Week 12 Project

ACADEMIC HONESTY

As usual, the standard honour code and academic honesty policy applies. We will be using automated **plagiarism detection** software to ensure that only original work is given credit. Submissions isomorphic to (1) those that exist anywhere online, (2) those submitted by your classmates, or (3) those submitted by students in prior semesters, will be detected and considered plagiarism.

INSTRUCTIONS

In this assignment you will implement the probabilistic matrix factorization (PMF) model. Recall that this model fills in the values of a missing matrix M, where M_{ij} is an observed value if $(i,j) \in \Omega$, where Ω contains the measured pairs. The goal is to factorize this matrix into a product between vectors such that $M_{ij} \approx u_i^T v_j$, where each $u_i, v_j \in \mathbb{R}^d$.

The modeling problem is to learn u_i for $i = 1, \ldots, N_u$ and v_j for $j = 1, \ldots, N_v$ by maximizing the objective function

$$\mathcal{L} = -\sum_{(i,j)\in\Omega} \frac{1}{2\sigma^2}(M_{ij} - u_i^T v_j)^2 - \sum_{i=1}^{N_u} \frac{\lambda}{2}\|u_i\|^2 - \sum_{j=1}^{N_v} \frac{\lambda}{2}\|v_j\|^2$$

For this problem set $d = 5$, $\sigma^2 = \frac{1}{10}$ and $\lambda = 2$.

WHAT YOU NEED TO SUBMIT

You can use either Python or Octave coding languages to complete this assignment. Octave is a free version of Matlab. Your Matlab code should be able to directly run in Octave, but you should not assume that advanced built-in functions will be available to you in Octave. Unfortunately we will not be supporting other languages in this course.

.

Depending on which language you use, we will execute your program using one of the following two commands.

.

Week 12 Project: Matrix Factorization
Project due Apr 11, 2017 07:30 MYT

Week 12 Discussion Questions

- Post-Course Survey

Either

```
$ python hw4_PMF.py ratings.csv
```

Or

```
$ octave -q hw4_PMF.m ratings.csv
```

.

You must name your file as indicated above for your chosen language. If both files are present, we will only run your Python code. We will create and input the csv data file to your code.

.

The csv files that we will input into your code are formatted as follows:.

1. **ratings.csv:** A comma separated file containing the data. Each *row* contains a three values that correspond in order to: user_index, object_index, rating

---

WHAT YOUR PROGRAM OUTPUTS

You should write your PMF algorithm to learn 5 dimensions. Run your algorithm for 50 iterations.

.

When executed, you will have your code write several output files each described below. It is required that you follow the formatting instructions given below. Where you see [iteration] below, replace this with the iteration number.

.

`objective.csv`: This is a comma separated file containing the PMF objective function given above along each row. There should be 50 rows and each row should have one value.

.

`U-[iteration].csv`: This is a comma separated file containing the locations corresponding to the rows, or "users", of the missing matrix $\boldsymbol{M}$. The $\boldsymbol{i}$th row should contain the $\boldsymbol{i}$th user's vector (5 values). You only need to create this file for iteration number 10, 25, and 50. For example, the 10th iteration will produce file U-10.csv

.

`V-[iteration].csv`: This is a comma separated file containing the locations corresponding to the columns, or "objects", of the missing matrix $\boldsymbol{M}$. The $\boldsymbol{j}$th row should contain the $\boldsymbol{j}$th object's vector (5 values). You only need to create this file for iteration number 10, 25, and 50. For example, the 10th iteration will produce file V-10.csv

.

Note on Correctness

Please note that for both of these problems, there are multiple potential answers depending on your initialization. However, the PMF algorithm has some known deterministic properties that we discussed in class, and so in this sense we can distinguish between correct and incorrect answers. We strongly suggest that you test out your code on your own computer before submitting.

USE OF VOCAREUM

This assignment uses Vocareum for submission and grading. Vocareum comes equipped with an editing environment that you may use to do your development work. You are **NOT** required to use the editor. In particular, you are free to choose your favorite editor / IDE to do your development work on. When you are done with your work, you can simply upload your files onto Vocareum for submission and grading.

However, your assignments will be graded on the platform, so you **MUST** make sure that your code passes at least the submission test cases. In particular, do not use third-party libraries and packages. We do not guarantee that they will work on the platform, even if they work on your personal computer. For the purposes of this project, everything that comes with the standard Python or Matlab libraries should be more than sufficient.

To check the formatting of your submission, select to have your code submitted, but not graded. We will output the results of the formatting check to SubmissionReport.txt. We can guarantee a very low grade if you do not pass this submission test. Once your outputs satisfy the formatting requirements and you are confident in your code, select to have it graded.

YOU WILL HAVE THIRTY (30) OPPORTUNITIES TO SUBMIT YOUR CODE FOR GRADING.

WORK ON PROJECT (ML.A) (External resource)

(25.0 points possible)

Your email address will be used to identify your submission entry.

Launch Project