

Hands-on Activity 8.1: Aggregating Data with Pandas

8.1.1 Intended Learning Outcomes

After this activity, the student should be able to:

- Demonstrate querying and merging of dataframes
- Perform advanced calculations on dataframes
- Aggregate dataframes with pandas and numpy
- Work with time series data

8.1.2 Resources

- Computing Environment using Python 3.x
- Attached Datasets (under Instructional Materials)

8.1.3 Procedures

The procedures can be found in the canvas module. Check the following under topics:

- 8.1 Weather Data Collection
- 8.2 Querying and Merging
- 8.3 Dataframe Operations
- 8.4 Aggregations
- 8.5 Time Series

8.1.4 Data Analysis

Provide some comments here about the results of the procedures.

8.1.5 Supplementary Activity

Using the CSV files provided and what we have learned so far in this module complete the following exercises:

1. With the earthquakes.csv file, select all the earthquakes in Japan with a magType of mb and a magnitude of 4.9 or greater.
2. Create bins for each full number of magnitude (for example, the first bin is 0-1, the second is 1-2, and so on) with a magType of ml and count how many are in each bin.
3. Using the faang.csv file, group by the ticker and resample to monthly frequency. Make the following aggregations:
 - Mean of the opening price
 - Maximum of the high price
 - Minimum of the low price
 - Mean of the closing price
 - Sum of the volume traded
4. Build a crosstab with the earthquake data between the tsunami column and the magType column. Rather than showing the frequency count, show the maximum magnitude that was observed for each combination. Put the magType along the columns.
5. Calculate the rolling 60-day aggregations of OHLC data by ticker for the FAANG data. Use the same aggregations as exercise no. 3.
6. Create a pivot table of the FAANG data that compares the stocks. Put the ticker in the rows and show the averages of the OHLC and volume traded data.
7. Calculate the Z-scores for each numeric column of Netflix's data (ticker is NFLX) using apply().
8. Add event descriptions:
 - Create a dataframe with the following three columns: ticker, date, and event. The columns should have the following values:
 - ticker: 'FB'
 - date: ['2018-07-25', '2018-03-19', '2018-03-20']
 - event: ['Disappointing user growth announced after close.', 'Cambridge Analytica story', 'FTC investigation']
 - Set the index to ['date', 'ticker']
 - Merge this data with the FAANG data using an outer join
9. Use the transform() method on the FAANG data to represent all the values in terms of the first date in the data. To do so, divide all the values for each ticker by the values for the first date in the data for that ticker. This is referred to as an index, and the data for the first date is the base ([https://ec.europa.eu/eurostat/statistics-explained/index.php/](https://ec.europa.eu/eurostat/statistics-explained/index.php/Beginners:Statisticalconcept-Indexandbaseyear) Beginners:Statisticalconcept-Indexandbaseyear). When data is in this format, we can easily see growth over time. Hint: transform() can take a function name.

In []: