Joseph Tuong
z3381097
Advanced Algorithms 14s2

Clustering with Positive Matrix Decomposition for Guided Execution of Entries in a Relational Matrix

Introduction

Positive Matrix Decomposition is a method used in recommender systems for predicting relational data by reducing the dimensionality of a partial matrix. We use this technique to guide the execution of a real-time algorithm where evaluation of an entry in the matrix is expensive, but obtainable. The distance function in this case may be an expensive computation, a prompt for data collection, or an experiment conducted over a period of time. The algorithm uses current predictions to determine which entries to evaluate to maximise it's own predictive abilities on partial data at any given time. Clustering on the matrix will allow us to perform experiments that maximise the information obtained by earlier experiments.
This is useful, for example, in efficiently conducting user experiments to determine what content to show in a newsfeed, by boosting selected individuals and monitoring interaction signals over a period of time. In this paper we will use Last.FM similarity scores in a group of users to determine each user's most-similar users. We will use Last.FM's tasteometer feature to compare users, and test results by running the algorithm under different conditions with different user subsets.

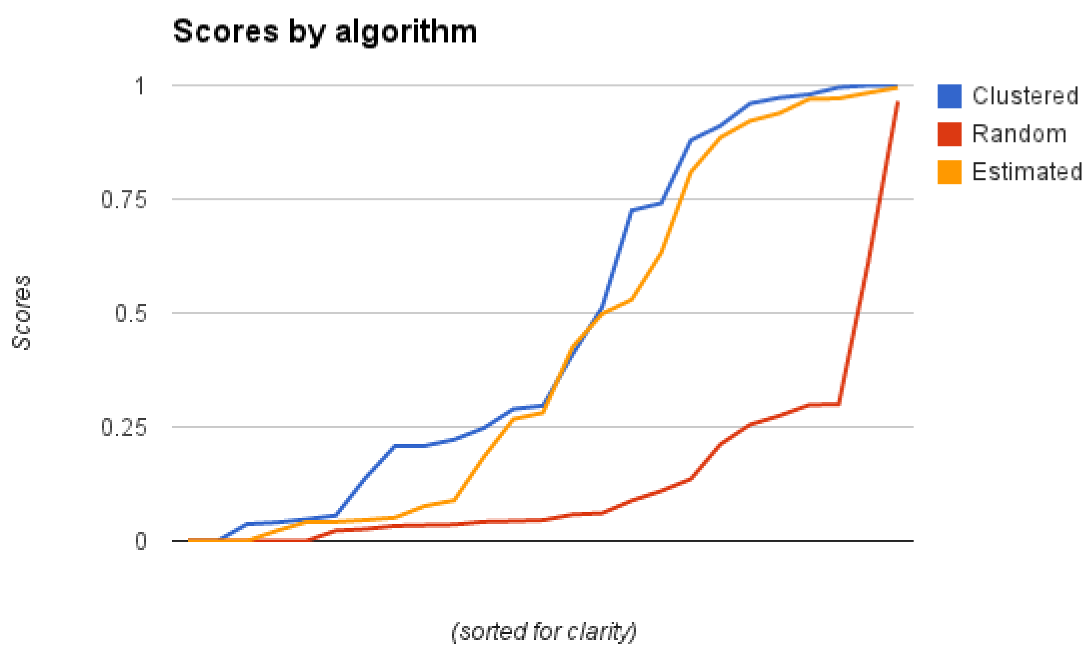
Approach

Positive matrix decomposition produces two matrices which multiply to an approximation of the original matrix. We can conceptualise these matrices as a set of characteristic features associated with each user, and thus we can consider each value assigned by the algorithm a score for some arbitrary "feature" identified by the algorithm. We can thus view PMD as a clustering algorithm, which clusters users by identifying the distance from each user to some abstract "feature".
Finding the most informative unknown to evaluate is then simply a matter of investigating the most characteristic unevaluated user of the most "likely" cluster at any given time. We can use bayesian reasoning to determine the most likely cluster that a user will have the highest affinity, and the most likely user is simply given by the highest un-evaluated activation of the selected feature. We select the most likely cluster by considering the evaluated values in the matrix as weights on each user's affinity to the cluster, i.e it is the weighted average of all the "activations" for each related user in the matrix.

Joseph Tuong
z3381097
Advanced Algorithms 14s2

Data

A dataset of 84 users (awful_users.py, collected from the Awfulnet IRC network) was constructed with affinity ratings collected over a long time period. Due to imperfections in data collection, the resulting matrix (awful_matrix.py) is not perfectly symmetrical. 6543 of the possible 7053 entries have a value.
For simplicity, we fill the matrix with zeros for the evaluation algorithm. The objective is to obtain the best- in this case, highest, scores as soon as possible.
We test this by running the selection algorithm 25 times, dropping 10% of the original matrix. Two alternative algorithms are provided for comparative purposes- picking the maximum estimated entry by positive matrix decomposition, and random choice.



The average for each run are as follows:

Clustered	14.6%
Random	42.6%
Estimated	47.5%

We can see that this approach provides a marginal boost over taking the largest estimated value.