Gimme Food by The Yelping Stones

Mark Kwon

Problem

Use the dataset from the Yelp Dataset Challenge to use collaborative filtering to recommend restaurants to users based on their review history.
Instead of recommending, the problem is translated to predicting the ratings of the users.

Dataset

The dataset needed to be relatively dense and also small enough to be feasible, so the restaurants and users in the city "North Las Vegas" were chosen, which contains 139 users, 213 restaurants and 1447 reviews, which has around 5% density.
70% of the users were used as the training set and 30% of the users as testing set. 30% of the reviews of the testing set users were hidden for testing.

Example of data (with features used in method):

- business: {id: '-IM6Y4Z2pZ21xvZl1mHUSA', attributes:{'Wheelchair Accessible': True…}, categories: [Mexcian, Restaurants], city: 'North Las Vegas' …}
- user: {id: '-Da_9va6gVvSzItIJftuZA', compliments: {'note': 1, 'funny': 1}, …}
- review: {id: '-0Bow-oUs1Ymn9kOY5QQpg', business_id: '4vkP8WxtHA0uGkcmX3BUdQ', text: 'Rocco's is probably the best pizza you're goin…', stars: 3 …}

Method

Collaborative Filtering Basic models:

- neighborhood methods
  - user-user similarity

$$p_{u,i} = \bar{r}_u + \frac{\sum_{u' \in N} s(u,u')(r_{u',i} - \bar{r}_{u'})}{\sum_{u' \in N} |s(u,u')|}$$

  - item-item similarity

$$p_{u,i} = \frac{\sum_{j \in S} s(i,j)(r_{u,j} - b_{u,i})}{\sum_{j \in S} |s(i,j)|} + b_{u,i}$$

- latent factor models
  - matrix factorization

$$\min_{q^{\cdot},p^{\cdot}} \sum_{(u,i)\in\kappa} (r_{ui} - q_i^T p_u)^2 + \lambda(\| q_i \|^2 + \| p_u \|^2)$$

These basic models were used to create models using additional features from the dataset

11 different models:

- Model 1: Basic item-item similarity with rating
- Model 2: Model 1 measuring item similarity with business categories
- Model 3: Model 1 measuring item similarity with review bag of words
- Model 4: Model 1 measuring item similarity with bag of words counting words in categories "taste", "service", "price", "positive", and "negative"
- Model 5: Basic user-user similarity with rating
- Model 6: Model 5 measuring user similarity with user compliments
- Model 7: Model 5 measuring user similarity with review bag of words
- Model 8: Model 5 measuring user similarity with bag of words counting words in categories "taste", "service", "price", "positive", and "negative"
- Model 9: Basic matrix factorization
- Model 10: Matrix factorization with missing values filled in with user average rating
- Model 11: Matrix factorization with missing values filled in with item average rating

Hybrid model

- The prediction from the models were used to create the final prediction

$$p_{u,i} = \alpha_1 p_{u,i}^{(1)} + \cdots + \alpha_n p_{u,i}^{(n)}$$

- The weights were determined using lasso regression

$$\min_{w} \frac{1}{2n_{samples}} ||Xw - y||_2^2 + \alpha ||w||_1$$

and ridge regression

$$\min_{w} ||Xw - y||_2{}^2 + \alpha ||w||_2{}^2$$

Cross-Validation

- Cross-validation was used to validate the regularization parameters for matrix factorization and lasso/ridge regression
- For matrix factorizataion, 10-fold cross validation was used to run different stochastic gradient descents to find the best lambda and step size
- For lasso regression, coordinate descent was used to find the best alpha value
- For ridge regression, generalized cross validation was used to find the best alpha value

Result

Error was measured in mean square errors.

Model	1	2	3	4	5	6
Error	1.886	1.604	1.845	1.834	1.747	1.506
Model	7	8	9	10	11	
Error	1.509	1.493	1.345	1.367	1.365	

Coefficients for Lasso regression (Error: 1.253)

Model	1	2	3	4	5	6
Error	0	0	0	0	0	0
Model	7	8	9	10	11	Intercept
Error	0	0	0.382	0	0	2.066

Coefficients for Ridge regression (Error: 1.245)

Model	1	2	3	4	5	6
Error	0.01944	-0.0297	-0.1550	-0.092	0.1303	0.0173
Model	7	8	9	10	11	Intercept
Error	0.0325	0.0486	0.318	0.197	0.036	1.639

For comparison, error for baseline prediction (using user's average to guess any item's rating) was 1.648.

Analysis

- Lasso regression uses only model 9 and the intercept to get the lowest error, which signifies model 9 seems to predict the rating best with a bias.
- Ridge regression uses all models to reach an error slightly lower than that of lasso, but some weights are negative, the significance
- Getting the results using this hybrid recommender system with other datasets would help clarify which hybrid model works better in predicting the user's ratings.