Matrix Factorization
(Non-negative)

Lesson 5 : Lab Session
Advanced Machine Learning, CentraleSupelec

Teacher's Assistant: Omar CHEHAB
Professors : Emilie CHOUZENOUX, Frederic PASCAL

CentraleSupélec | université PARIS-SACLAY

General Information

- **Assignment :** <u>alone or in pairs</u>, you will code the algorithms you learnt in 'scikit-learn formalism', and apply them to images and text.

- **Due :** the 5 lab assignments for lessons 3-7 are due <u>a week</u> from when they are given, at aml.centralesupelec.2020@gmail.com

- **Grading** : each assignment is worth <u>4 points</u> — your <u>4 best labs</u> out of the 5 will be retained and will count for <u>half of your final grade.</u>

- **Questions :** questions or feedback are welcome after class or by email at l-emir-omar.chehab@inria.fr

Lesson: recap

X = D S
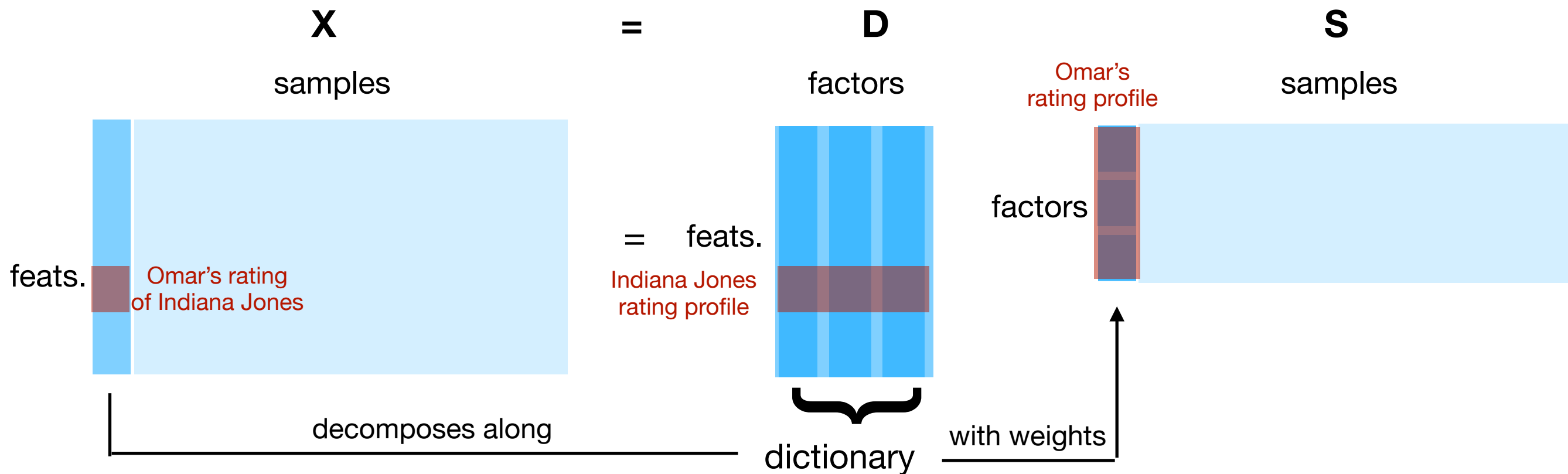
Each *observed* vector $x \in \mathbb{R}^p$ is **embedded** as a *source* vector $s \in \mathbb{R}^r$ via a linear map D

The map is *interpretable* :   clusters   soft-assignment
  its columns form a **dictionary** (~basis) for x that is **weighted** (~coefficients) by s
  its rows provide a **feature embedding**

Application	Recommender	Vision	Video
features	movie ratings	pixels	a timeseries
factor	movie genre rating	image template	timeseries template
sample	user	image	recording

Algorithm

Loss :
penalized Least-Squares

$$\mathscr{L}(D, S) = \frac{1}{2}\|X - DS\|_F^2 + \frac{\mu}{2}\|S\|_F^2 + \lambda\|S\|_1 + \frac{\nu}{2}\|D\|_F^2$$

Updates :
multiplicative

$$D \leftarrow D \circ \frac{XS^\top}{D(SS^\top + \nu I_r)}$$

$$S \leftarrow S \circ \frac{D^\top X - \lambda 1_{r \times n}}{(D^\top D + \mu I_r)S}$$

Food for thought :

We chose classical L1 and L2 constraints, for sparsity and shrinkage.

Other constraints recover well-known linear embedding techniques (PCA, ICA, soft K-Means), as do other losses.

Matrix factorization *is a general framework* for linear embedding.

Assignment: plan

1. NMF : multiplicative updates (*your own code*)

2. Application : vision (*verify your code, then use scikit-learn*)

3. Application : text (*verify your code, then use scikit-learn*)