

Data Visualization Concepts



BINF4234

Dr. Alexandra Diehl

Exercise and Homework Completion Requirements

1. Exercises and reading assignments are **mandatory** and they must be completed successfully to finish the class and get a sufficient passing final grade.
2. Exercises are graded coarsely into categories **incomplete**, **partial complete** or **complete**.
 - An **incomplete** is given to failed submissions, thus no bonus points are awarded.
 - A **partial complete** is given to partial solutions, and different points will be awarded according to the level of completion.
 - A **complete** indicates that the exercise is sufficiently good to receive all the points.
3. A **minimum of 5 points** from the exercises 1, 2, and 4 must be achieved to pass the module, whereas exercises 3 and 5 are optional and can offer you bonus points as described in Point 4.
 - Failure to achieve this minimum will result in a failing grade for the entire module.
 - Hence at least the first two, or the fourth exercise has to be fully solved.
4. The five exercises give rise to the following point distribution: 2 – 3 – 2 (**optional bonus points**) – 5 – 3 (**optional 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 and therefore the handed in attempts and solutions to the exercises **must be your personal work**.

Submission Rules

- Submitted code must compile and run without errors using the indicated Python environment, using the included libraries, packages and frameworks. If additional libraries/packages are needed, please specify in your 'readme.txt' file.
- The whole project source code must be zipped and submitted before the given deadline, including the output results (saved in .html file or a screenshot picture).
- Submit your .zip archive named **dvc_ex4_MATRIKELNUMBER1_MATRIKELNUMBER2.zip** (e.g. **dvc_ex4_01234567_01234568.zip**) through the OLAT course page.
- **Deadline is Wednesday, 11 December 2019 at 23:59h**

Exercise 4

After learning the basic data manipulation using Python in exercise 1 and single view data visualization in exercise 2, in this exercise, you will work in groups to accomplish a multiple-view data visualization application based on the eBrid dataset. The main purpose of this exercise is: 1) getting familiar with the data visualization pipeline and being able to finish a complete visualization work from scratch, 2) learning and applying interactive data visualization techniques like linking and brushing, and 3) applying the visual design concepts into real applications, like color encoding and highlighting.

To complete this exercise, you need to work in groups of **two** or **three** and finish a **two-view** or **three-view** interactive data visualization, respectively. The specific requirements are described below:

Task1: For each student

- a. Implement one single-view visualization with necessary widgets (e.g. high-dimensional visualization, geo-visualization, temporal visualization, network visualization, etc.) using eBird dataset;
- b. Implement interaction via event handling (https://docs.bokeh.org/en/0.12.16/docs/user_guide/interaction/widgets.html).

Task2: For each group

- c. Link all the two (or three) views together by linking and brushing:

You should be able to pick a point on the map, or brush an area on the map or a scatterplot, then the data should be filtered in all the views, and all the views should be updated.

- d. Write a group project report which should include but not limited to:

- specific question(s) you want to address;
- visual design rationale regarding the question(s), i.e. how do you think this/these question(s) can be answered via which type of visual explanation;
- explanation of the functions of each view in detail, i.e. how each view helps for information exploration and/or data analysis;
- findings/conclusion;
- task division.

Suggested division of roles (e.g. for groups of three):

- visual designer: responsible for leading the general visual design for the whole project and finding the best solution to make all the views work coordinately.
- coder: responsible for the final implementation of linking views (**attention**: each group member should still be able to finish one of the views by him/her-self)
- writer: responsible for the writing of the final report (one-page report)

GRADING CRITERIA:

5 points: If both task1 and task2 are successfully completed.

3 points: If only task1 is successfully completed.

0 point: If you miss any required document/file

The deliverables of this exercise will be a clean version of your code with proper comments, any additional files necessary for executing it, a “readme.txt” file for your comments or remarks (if necessary), as well as a one-page report in .pdf format. The absence of any required deliverable files will automatically lead to a **FAIL**.