Stat4ACSAI / Homework 01

Pierpaolo Brutti

Due Wednesday, March 30 (on Moodle)

General Instructions

I expect you to upload your solutions (one per team) on Moodle as a **single running** `Notebook` (`.ipynb`) **named with your surnames** + pictures converted and collected in a single `pdf`-file of your handwritten exercises. Alternatively, a `zip`-file with all the material inside will be fine too.

Your responses must be supported by textual explanations, the code **you** wrote to produce the results and **extensive comments** on both the code and the results.

Please Notice

- Remember our **policy on collaboration**: *collaboration on homework assignments with fellow students is **encouraged**. However, such collaboration should be clearly acknowledged, by listing the names of the students with whom you have had **discussions** (no more) concerning your solution. You may **not**, however, share written work or code after discussing a problem with others. The solutions should be written by **you and your group** only.*

Exercise 1: It's Prob-time (. . . again -.-)!

Suppose that a student's score X in ML is a number between 0 and 1, and that her score Z in Statistics is also a number between 0 and 1. Suppose further that in the population of all ACSAI students in the world (!), these scores are distributed according to the following *joint* pdf:

$$f_{X,Z}(x,z) = \begin{cases} 8 \cdot (x \cdot z) & \text{for } 0 < z < x < 1, \\ 0 & \text{otherwise.} \end{cases}$$

1) Check that $f_{X,Z}(x,z)$ is a legit joint pdf and plot it in Python. What proportion of students obtain a score greater than 0.5 in Statistics, and what is the probability that a randomly selected student will have a Stat-score *exactly equal* to 0.5?
2) Let $W = \log(Z)$ be the log-Stat score. Find and then plot its density in Python. What predicted value of W has the smallest mean squared error (MSE)? Find it analytically. Find also the *median* log-Stat score.
3) Assuming a student got 0.8 in ML, find analytically the best MSE predictor for her Stat-score.

Exercise 2: Stat | 1$^{\texttt{st}}$ contact

The **muon** is an elementary particle with an electric charge of -1 and a spin (an intrinsic angular momentum) of $1/2$. It is an unstable subatomic particle with a mean lifetime of 2.2μs (micro seconds).

Muons have a mass of about 200 times the mass of an electron. Since the muon's charge and spin are the same as the electron, a muon can be viewed as a much heavier version of the electron.

Roughly speaking, muons can be produced in particle accelerators by colliding protons with the nuclei of a specific target. Then the muons quickly decay and the **random angle** X between the by–products (namely positrons and antineutrinos) is distributed as

$$f_X(x \,|\, \alpha) = \frac{1}{2\pi}\big(1 + \alpha \cdot \cos(x)\big) \quad \text{for} \quad x \in [0, 2\pi],$$

where $\alpha \in [-1/3, +1/3]$ is a parameter to be learnt from data.

1. Check that $f_X(x \,|\, \alpha)$ is a valid probability density function (PDF).

2. Find a good, expressive way to visualize (in Python) a few members of this parametric family as α varies.
3. Based on n independent and identically distributed measurements $\{X_1, \ldots, X_n\}$ from $f_X(x \,|\, \alpha)$, find **analytically** (i.e. pen and paper) the method of moments estimator for α.
4. Based on the following small dataset:

$$\boldsymbol{x}_{20} = \{1.7, 5.1, 2.5, 5.6, 6, 0.3, 3.3, 5.7, 3.5, 2.8, 6, 2.8, 4.4, 3.7, 0.6, 5.7, 1.4, 0.2, 1.9, 6\},$$

   evaluate the method-of-moments estimate for α.
5. Write on paper and then implement in Python the log-likelihood function $\ell(\alpha)$ for a generic set of n IID measurements. Then, based on $\boldsymbol{x}_{20}$, plot $\ell(\alpha)$.
6. What about the maximum likelihood estimator for α? Well, let's use `L-BFGS` to find the MLE estimate **numerically** based on $\boldsymbol{x}_{20}$ with and without gradient information (notice that α is *bounded* between $-1/3$ and $+1/3$). Start the optimizer at multiple initial points and then report and comment your results (also in comparison with the method of moments solution).
7. Finally, graphically compare the raw data with the fitted models (MoM and MLE), is there any clear evidence of lack-of-fit? Explain.

Exercise 3: Groupify!

Each year, the *United States Department of Energy* maintains automobile characteristics for thousands of cars: miles per gallon, engine size, number of cylinders, number of gears, etc. Please see their guide for more information.

Large automakers like Toyota and others, have a diverse lineup of cars, trucks, SUVs, and hybrid vehicles, so, can we use clustering techniques (k-means and `MoG`) to categorize these vehicles in a sensible way with the data I made available in `cars21.csv`?

Let's break this down a bit (show the code for all the steps).

1. Import the data, and look at the automakers listed in the variable `make`. Select a large one and obtain a derived dataset by filtering out all the others.
2. Look at the other five **numeric variables** available, and pick the two that you feel are the most relevant to perform the task described. Briefly explain your choice.
3. Check the presence of missing values, filter them out and then visualize and numerically summarize the distribution of the two variables selected. Based on these preliminary analyses, do you still think it was a good choice? You don't have to change your mind, just explain... and *maybe* change...
4. Cluster the data using `MoG` by selecting the number of components via a 70%-30% sample-splitting scheme that you implement from scratch (just the evaluation, for the fit use of course `scikit-learn`). Compare the results with the `AIC` selection (again, use the implementation in `scikit-learn`). Beside the (possibly different) optimal number of components, can you figure out a way to evaluate some form of "agreement" between the two *clusterizations* obtained? In general, do you think that one is any better then the other? Explain, possibly with relevant plots or stats.
5. Repeat the analysis using k-mean by selecting k via the *Elbow method*. Compare with the previous results.
6. [`Bonus`][1] Repeat the analysis with all the 5 numerical variables. Do you see any difference or improvement in the clustering obtained? Explain.

---

[1] *Bonus* means that you can get full grade also without this part...but with it is more likely!