Amazon Product Recommendation System

Recommendation System Project Assessment by Anthony Daniels

03-25-2024

Contents / Agenda

Business Problem and Data Overview

- A new recommendation system is needed to assess our existing data and develop a new recommendation system to recommend products to our customers based on their previous ratings for other products
- The data set is broken down by users identified by their user id, every product purchased by each individual user, product ratings when provided by each individual user, and the timestamp of the rating.
- This data set is especially large and difficult to work with so we are limiting the data by removing the unneeded data of time stamping, and limiting the data by the parameters of users who have given 50 or more ratings an products that have received 5 or more ratings. This will allow us to base our findings on regular customers who have likely encountered good and bad products and aren't skewing the data set by a limited experience. Also, since there could be thousands of products with low product sales by ratings skewed far left or right, this impact will be removed.

***Note**: You can use more than one slide if needed*

Exploratory Data Analysis

- After limiting the data to users who have given 50 plus ratings and products that have received 5 or more ratings, our data set includes 1226 users out of a possible 412,319. This .3% representation of our customers are the most impactful raters regarding the most consistently rated products. These are the products our system will ultimately recommend

- Our customers (user_id's) are countable objects. Our products (product_id's) are countable objects. Our rating is not a countable object, but rather a number between 1 and 5 that can be manipulated with mathematical functions including averaging.

- Once we isolated the data, we can see that 1226 out of the 1226 customers in our remaining data set have provided ratings regarding every product included in the data we are using.

- It was found that out of these 1226 users, there were no missing data points. For each transaction there was a customer_id, a product_id, and a rating. This is to be expected as we limited the data by an 'and' restriction meaning that every data point would need to have a customer_id to count by default. If we were to bring in every product with 50 ratings, we would invite in several thousand customer id's that may have only rated one product.

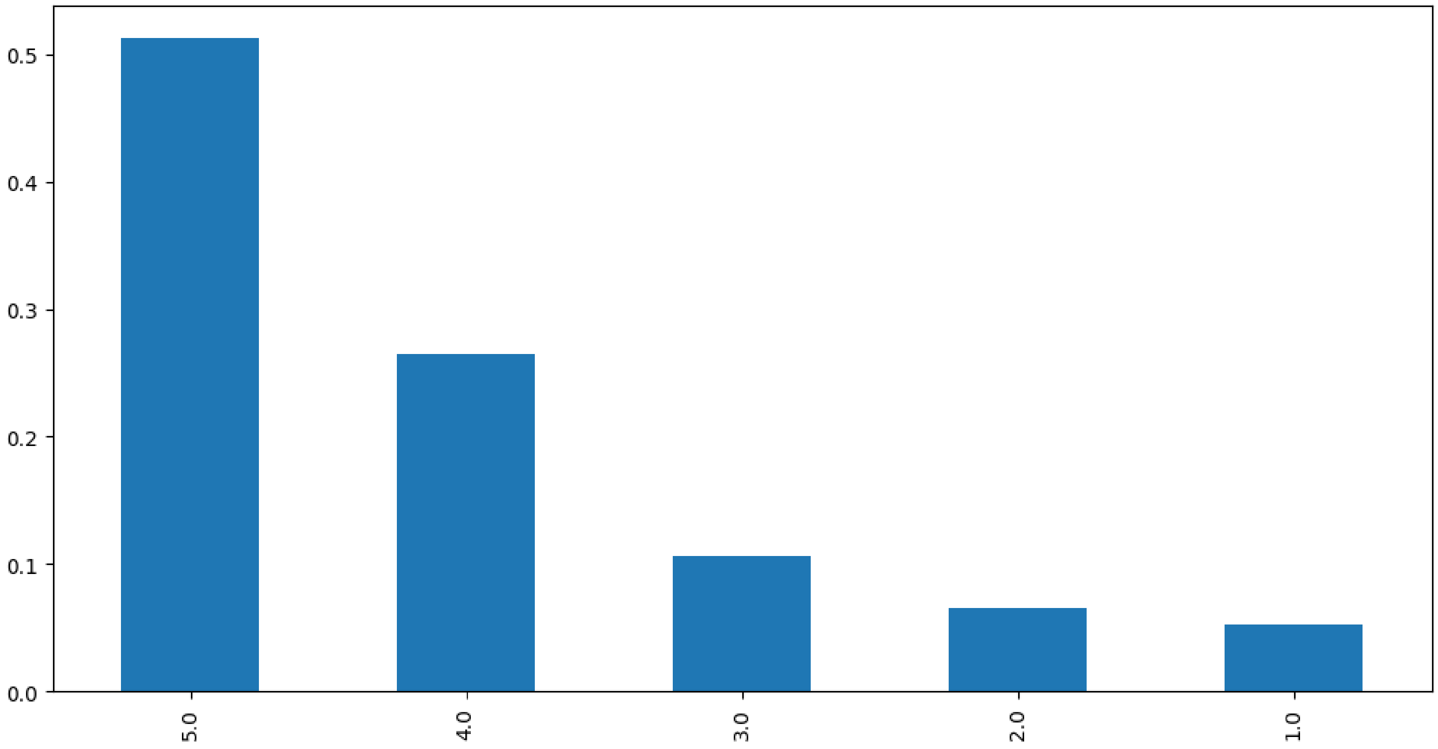
Exploratory Data Analysis Section 2

- Our system will take into account 1020 unique products, but rated by only 18 customers. Similarly as stated before, to review all transactions from customers who have given more than 50 ratings, we would invite in 1000s of products that may have only have that one users rating. As things are, while we are only reflecting in our unique data 18 users, being that each product has 50 or more unique ratings, we know we are taking in input by users outside of the 18.

- We are observing that roughly 78% of our observed ratings data; for products that will be recommended, comes in at a 4 or 5 with an average of 4.12 which can be marketed to consumers to help convince them to participate in surveys more.

- In our data set, with the reduction to 18 unique customers, the maximum amount of ratings a product can receive is 18, not 50. In our case, the maximum number of ratings any product received was 3. Because of this, we will set our minimum number of interactions at 1 and 3

Bar Graph of ratings frequency



Exploratory Data Analysis – Negative Observations

- Restricting our data to only users who have given 50 plus ratings and products that have been rated 5 or more times, opens us up to two biases.

    - Bias 1 is that the customer sample we are using tend to give higher ratings because they are biased to our platform and the service we deliver, which could skew their review of the product. Looking at the fact that the 50 ratings given restriction cut the user ids all the way to 1570 customers from approximately 412,000 indicates that these would be considered not only the most regular, but likely the most elite of our customer base.

    - Bias 2 is that by restricting products to those that have received 5 or more ratings, we potentially could have brought in a bias of products that are highly purchased and are so elite that they influence customers to share their excitement hence presenting skewed data for the big picture. While this observation is more of a reach, it was a thought based on the fact that 51% of our observed data had a rating of 5.

Exploratory Data Analysis – Negative Observations (cont.)

- Cont –
    - There are 2 users who have twice as many ratings as any of the other users in our sample group. Between the two users, they have given 448 ratings out of 1020 ratings. We know that there was no missing data and that every user in our sample had rated every product in our sample as well. These two user tendencies would greatly impact our results in a way that the results could be primarily based on their tendencies vs more randomness, so it would be best to program our system in a way that disregards their ratings.
    - Making this change reduced our ratings to 778 with an average of 4.11. Observing this, the change in average rating is only .01 which is insignificant, and the change in standard deviation is minimal as well. Since we are looking at a situation where we have minimum data to train with, we will go ahead and include the 2 users with more ratings than everyone else so that we can have more data points to work with. So we are back to 1020 unique products and 18 unique users which implies 1020 ratings.

Rank Based Model

- We used the popularity approach to build our rank based model. This approach took the products with the top 5 averages based on maximum and minimum interactions.
- In this case, a popularity based model could actually be our best approach. This is because we have a relatively small data set. We may not have enough data to train the other approaches.
- That being said, some negative regarding this approach is that there is no way to tell beyond the top 5 of any product sorted by number of ratings given. Our data points are so tight, that an alternative approach may give us more insight to similarities between the different consumers to create our recommendations vs just giving the top 5 withing a list which at some point, may have ended up being sorted by something other than rating and may limit the versatility of the recommendations.
- This model also had very limited returns when limiting it to mins and maxes of interactions. The minimum interactions of 4 for products with ratings of 5 only returned 3 items meaning that there are only 3 that meat this criteria and is not really a ranking. I had to reduce the maximum interactions to 4 in order to get 5 recommended items returned.
- This model recommended the following products with only 1 recommendation: B00006B9Qg, B0000DK3I4, B0000U1OD2, B00005I9P3, B0000TU7I6
- The top 5 products returned with a minimum of 3 interactions were: B000050AQ7, B00008OE5G, B00008OE6I, B00006RSJ1, B00006J6RN

***Note**: You can use more than one slide if needed*

User-User Similarity-based Model

- For our user-user similarity based model, we are observing an rmse of 1.3590 which isn't great, a precision of 0.807 which indicates which means that of all the products, 80% are relevant, a recall score of 0.833 which means that of all the relevant products 83% of them are recommended, and lastly and F-1 score of 0.82 which indicates that mostly recommended products were relevant and relevant products were reocommended.

- After tuning hyperparameters, rmse reduced to 1.2385, precision reduced to .79 or 79% of the products being relevant, recall increased very slightly to 0.834 or still 83% of the relevant products being recommended and lastly the F_1 score reduced slightly to .812 still indicating that mostly recommended products were relevant and relevant products were recommended

- Since we have so many overlapping data points, I'd look at this as an slight improvement as rmse did reduce and the other parameters stayed the same for the most part.

- There was no change in the prediction of the baseline model as they both predicted 4.16 which as compared to the original actual rating of 5

Item-Item Similarity-based Model

- Our baseline model with default parameters has rmse of 1.2585 and an F-1 Score of 0.816 on the test set. Precision is still approximately 0.80 for both models and a recall of 0.83 o 0.84 for both models. The RMSE of 1.2585 is a pre-tuning improvement. We continue to get an estimated rating of 4.15 as compared to the baseline of 5

- Explain the difference in predictions using both models.

  After parameter optimization, our RMSE improved to 1.2286. Our precision went down to .79 or 79% of all the products being relevant. Our recall fell a little bit but I still approximately .83 or 83% percent of the relevant products are recommended. Our F-1 score did fall a little bit, but is still sitting at around .81.

  So far, I'd say that our user-user based model and our item-item similarity model are performing similarly. I like the improvement in the RMSE for our item-item model and would lean towards that since it is likely that at any given point and time, an individual is looking at products in a specified category, vs alternative categories that someone like them as a whole, may be shopping for at any given point and time. Our estimated rating remains at 4.15.

***Note**: You can use more than one slide if needed*

Matrix Factorization based Model

- Our training set SVD matrix returned an RMSE of 1.154, which is our best as of yet, but our precision has dropped to 0.77, and our recall to 0.71. This indicates that our products being recommended are significantly less relevant and fewer of our recommended products are being recommended. Since in this scenario, we have not used a recommendation based system before, So far, Id shy away from this model, because I'm hoping that better recommendations will not only increase purchases, but improve purchase recommendation leading to a smaller deviation from the actual data (an improved RMSE) over time when using the item-item based model. Estimation of rating remains at 4.15.
- Once optimized, or RMSE got much much worse ending at 2.5650. Precision improved to .79, but recall dropped to .644 with an F-1 score of .711. Estimated average rating dropped slightly but was still in the approximate range of 4.15.

Note*: You can use more than one slide if needed*

Conclusion and Recommendations

- I would definitely not use the SVD Matrix Model. In my opinion, the item-item model is the best option as I believe most people are shopping for items that are related at any given point and time. If someone is cutting their grass and shopping for a lawnmower, if fertilizer, or lawn blades, or even warranties show up as recommended products, those would be much better recommendations vs T-shirts, or boots, because similar users are shopping for those items at a given point and time.

- It is my recommendation to institute the item-item systems recommendation model, allow it to run for a period of time to accumulate a significant amount of new data, and then run a project to see if sales have improved as a result of the model implementation and to determine if RMSE has improved as well.

Note*: You can use more than one slide if needed*