SIT220/731 2022.T3: Task 3P

Working with *numpy* Matrices (Multidimensional Data)

Last updated: 2022-12-12

Contents

1 Task

This task is related to Module 3 (Sections 3.1-3.3; see the *Learning Resources* on the unit site). Aim at submitting your solution by the end of Week 6. In case of any problems/questions, do hot hesitate to attend our on-campus/online classes. The hard deadline is Week 11 (Friday).

Create a single Jupyter/IPython notebook (see the *Artefacts* section below for all the requirements), where you perform what follows.

1. From https://github.com/gagolews/teaching-data/tree/master/marek, download the two following excerpts from the National Health and Nutrition Examination Survey (NHANES dataset):

   - `nhanes_adult_male_bmx_2020.csv`,
   - `nhanes_adult_female_bmx_2020.csv`.

   They give body measurements of adult males and females.

2. Read the two files as *numpy* matrices named `male` and `female`. Each matrix consists of seven columns:

   1. weight (kg),
   2. standing height (cm),
   3. upper arm length (cm),
   4. upper leg length (cm),
   5. arm circumference (cm),
   6. hip circumference (cm),
   7. waist circumference (cm).

3. On a single plot (use `matplotlib.pyplot.subplot`), draw two histograms: for female weights (top subfigure) and for male weights (bottom subfigure). Call `matplotlib.pyplot.xlim` to make the x-axis limits identical for both subfigures (work out the appropriate limits yourself).

4. Call `matplotlib.pyplot.boxplot` to draw a box-and-whisker plot, with two boxes side by side, giving the male and female weights so that they can be compared to each other. Note that the `boxplot` function can be fed with a list of two vectors like `[female_weights, male_weights]`. In your own words, discuss the results.
5. Compute the basic numerical aggregates of the male and female weights (measures of location, dispersion, and shape). In your own words, describe and compare the two distributions (e.g., are they left skewed, which one has more dispersion, and so forth).
6. To the `female` matrix, add the eight column which gives the body mass indices of all the female participants.
7. Create a new matrix `zfemale` being a version of the `female` dataset with all its columns standardised (by computing the z-scores of each column).
8. Draw a scatterplot matrix (pairplot) for the standardised versions of height, weight, waist circumference, hip circumference, and BMI of the females (based on `zfemale`). Compute Pearson's and Spearman's correlation coefficients for all pairs of variables. Interpret the obtained results.

2 Additional Tasks for Postgraduate (SIT731) Students (*)

Postgraduate students, apart from the above tasks, are additionally **required** to solve/address/discuss what follows.

1. Compute the waist circumference to height ratio and the waist circumference to hip circumference ratio of the male and female participants by adding two more columns to the `males` and `females` matrices.
2. Draw a box-and-whisker plot with four boxes side by side, comparing the distribution of the waist-to-height ratio and the waist-to-hip ratio of both male and female participants. Explain what you see.
3. In your own words, list some advantages and disadvantages of BMI, waist-to-height ratio, and waist-to-hip ratio.

3 Optional Features (**)

The following suggestions are not part of the requirements for a pass grade, therefore you can skip them. Nevertheless, you might still want to tackle them, as only practice makes perfect.

1. Print out the standardised body measurements for the 5 persons with the lowest BMI and the 5 persons with the 5 highest BMI (e.g., call `print` for a subset of `zfemale` comprised of 10 chosen rows as determined by a call to `numpy.argsort`). Interpret the results.

4 Artefacts

The solution to the task must be included in a single Jupyter/IPython notebook (an .ipynb file) running against a Python 3 kernel.

Make sure that your notebook has a **readable structure**; in particular, that it is divided into sections. Use rich Markdown formatting (text in dedicated Markdown chunks – not just Python comments).

Imagine it is a report that you would like to show to your manager or clients — you certainly want to make a good impression. Check your spelling and grammar. Also, use formal language.

At the start of the notebook, you need to provide: the **title** of the report (e.g., *Task 42: How Much I Love This Unit*), your **name**, **student number**, **email address**, and whether you are an **undergraduate (SIT220) or postgraduate (SIT731)** student.

Then, add 1-2 introductory paragraphs (an introduction/abstract – what the task is about).

Before each nontrivial code chunk, briefly **explain** what its purpose is. After each code chunk, **summarise and discuss the obtained results** (in a few sentences).

Conclude the report with 1-2 paragraphs (summary/discussion/possible extensions of the analysis/etc.).

Submit one file via OnTrack:

1. the version of the Jupyter/IPython notebook converted to a PDF file (e.g., via *File → Export Notebook As → PDF* or convert to HTML and from that to PDF with your web browser; any method will do).

You do not need to submit the .ipynb file via OnTrack, but you must store it for further reference – a marking tutor might ask for it later, e.g., at the end of the trimester.

5 Intended Learning Outcomes

ULO	Is Related?
ULO1 (Data Processing/Wrangling)	YES
ULO2 (Data Discovery/Extraction)	YES
ULO3 (Requirement Analysis/Data Sources)	YES
ULO4 (Exploratory Data Analysis)	YES
ULO5 (Data Privacy and Ethics)	YES