Run the training from scratch (without using the pre-trained model), with the default dataset (“mixture of Gaussians”) and settings, for exactly 1000 epochs. Take a screenshot and repeat the experiment. Did you obtain the same result in both cases? Why? (2P)

c) Watch a few epochs in slow-motion mode. Focus on the gradients (pink lines) and the movement of the pink points from iteration to iteration. Do the pink points always follow the gradient direction? Why? (2P)

d) When training a neural network for classification, successful training goes along with a substantial reduction in loss. However, this is not what we observe in this interface. Briefly explain why. (1P)

e) Continue running the training with the default parameters, for a large number of epochs (e.g., 5000). Did it converge to a stable state? Propose a strategy to improve the convergence behavior. (2P)

f) As explained in the paper, mode collapse is a frequent problem in the training of GANs. In the "three disjoint region" dataset, it happens when all generated samples fall into a single cluster (*Note:* They do not necessarily have to collapse into a single point, as in the example shown in the paper.). Can you reproduce this problem? Can you provoke this even in the "mixture of Gaussians"? Please submit corresponding screenshots, and briefly describe what you did. (3P)

Exercise 3 (Discrete Convolution and Cross-Correlation, *4 Points*)

Convolution and cross-correlation are linear operations. As such, they can be expressed in matrix notation. Assume two discrete one-dimensional functions g and h are represented as column vectors $\mathbf{g}, \mathbf{h}$ with coefficients $g_i, i \in \{1, \dots, n\}$ and $h_j, j \in \{1, \dots, m\}$ with $m = 2k + 1$, respectively.

a) Write down a matrix H that, when the vector $\mathbf{g}$ is multiplied to it from the right ($\mathbf{f} = H\mathbf{g}$), has the effect of convolving g with h ($f = g * h$). Assume a stride of one and no padding. Specify the number of rows and columns in H, and the pattern of its coefficients in terms of h_j. (3P)

b) How do you need to modify the matrix to achieve a cross-correlation $f = g \otimes h$ instead of a convolution? (1P)

Good Luck!