The objectives of the lab

This practical session aims at showing how to build a simple recommender system on the movie lens data set

Exercise 1 (SVD and weighted SVD on MovieLens)

1. The MovieLens Dataset
   (a) What is the Movie Lens data set? You can look at `https://grouplens.org/datasets/movielens/`
   (b) Why is it preferable to begin with the MovieLens 100K Dataset?
   (c) Download the MovieLens100k dataset that is the ml-100k.zip file (size: 5 MB) from `https://grouplens.org/datasets/movielens/100k/`. The data can be imported in Python with the following code:

```
import pandas as pd
import numpy as np
from numpy import random
import scipy
import scipy.sparse

data_dir = "ml-100k/"
data_shape = (943, 1682)

df = pd.read_csv(data_dir + "u.data", sep="\t", header=None)
values = df.values
```

   (d) What are the differences between the table `df` and the rating matrix M?

2. Data preprocessing
   (a) Use scipy sparse type to obtain a (sparse) rating matrix M

```
M = scipy.sparse.csr_matrix((values[:, 2], (values[:, 0], values[:, 1])), dtype=np.float, shape=
    data_shape)
```

   (b) How is coded the missing data in the sparse matrix?
   (c) Split the data into two matrices. Use 90% for training and 10 % for testing.
   (d) Compute the global mean of the ratings given by the users to the movies
   (e) Center the data and compute the test error when predicting missing values by the mean.

3. Recommending using SVD
   (a) Compute the 20 first factors of the SVD of the centered training data
   (b) Predict the missing test values using the SVD with an increasing number of component (up to 20). Evaluate the performance of this approach on the test matrix and plot the resulting performance as a function of the number of factors of the SVD used to perform the reconstruction.

4. Recommending using the weighted SVD

   We want to improve the predictions by using the weighted SVD, that is solving

$$\min_{U \in R^{n \times k}, V \in R^{p \times k}} J_w(U,V) \quad \text{with} \quad J_w(U,V) = \sum_{i=1}^{n} \sum_{j=1}^{p} w_{ij} (M_{ij} - u_i v_j^\top)^2$$

   where $w_{ij} = 0$ if M_{ij} is unknown (and else $w_{ij} = 1$)

(a) Solve the penalized weighted SVD problem

$$\min_{U,V} \|M - UV^\top\|_W^2 + \lambda\|U\|^2 + \lambda\|V\|^2$$

with $\lambda = 2$ by implementing the penalised Alternating Least Square(ALS) described as follows:
Initialize U and V with the SVD on the full matrix

loop
compute U that $\min_U \|M - UV^\top\|_W^2 + \lambda\|U\|^2$ with a fix V
compute V that $\min_V \|M - UV^\top\|_W^2 + \lambda\|V\|^2$ with a fix U

(b) Can you implement the Funk SVD algorithm which is a Stochastic Gradient Descent on J_w along $U_{i,\cdot}$ and $V_{\cdot,j}$ where the known ratings are sampled uniformly?

(c) Find a better recommendation (you can look at the scikit-surprise package)