ENSAI **Machine learning** **Computer class**

This is personal work which will take part in your final evaluation: 50 percent of your final grade (the other 50 percent are given by the final exam). In the end each student shall provide a single `ipynb` file containing his or her answers to the questions. The notebook must be sent[1] before Sunday 27th of October, 23:59. Out of 20 points, 5 are specifically dedicated to:

- Presentation quality: writing, clarity, no typos, visual efforts for graphs, titles, legend, colorblindness, etc. (2 points).
- Coding quality: indentation, PEP8 Style, readability, adapted comments, brevity (2 points)
- No bug on the grader's machine (1 point). Latest version of python is recommended but any older version of `python3` should do the job.

You can use https://www.python.org/dev/peps/pep-0008/ to get some info about PEP8. In addition, some of the questions are kind of "open" leaving some room to the students personal developments. The originality of the given answer will be valued.

(Q1) The blessing of independence was illustrated with the help of two key results in the course. The first one was that whenever $Z, (Z_i)_{i\geq 1}$ is are independent and identically distributed random variables with finite order 2 moments, then

$$\mathbb{E}[(n^{-1}\sum_{i=1}^{n} Z_i - \mathbb{E}[Z])^2] = \frac{1}{n}\mathrm{var}(Z)$$

Using some computer experiments and considering a suitable simulation set-up, illustrate the previous statement.

(Q2) The second key result given in the course, is the *central limit theorem*. It asserts that

$$\sqrt{n}(n^{-1}\sum_{i=1}^{n} Z_i - \mathbb{E}[Z]) \rightsquigarrow \mathcal{N}(0, \mathrm{var}(Z))$$

Using some computer experiments and considering a suitable simulation set-up, illustrate the previous statement.

(Q3) From now on we consider the following data generation process. Define $p = 8$ and let X be a Gaussian vector in $\mathbb{R}^p$ such that $X \sim \mathcal{N}(0, I_p)$ where I_p is identity matrix. Let

$$p(x) = \frac{\exp(x_1 + x_2^2)}{1 + \exp(x_1 + x_2^2)}$$

with $x = (x_1, x_2, \ldots, x_p)$. The output Y is given by the following Bernoulli distribution:

$$Y \sim \mathcal{B}(p(X)).$$

Generate an independent and identically distributed collection of $n = 3000$ random variables from this distribution. With the help of simple graphs show that x_1 and x_2 have a particular relationship with the output y compared to the others.

(Q4) (no package should be used here apart from the basics like numpy) Let

$$q_\beta(x) = \frac{\exp(\beta_0 + \beta^T x)}{1 + \exp(\beta_0 + \beta^T x)} \tag{1}$$

[1] `francois.portier@gmail.com`

Define the logistic regression estimator as

$$\hat{\beta}_n \in \arg\min_{\beta \in \mathbb{R}^{p+1}} -\sum_{i=1}^{n} Y_i \log(q_\beta(X_i)) + (1 - Y_i) \log(1 - q_\beta(X_i))$$

Apply GD algorithm to the above. Apply SGD algorithm to the above. Learning rate will be taken as $1/k$, k being the iteration of the optimization algorithm, and the number of evaluation of $q_\beta(X_i)$ shall be the same for each algorithm (take care that one step of GD takes many more evaluations). With the help of several experimental runs visualize the random (and non-randomness) of SGD (of GD).

(Q5) Evaluate the misclassification risk using $n = 1000$ additional generations from the previous data generation process. Compare the two above approaches SGD and GD (with fixed number of evaluation of $q_\beta(X_i)$).

(Q6) Now consider a more flexible model (but more complex) such as

$$q_{\beta,\gamma}(x) = \frac{\exp(\beta_0 + \beta^T x + \gamma^T x^2)}{1 + \exp(\beta_0 + \beta^T x + \gamma^T x^2)} \tag{2}$$

where $x \in \mathbb{R}^p \mapsto x^2 \in \mathbb{R}^p$ is applied component-wise. Train the model with SGD (now two vectors need to be estimated) and compare it in terms of misclassification risk with the SGD from before using the same number of iterations for both SGD.

(Q7) Now we use logistic Lasso with sklearn. Compare the two models (1) and (2) (both trained with logisitc Lasso) using a ROC curve visualization.

(Q8) Import the **spam** database from https://www.dropbox.com/scl/fi/9uz7mu64ew91go651qn11/spam.csv?rlkey=rp07p7kmt27gox1c3sckeehdo&st=yky8ql7n&dl=0 and apply LASSO as well as Ridge to the given data after applying a tokenization of the text feature (other available columns will be removed).