

Dino Fun World Time Series Analysis 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

The administrators of Dino Fun World, a local amusement park, have asked you, one of their data analysts, to perform three data analysis tasks for the park. These tasks will involve understanding, analyzing, and graphing attendance data that the park has provided for you to use in the form of a database.

Question 1: The park's administrators are worried about the attendance at the ride 'Atmosfear' in the data window. To assuage their fears, they have asked you to create a control chart of the total attendance at this ride. Using the provided data, create a control chart displaying the attendance, mean, and standard deviation bands at one and two standard deviations.

- For this question, display a control chart in the notebook and print the mean and standard deviation as a list. For example, [mean,standard_deviation]

Question 2: Some of the park's administrators are having trouble interpreting the control chart graph of 'Atmosfear' attendance, so they ask you to also provide a moving average chart of the attendance in addition to the control chart created in Part 1. In this case, they request that you use 50 samples for the size of the moving average window.

Question 3: In order to have options concerning the graphs presented, the park's administrators also ask you to provide a 50 samples moving average window with the average computed with exponential weighting (i.e., an exponentially weighted moving average) over the same 'Atmosfear' attendance data.

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: Time Series Analysis Assignment**".
2. Click the "**Load Submission...in new window**" button.
3. Once in Ed Lesson, select the assignment titled "**Time Series Analysis Assignment**".
4. Review the resources provided in the demonstration.
5. When ready, click on the code challenge and start working in the notebook titled "**Assignment4.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.

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:

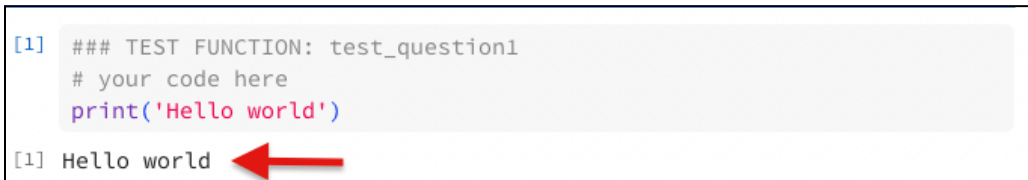
- The check-in sequences of visitors. These sequences list the position of each visitor to the park every five minutes. If the visitor has not entered the park yet, the sequence has a value of 0 for that time interval. If the visitor is in the park, the sequence lists the attraction they have most recently checked in to until they check in to a new one or leave the park.
- Fields: visitorID, sequence

Using the data provided, perform the required analyses and create the requested charts.

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.



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

[1] Hello world

A red arrow points to the output "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 are three parts in the grading, and each part has one test case where the total number of points for all parts is 30. If the submission is correct, you will see "The data used for the chart is correct. The plot is a valid chart." with 10 points for each part. If your output data is correct but the graph is not, you will receive a partial score of 5. **The auto-grader first validates your output data, and if it is correct, it proceeds to evaluate the correctness of the graph.** If the submission fails, the grader will return the corresponding error messages.