Dummy exam: Data science at Disney World

Overview

In this dummy exam, we are looking at data which contains waiting time information for all attraction in Disney World. We first validate the given data, We then investigate a specific business question.

Detailed assignment

Part 1: Setting up your workspace

Initialize a git repository on your personal GitHub account. You should not add the data into this git repo (use a `.gitignore`). In the end, you should provide me with a `.yml` file which I can use to create a virtual environment in which I can run all your code.

Tip: You can start working in a base environment and then create a `.yml` file at the end of the project.

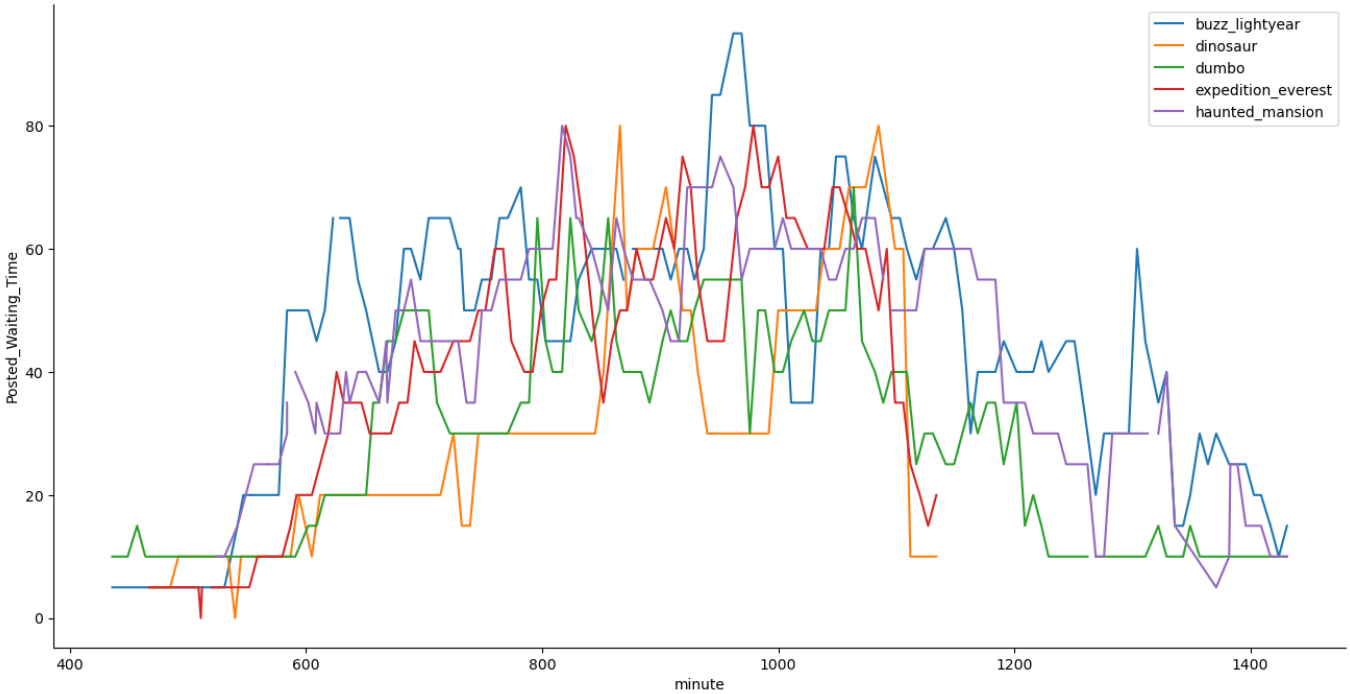
Part 2: Read in & inspect

Step 1: Read in and summarize the data

Read in all waiting time csv files which are located in `data/waiting times`. Collect them in one big dataframe where you add an additional column `attraction` in which you place the name of the attractions (which can be taken from the filename).

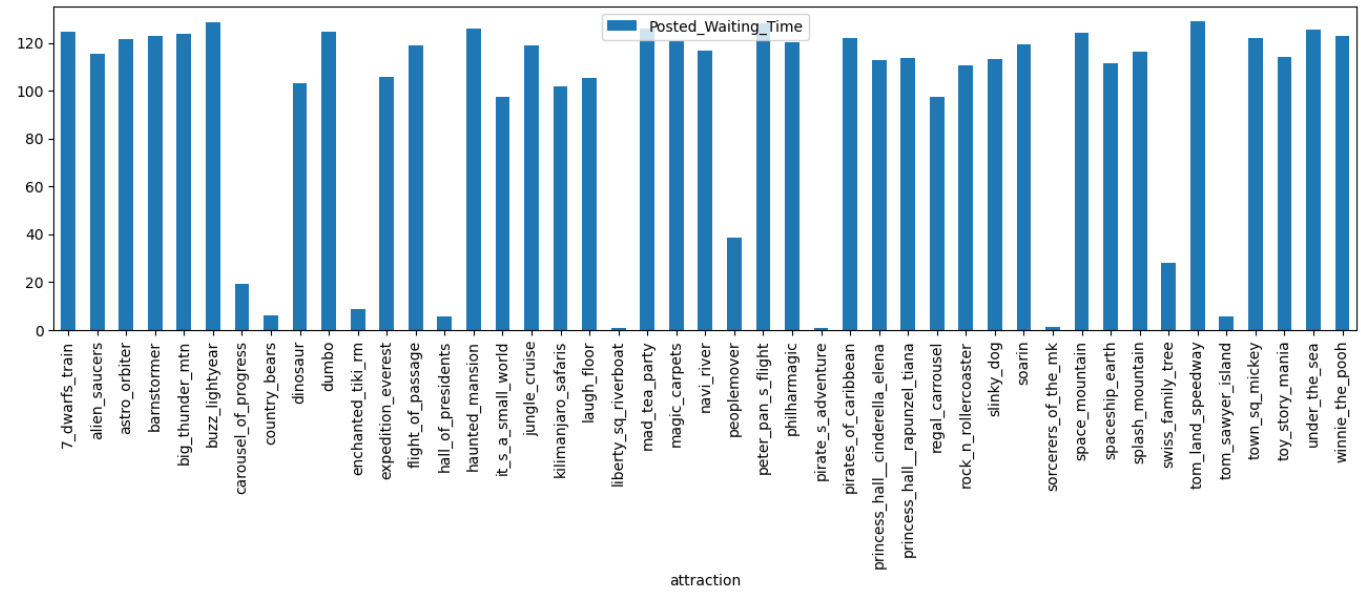
Step 2: Validate the data with some visuals

Create a function which takes in the dataframe you just created, a date and a list of attractions. The function should then create a plot in which you have the waiting time over the given date for all selected attractions. You can use this function to get a feeling for how well filled the actual and posted waiting times are.



Step 3: Study for which attractions we have sufficient information for posted waiting time.

Look into how many data points of `posted waiting time` you have per day for each attraction. This way, you can exclude those attractions with too few data points on average.



Step 4: Study the distribution of data availability for actual waiting time information.

There seems to be not that much information on the actual waiting times at attractions. Is this because you simply have limited data each day, or are there some days with a lot of information? Can you make some visualizations to look into this?

Research Question 1: Compare predicted and actual waiting time

Create a visualization that shows how the predicted and actual waiting times compare, one option would be to execute the following steps:

- For each actual waiting time, find the previous and next posted waiting time.
- Use linear interpolation to find the best estimate of posted waiting time for the actual waiting time we are investigating.
- Compute the difference between the actual and posted waiting time.
- Create a histogram of all these differences, you can also use a scatterplot and a heatmap/2D histogram.

You can then look at some specific dates and attractions for which you have a lot of actual waiting time information for and then plot the actual and posted waiting time together ina single plot.

Research Question 2: best day for rides

Filter out the attractions which have `category_code == ride` and find out which day of the week (Monday/Tuesday/...) these rides have the smallest posted waiting times on average. In a second step, try to further look specifically into the combination of day of the week and `HOLIDAYM` value. Ideally, you should also explain what this means.