Assignment 2 (15%)

Due: March 11

This assignment requires the use of pandas and concepts of data wrangling from Days 3-4.

Instructions

Submit a Jupyter notebook or Python script through Dropbox or Github. Your submission should include answers to the following questions:

1. ♦ - Navigate to Statistics Canada and find Table 27-10-0273-02, 'Expenditures on research and development (R&D) by performing sector'. Also navigate to Eurostat and find 'Intramural R&D expenditure (GERD) by sectors of performance (rd_e_gerdtot)'. Using pandas, load both datasets into DataFrames, combine the DataFrames, and save the result in a file format of your choice.

2. This problem involves cleaning a dataset so it can be used for analysis.
   a. Load the included dataset called 'A2data.json' into Python using pandas. Rename the columns according to the following scheme:
   `{'id': 'Identifier', 'emp': 'Employment Status', 'ic': 'Income', 'hh': 'Household Pop'}`
   b. Drop the 'Identifier' column'

   c. Compute dummy variables for the column 'Employment Status'.

   d. For the column 'Household Pop', replace any value of 999 with 'NaN'.

   e. Compute the standard deviation and the mean for the 'Income' column. Drop any row that contains a value more than 2 standard deviations from the mean. In other words, if the mean is μ and the standard deviation is σ, then drop any row that contains a value x such that

   $$x > \mu + 2\sigma$$

   or

   $$x < \mu - 2\sigma.$$

Please refer to the rubric at the end of this document for evaluation details.

Requirements

a. Working Jupyter notebook or Python script.

b. Make sure to include clear comments in your code.

Submission

Submit your assignment through D2L Dropbox on Mar 11.

Evaluation

This assignment is graded out of 16 points using the following rubric and is worth 15% of the final grade.

Learners may receive partial scores or a zero for unacceptable work.

Criteria	Does Not Meet Expectations 1	Partially Meets Expectations 2	Meets Expectations 3	Exceeds Expectations 4	Max Points
Code Functionality	Inadequate comprehension of concepts and conventions presented in course materials. Syntax, runtime, and semantic errors throughout.	Partial comprehension of concepts and conventions presented in course materials. Some runtime and semantic errors and almost no syntax errors.	Adequate comprehension of concepts and conventions presented in course materials. No syntax or runtime errors and almost no semantic errors.	Broad and in-depth comprehension of concepts and conventions presented in course materials. No syntax, runtime, or semantic errors.	4
Required Components	Response is missing/does not address required components indicated in the instructions.	Response is missing/does not address some of the required components indicated in the instructions.	Response includes all components and meets the requirements indicated in the instructions.	Response includes and exceeds the requirements indicated in the instructions.	4
Python-specific Code	Offers brute-force solutions to problems. No functions, for/while loops, or iteration is used.	Brute-force solution with no built-in functions. Iteration used incorrectly or inconsistently.	Some iteration used to solve problems. Built-in functions are used incorrectly or inconsistently.	Well written code firmly using Python's particular style. Built-in functions are used when possible, and iteration is used correctly. Some list or dictionary comprehensions used where applicable.	4
Writing Quality	Unclear organization. Many grammatical, spelling, or punctuation errors.	Some signs of logical organization. A few grammar, spelling, or punctuation errors.	Organization supports purpose. Well-constructed body of code. Almost no grammatical, spelling, or punctuation errors.	Organization fully and imaginatively supports purpose. Well-constructed body of code. No grammatical, spelling or punctuation errors.	4