Ira A. Fulton Schools of
Engineering
Arizona State University

CSE 578: Data Visualization

Dino Fun World Hierarchical Clustering Assignment

Technical Requirements

If you choose to work on your assignment locally, you can use the following versions:

- Python 3.11.3
- Sqlite3
- Pandas == 1.5.3
- Matplotlib == 3.7.2
- Numpy == 1.25.1

Assignment Description

As in your previous assignments, the administrators of the Dino Fun World theme park have asked you, one of their data analysts, to perform a data analysis task in order to help them administer the park. In this case, your task builds upon one of the tasks the administrators previously asked you to perform. In a prior task, you were asked to find the distance between a set of visitor trajectories using a simple edit distance algorithm and report the distances. For this task, you must construct and display a dendrogram of those distances. Again, the administrators of the park have provided a database which contains the information needed.

This assignment consists of only one task, which is to generate a dendrogram. Create this dendrogram using the trajectories of the visitors with the IDs: 165316, 1835254, 296394, 404385, and 448990. When performing clustering over the trajectories to inform the dendrogram, use an average distance over all points in the cluster.

Directions

Accessing Ed Lessons

You will complete and submit your work through Ed Lessons. Follow the directions to correctly access the provided workspace:

1. Go to the Canvas Assignment, "**Submission: Hierarchical Clustering Assignment**".
2. Click the “**Load Submission…in new window**” button.
3. Once in Ed Lesson, select the assignment titled "**Hierarchical Clustering Assignment**".
4. Review the resources provided in the demonstration.
5. When ready, click on the code challenge and start working in the notebook titled “**Assignment6.ipynb**”.

Assignment Directions

The database provided by the park administration is formatted to be readable by any SQL database library. The course staff recommends the sqlite3 library. The database contains three tables, named 'checkin', 'attractions', and 'sequences'. The database file is named 'dinofunworld.db' and is available in the '**/course/data/CSE-578/dinofunworld.db**' path.

Note: Please note that the database file is accessible through the learner submission workspace, which requires establishing a connection with the database. For downloading the dataset and potentially working locally, refer to the overview document page in your course.

The information contained in each of these tables is listed below:

`checkin`:

- The check-in data for all visitors for the day in the park. The data includes two types of check-ins: inferred and actual checkins.
- Fields: visitorID, timestamp, attraction, duration, type

`attraction`:

- The attractions in the park by their corresponding AttractionID, Name, Region, Category, and type. Regions are from the VAST Challenge map such as Coaster Alley, Tundra Land, etc. Categories include Thrill rides, Kiddie Rides, etc. Type is broken into Outdoor Coaster, Other Ride, Carousel, etc.
- Fields: AttractionID, Name, Region, Category, type

`sequences`:

- The check-in sequences of visitors. These sequences list the position of each visitor to the park. If the visitor has not entered the part yet, the sequence has a value of 0 for that time interval. If the visitor is in the park, the sequence lists are the most visited.
- Fields: visitorID, sequence

Using the data provided, create the dendrogram.

Submission Directions for Assignment Deliverables

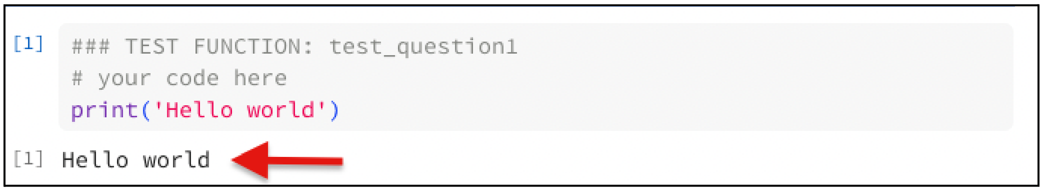
This assignment will be auto-graded. You must complete and submit your work through Ed Lesson's code challenge to receive credit for the course:

1. In order for your answers to be correctly registered in the system, you must place the code for your answers in the cell indicated for each question.

    a. You should submit the assignment with the output of the code in the cell's display area. The display area should contain only your answer to the question with no extraneous information, or else the answer may not be picked up correctly.

    b. Each cell that is going to be graded has a set of comment lines (ex: ### TEST FUNCTION: test_question1) at the beginning of the cell. **This line is extremely important and must not be modified or removed.**

2. After completing the notebook, run each code cell individually or click "**Run All**" at the top to print the outputs.



```
### TEST FUNCTION: test_question1
# your code here
print('Hello world')
```

```
Hello world
```

3. When you are ready to submit your completed work, click on "**Mark**" at the bottom right of the screen.

4. You will know you have successfully completed the assignment when feedback appears for each test case with a score.

5. If needed: to resubmit the assignment in Ed Lesson

    a. Edit your work in the notebook
    b. Run the code cells again
    c. Click "**Mark**" at the bottom of the screen

Your submission will be reviewed by the course team and then, after the due date has passed, your score will be populated from Ed Lesson into your Canvas grade.

Evaluation

There is one part in the grading with a total of 10 points. If the submission fails, we will return the corresponding error messages. If the submission is correct, you will see "The plot is valid" with 10 points for the part.