# Factorization-Based Data Modeling
# Practical Work 3

Umut Şimşekli
Télécom ParisTech, Paris, France
`umut.simsekli@telecom-paristech.fr`

**Instructions: (please read carefully)**

1. This homework can be done in groups of **maximum 2** people.
2. Prepare your report as a pdf file in English by using LaTeX or a similar software (Word etc). Do not submit scanned papers.
3. Put all your files (code and/or report) in a zip file: *surname_name_tp3.zip* and submit it to `https://www.dropbox.com/request/Djre0D0PMY0ZcThpX84m`. The deadline is **January 28th, 2020, 23.00**. Late submissions will not be accepted.
4. One submission per group is sufficient.

## 1 Coupled Tensor-Matrix Factorization

In this practical work, the aim is as follows:

1. Design a factorization model for a given problem
2. Derive the multiplicative update rules for this problem
3. Implement the algorithm

Throughout this practical work, we will use the UCLAF dataset[1]. This dataset has a main tensor $X_1$ of size $146 \times 168 \times 5$, which encapsulates *user-location-activity* informations, where $X_1(i, j, k) = 1$ if the user $i$ visits location $j$ and performs activity $k$ there and $X_1(i, j, k) = 0$ otherwise. The dataset also includes additional side information: the user-location preferences matrix $X_2$, the location-feature matrix $X_3$, the user-user similarity matrix $X_4$, and the activity-activity matrix $X_5$. The aim in this application is to predict the missing parts of $X_1$. The data is given in `uclaf_data.mat`.

## 2 Exercises

1. Develop a (non-negative) coupled factorization model for decomposing all these observed matrices/tensors simultaneously.

[1] V. W. Zheng, B. Cao, Y. Zheng, X. Xie, and Q. Yang, "Collaborative filtering meets mobile recommendation: A user-centered approach," in AAAI, 2010

2. Write down the cost function by using the $\beta$-divergence, given as follows:

$$d_\beta(x||\hat{x}) = \frac{x^\beta}{\beta} - \frac{x\hat{x}^{\beta-1}}{\beta-1} + \frac{\hat{x}^\beta}{\beta}. \tag{1}$$

   Use a different $\beta$ for each observed tensor (use $\beta_{1:5}$ for $X_{1:5}$).

3. Explain why your model makes sense.

4. Develop the multiplicative update rules algorithm for the model that you developed in the previous question.

5. Implement your algorithm in MATLAB or octave. Monitor the overall cost function. What are the effect of choosing different $\beta$ for each tensor? When the algorithm converges, check whether the individual model predictions $\hat{X}_{1:5}$ are close to the original tensors or not.