Project 4

ESE 545, Data Mining: Learning from Massive Datasets

November 27, 2017

Due at 11:59PM on **December 4, 2017**

This question consists of several parts. You are required to solve the problems using Python, Matlab or C and turn in your code. You are allowed to work in groups of at most two members. You should submit your code with a brief report containing responses to each part. Turn in one report and script per group. Upload a zipped file containing the report and the code on Canvas.

Question 1.1. In this project, we are returning to the MovieLens dataset. For this project you may use a smaller dataset to help with runtime: http://grouplens.org/datasets/movielens/1m/. The goal of this project is to implement a type of recommender system, in which we are asked to choose for example 20 movies to advertise to users of a website. We want to maximize the chance that a user of the site will "like" some of the the movies. Recall that each user rates the movies by a number between 0-5 (if a user has not rated a movie, we simply let the corresponding rating value to be 0). In your python program, construct a suitable representation of the data set in a matrix form. **10 pts**

Question 1.2. The first step is to define an objective function which assigns to each subset of the movies a real value representing how much the users '1ike" that subset. One way to define such an objective function is as follows. Let us first introduce some notation. Let n, m to be the number of users and movies, respectively. We also let $r_{i,j}$ denote the rating that user i assigns to movie j. Given any subset A of the movies, we define $F(A) = \frac{1}{n}\sum_{i=1}^{n} \max_{j\in A} r_{i,j}$. Prove that the objective function F is both monotone and submodular. **30 pts**

Question 1.3. In the next part, we will implement the greedy submodular maximization algorithm described in class. Note that, due to monotonicity and submodularity, the greedy algorithm guarantees a solution A such that $F(A) \geq (1-1/e)F(A^*)$, where A^* is the true optimal set. Implement the greedy algorithm for maximization of F over all the subsets of movies that have cardinality at most k. Plot the objective values of the greedy algorithm versus k for $k = 10, 20, 30, 40, 50$. **30 pts**

Question 1.4. One way to make the greedy algorithm faster is to use the so-called "lazy" version. Expand on your implementation for part 2, and implement the lazy greedy algorithm. You should get the same greedy solution as in the previous part but with a smaller runtime. Record the runtime (in seconds) of both the greedy algorithm and its lazy version for $k = 10, 20, 30, 40, 50$ and plot the values. **30 pts**