Capstone 2 – Book Recommendation System

I've come across a list of 278k users who have implicitly or explicitly reviewed 271k books. I'd like to build a recommendation system to recommend the right books to the right users. I'll attempt both a collaborative approach and a content based approach (although we do not have a lot of dimensional data about the books/users).

The Customer

My client for this project could be any retailer who sells books or a platform like Good Reads that allows users to follow each other and get recommendations on what book to read next. After this analysis, retailers should be able to offer better products to their customers and increase lifetime value or sales per customer metrics while Good Reads can demand higher advertising revenue by proving they can put publisher's books in front of the right audience under their sponsored content section.

The Data

I'll be using the "Book-Crossing" data set win which more info can be found at http://www2.informatik.uni-freiburg.de/~cziegler/BX/.

> *Improving Recommendation Lists Through Topic Diversification*,
>
> *Cai-Nicolas Ziegler, Sean M. McNee, Joseph A. Konstan, Georg Lausen; Proceedings of the 14th International World Wide Web Conference (WWW '05), May 10-14, 2005, Chiba, Japan.*

Overall there are 1.1M reviews by 278k users across 271k books, most are implicit (not rated or in the case of this data, 0) while some of explicit ratings of 1 – 10. The data also contains other dimensions, such as the user 's physical location, book's publisher, year of publication, and thumbnail to the image.

Cleaning The Data

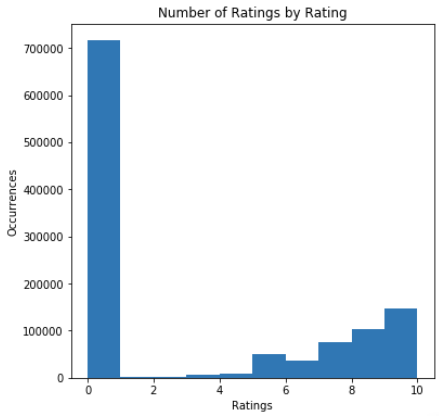
There were some initial issues opening the data. It was stored in a csv with a semicolon separator while book titles contained semicolons in their names. Also, ampersands were converted to "&" which also added to the complexity of opening the file. This was easily overcome by opening the file with the python open function, replacing the "&" occurrences with "&", and using regex to find semi colons located in between the open and close quotes of the file and replacing it with "||". This made the data much easier to split and load into pandas.

Once, in pandas, I found there were about 4600 books without a year associated with it (about 1.8%) and are annotated with a 0. If we end up using year as a feature to predict a top n book, we may have to drop them from the analysis or use the median year of the set.

Many titles and users are from different countries as well. For my purposes of interest, I'll focus on users from the united states only.

Looking at some of the feature data from the books, there are many areas to clean up. Authors are often misspelled and may be listed first name/last name or last name/first name. The same goes for publishers of those books.

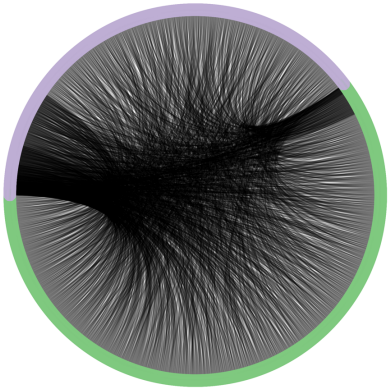
If we are to try and predict based on ratings, we need to take into account the fact that more than 70% of ratings are 0, or implicit. This is an issue with most rating recommendation systems as many users will not rate their product after purchase.



There is something to be said for users who were moved enough (or active in the community) enough to rate their books. We'll see if there is some ratio of explicit rating to implicit ratings that may help increase our model.

To help increase the accuracy of my model, I may want to grab additional information about the books being rated such as genre and a possible summary to add more features about books that users buy/read.

Plot of users (purple) and books (green) and how they are connected



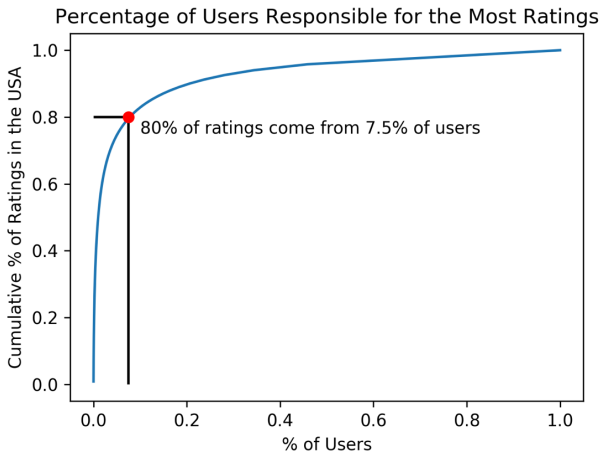
Early Findings

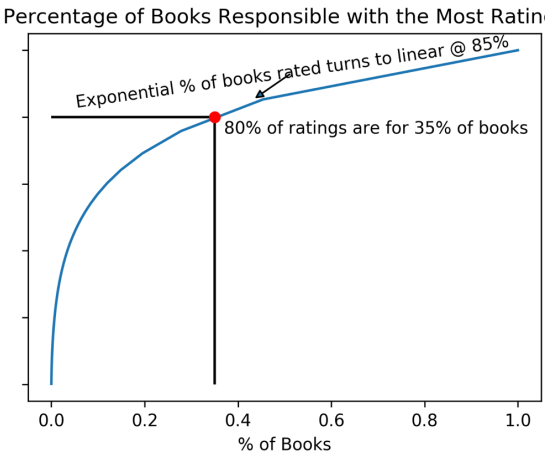
I decided to use a graph data structure to model this data. The data is partitioned (bipartite will be used as the partitioning term) by books and users.

The visualizations that follow are a sampling of the data as there is too much data to show all at once.

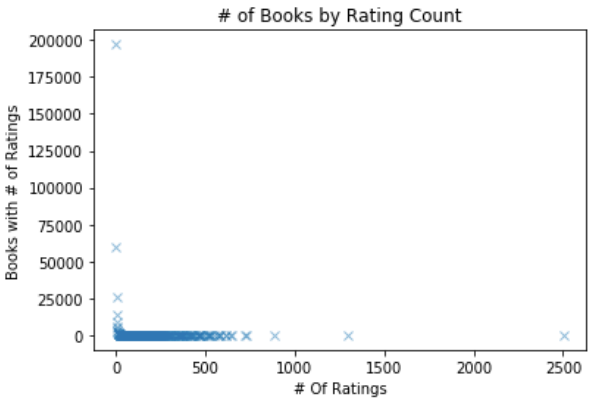
In the Circos plot on the left, we are able to identify users by the color purple and books by the color green. Although you can't see it, all user nodes (Nodes: each individual occurrence of customers or books) flow into the book nodes. Each line represents a rating in our data.

What we see is that there are a large amount of ratings coming from a fairly small amount of customers with most of the reviews flowing into a small amount of books. In fact, about 80% of reviews come from 7.5% of users and 80% of reviews flow into 35% of books. This may indicate that most activity is generated by a few key users and we may want to focus our models on those users who are active on this service.

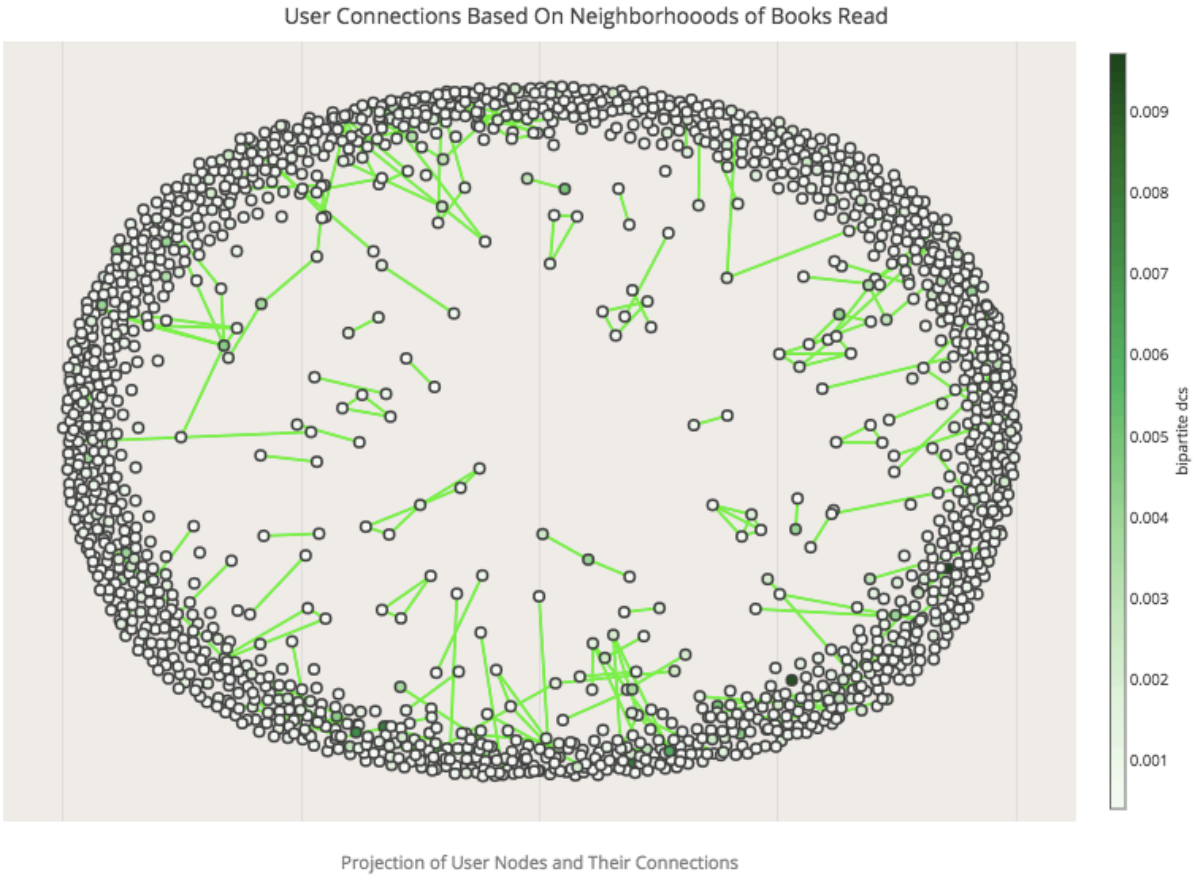




Looking at the number of books by ratings is important as well. Over 57% of books have one rating while over 75% of books have less than 3 ratings. As we're looking at using a collaborative recommendation system, these books do not add a lot of value, especially those with 1 rating. These books may never come up in a top nth predication and could slow the algorithm down. We may want to think about filtering these out.

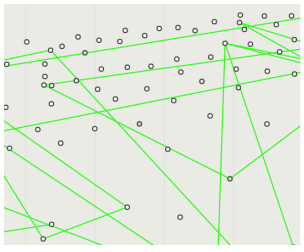


Next, I took a projected graph by bipartite for both users and books.



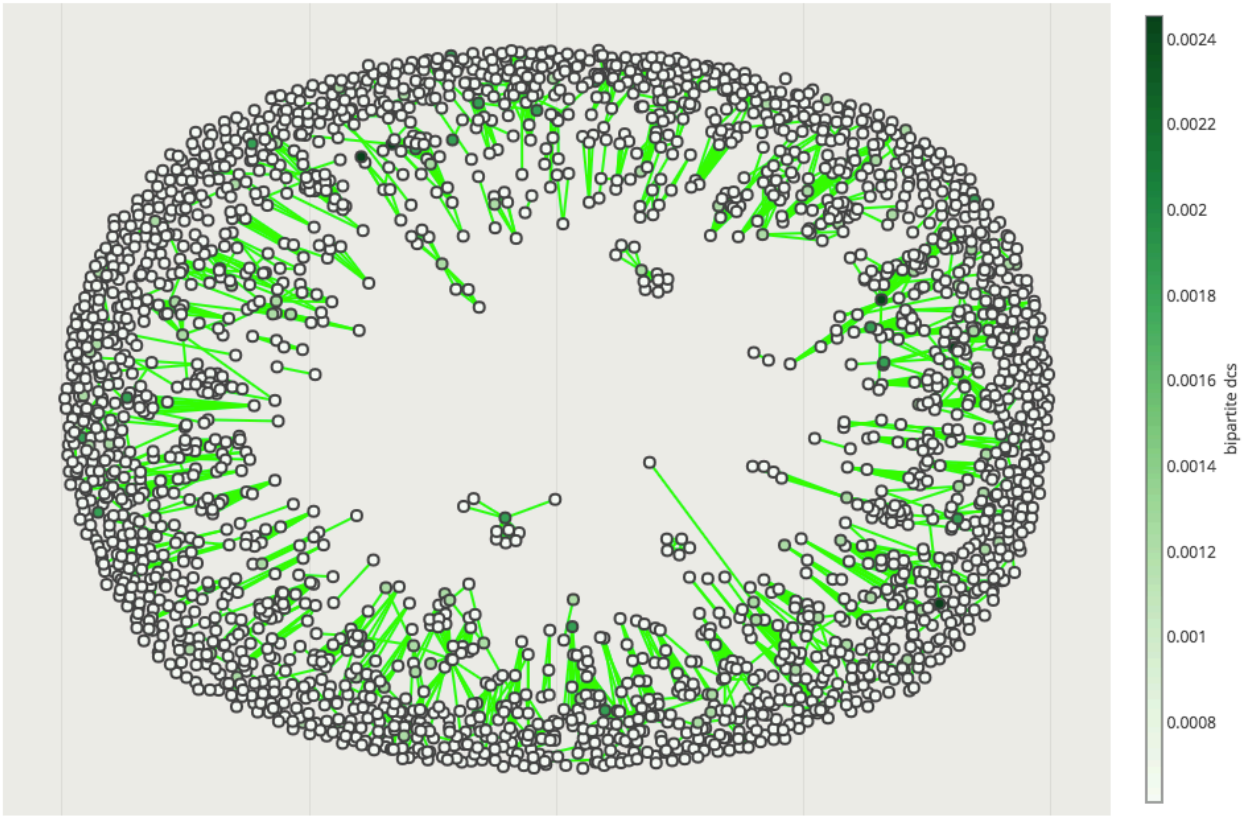
Here we see user nodes represented by the circles and colored based on their bipartite degree centrality, or by how many books the user rated over the total number of books. The nodes are connected by lines based on whether or not they have a rating for the same book whether it's implicit or explicit.

This visualization was a bit surprising to me as I expected to see the higher degree centrality nodes to be more connected to other nodes. In fact, there is practically no correlation with a Pearson coefficient of .15. When you zoom in to the outer regions of the plot, you can see mostly no users being connected.



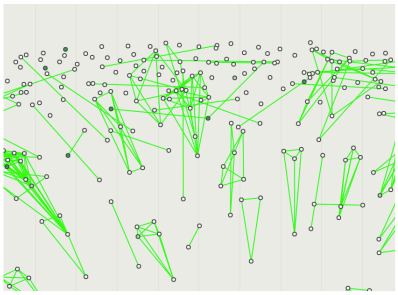
Looking at book nodes that are connected, we see a slightly different story. The connections are based on two books are read by the same user.

Book Connections Based On Neighborhoods of Readers



Projection of Book Nodes and Their Connections

There are more connections here because a connection between books is made when any user reads two books. This does not rely on users having books in common to make connections like the prior visualization.



When zoomed in, we still see more connections albeit still a high count of nodes without connections.

Although there are more connections, there is still no correlation between degree centrality and number of connections as there are still mostly 0 connection values. We find higher correlations (although still low) with the betweenness centrality score of the nodes and the number of connections being .25 and a correlation of .14 between the number of connections between books and their betweenness centrality.

I'm going to start with a collaborative filter recommendation system and move into a content based recommendation system as well. My goal will be to maximize the % of books that show

up in a top 10 recommendations filter. As a final deliverable set, my code, a paper in pdf form, and a slide deck will be published to my github, in this directory.