**Prediction for Yahoo Music Users**

**Stewart DeSoto**

A matrix factorization algorithm was implemented to predict user ratings (scale from 0 to 100) using stochastic gradient descent learning approach.

The set of users was too large (n\_users = 249,012) to fit easily into memory on the computer used, so two small subsets (5000 and 20,000) were used instead. The dataset includes ratings on a number of categories, including track, artist, etc. To make the project manageable, it was decided to predict a Yahoo users rating for a particular musical artist, rather than a specific track. The number of unique artists in the set is n\_artists = 18,674.

In the matrix factorization approach, it is assumed that there are a relatively small number of categories (latent features) that artists all into. For example, in this case it could be ska, reggae, EDM, country, rockabilly, etc. The output of the algorithm are two matrices. The User Matrix contains the preference of each user for each of these hidden categories, and the Artist Matric contains the actual classification of each artist in terms of these same categories. Thus a simple dot product determines the predicted value of a user rating for a given artist.

The Python code I developed implements this matrix factorization. Then to estimate the accuracy of the recommendation engine, the predicted rating for each entry in the actual ratings matrix are compared, and a root mean square error (RMSE) of the entire ratings matrix is calculated. Our results are summarized in the table below. As a reminder, the ratings are in the range 0-100 in steps of 10.

|  |  |  |  |
| --- | --- | --- | --- |
| **RMSE (values in 0-100)** | **# of Users Used** | | |
| **# of Latent Features** | **1,000** | **5,000** | **20,000** |
| **15** | 20.7 | 19.2 | 17.9 |
| **50** | 17.8 | 17.8 | 16.7 |

If we had more time to devote to the project, there are many significant improvements we would make. Firstly, the RMSE quoted is that from the training set. It is always essential to divide the data into training and test sets, and to only use out of sample, or test, data to estimate the accuracy of future predictions. In addition, it is important to use k-fold cross validation (often k=10) to better estimate out of sample errors, and to determine appropriate values for meta-parameters (such as the number of latent features to use in the present study). Finally, it is clear from the data that using more users (and thus a more iterations of stochastic gradient descent) improves the accuracy of the predictions.