UNIVERSITE DE GENEVE Section de Mathématiques
Faculté des sciences

# Statistical Machine Learning I - 2025-26
# TP 1

These exercises are meant to familiarize yourself with the development environment around python and basic matrix operations using the numpy library. Please install something like Jupyter Notebooks, Pycharm or use Google Colab if you have problems with setting up an environment. You can use matplotlib for visualization.

**Exercise 1** (Matrix normalization)**.** Write a function that takes as input a matrix $M \in \mathbb{R}^{n \times d}$ and returns its normalized version with every column having average 0 and variance 1. Note that this operation does not perform any sort of decorellation.

**Exercise 2** (Veryfying the Law Of Large Numbers and the Central Limit Theorem)**.** We want to generate random numbers and check the Law of Large Numbers and Central Limit Theorem numerically.

- Generate a large amount of i.i.d random numbers $\{X_i\}_{i=1}^n$ with the same mean $\mu$. Plot $\frac{1}{n}\sum_{i=1}^n X_i$ as a function of $n$ and visualize its convergence to $\mu$. Do this for $X_i$ sampled from different distributions and see that it still works. In particular, try with the Student distribution with one degree of freedom, a.k.a., the Cauchy distribution.
- Generate a large amount of i.i.d random numbers $\{X_i\}_i$ with the same mean $\mu$ and variance $\sigma^2$. Visualize the distribution of the terms $Y_n := \frac{1}{\sqrt{n\sigma^2}}\sum_{i=1}^n (X_i - \mu)$ as a history of histograms (or qq-plots) and make clear that they converge to the Gaussian distribution $\mathcal{N}(0, 1)$ for large $n$. Repeat this experiment with random numbers stemming from different distributions. Try with the Student distribution with three degrees of freedom.
- Same exercise as above, but where $\sigma$ is estimated from the sample; can you use the normalizing matrix function from exercise 1 for this? Try with the Student distribution with two and three degrees of freedom.

**IMPORTANT**: Every function and exercise must be tested. Plug in some values for which you know the correct answers and compare the output of your function.