Goodreads Recommendations

Eitan Angel

July 24, 2019

Contents

1 Introduction **3**

1.1 Problem: Make Book Recommendations for Goodreads Users . . . 3

1.2 Data: Goodbooks-10k . . . 3

1.3 Approach: Collaborative Filtering via Matrix Factorization . . . 3

2 Exploratory Data Analysis **3**

2.1 Books . . . 3

2.2 Ratings . . . 7

2.3 Tags . . . 8

3 Models **9**

3.1 Ratings Matrix . . . 9

3.2 Baseline Model . . . 11

3.3 Factorizations . . . 12

3.3.1 $k = 1$ . . . 12

3.3.2 $k > 1$ . . . 13

3.4 Hyperparameters . . . 17

4 Topic Extraction & Recommendations **18**

List of Figures

1 Average Rating by Ratings Count . . . 4

2 Average Rating by Reviews Count . . . 4

3 Average Rating by Publication Year . . . 5

4 Distribution of User Ratings . . . 7

5 Distribution of Ratings by Book . . . 8

6 Distribution of Ratings by User . . . 8

7 Ratings Submatrix . . . 9

8 Ratings Submatrix (Center) . . . 10

9 Baseline matrix . . . 11

10 NMF-W-1 . . . 12

11 NMF-H-1 . . . 12

12 NMF-1 . . . 13

13 NMF-10-Left . . . 13

14 NMF-H-10 . . . 14
15 NMF-10-Left-Close . . . 14
16 NMF-10-Left-Center-Close . . . 14
17 NMF-50-Left-Close . . . 15
18 NMF-50-Left-Center-Close . . . 15
19 NMF-H-50 . . . 15
20 NMF-250-Left-Close . . . 16
21 NMF-250-Left-center-Close . . . 16
22 RMSE Comparison . . . 17
23 $L1$- and $L2$-regularization . . . 18

List of Tables

1 Most Rated Books . . . 5
2 Most Highly-Rated Books . . . 6
3 Oldest Books . . . 6
4 Greatest Ratings Ratio . . . 6
5 Least Ratings Ratio . . . 7
6 `ratings.csv` and `to_read.csv` . . . 7
7 User Profile . . . 18
8 `book_tags.csv` and `tags.csv` . . . 19
9 Stephen King ($k = 10$) . . . 19
10 Top Tags ($k = 10$) . . . 20
11 Sci-Fi ($k = 25$) . . . 21
13 User Top Ratings . . . 23
14 User Recommendations . . . 24

1 Introduction

1.1 Problem: Make Book Recommendations for Goodreads Users

Goodreads is a social site for readers and for book recommendations. In this project we make recommendations to existing users of books they would most enjoy which they have not yet rated. To do so, we use a collaborative filtering filtering approach and compare the error in our recommendations to the error of some baseline models. Once we have made this model it is not so difficult to provide recommendations to new users who are willing to rate a few books.

1.2 Data: Goodbooks-10k

This is a dataset scraped from Goodreads of the 10,000 most popular books (by number of ratings). It contains book ratings by over 50,000 users, as well as user-created tags, including books tagged "to-read" and considerable data on the books themselves in both a `.csv` file and in an archive of `.xml` files. The basic model will only consider the explicit book ratings although a next step is to find implicit relationships, say among tags and users or books.

1.3 Approach: Collaborative Filtering via Matrix Factorization

We will use a Funk SVD-like collaborative-filtering approach. First we create a user-book matrix of ratings V (sparsity $\approx$ 99%). Following that, we can use Non-negative Matrix Factorization (NMF) to find matrices W and H which decompose V as $V \approx WH$ by minimizing a root-mean-square error (RMSE) between V and WH.

Consider W to be matrix of latent user features and H to be a matrix of latent book features. By matrix completion, we mean to consider the matrix $A = WH$ as "filling in" those ratings which are blank in V. To make recommendations for a user, return the top-N values in the row of A corresponding to that user (which they have not already rated). We can compare the RMSE matrix factorization techniques to various simpler baseline models.

While we only consider the explicit information of the matrix of user ratings in our model, there are many clear avenues for improvement. A slightly complex model which takes into account user co-likes and co-dislikes is discussed in [TLLL18]. The general idea of modifying matrix completion algorithms to account for implicit information (e.g. tags) began with Netflix's SVD++. Matrix completion techniques are surveyed in [RYL+18].

2 Exploratory Data Analysis

While the dataset has considerable features and metadata on books and tags, we will focus on ratings. The three relevant files are `books.csv`, `ratings.csv`, and `to_read.csv`.

2.1 Books

The file `books.csv` has a row for each of the 10,000 most rated books on Goodreads and the following 23 columns: `book_id`, `goodreads_book_id`, `best_book_id`, `work_id`, `books_count`, `isbn`, `isbn13`, `authors`, `original_publication_year`, `original_title`, `title`, `language_code`, `average_rating`, `ratings_count`, `work_ratings_count`, `work_text_reviews_count`, `ratings_1`, `ratings_2`, `ratings_3`, `ratings_4`, `ratings_5`, `image_url`, `small_image_url` .

We will inspect whether `average_rating` is influenced by other `books.csv` features, as well as some of the top-rated books, oldest books, most- and least-reviewed books

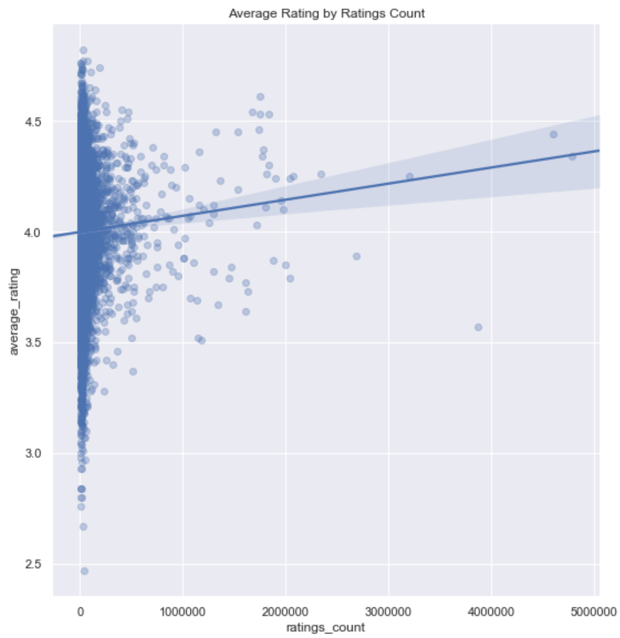

Figure 1: There is some effect of `ratings_count` on `average_rating` – more popular books are better rated.

Figure 2: The number of reviews does not have a significant effect on `average_rating`.

authors	title	avg_rating	ratings
Suzanne Collins	The Hunger Games (The ...	4.34	4942365
J.K. Rowling, Mary GrandPré	Harry Potter and the Sorcerer's...	4.44	4800065
Stephenie Meyer	Twilight (Twilight, #1)	3.57	3916824
Harper Lee	To Kill a Mockingbird	4.25	3340896
F. Scott Fitzgerald	The Great Gatsby	3.89	2773745
John Green	The Fault in Our Stars	4.26	2478609
Veronica Roth	Divergent (Divergent, #1)	4.24	2216814
J.R.R. Tolkien	The Hobbit	4.25	2196809
Jane Austen	Pride and Prejudice	4.24	2191465
J.D. Salinger	The Catcher in the Rye	3.79	2120637

Table 1: The most popular books on Goodreads.

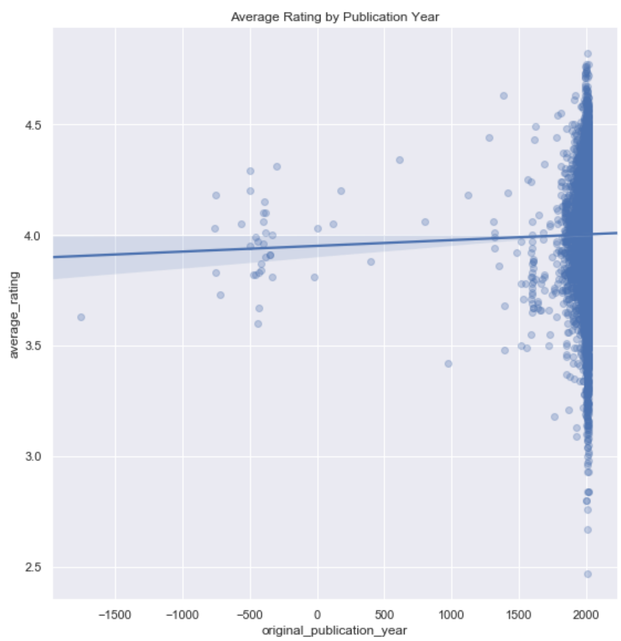

Figure 3: The effect of `original_publication_year` on `average_rating` is not significant. Negative values are books published 1 BCE or earlier.

authors	title	average_rating
Bill Watterson	The Complete Calvin and Hobbes	4.82
J.K. Rowling, Mary GrandPré	Harry Potter Boxed Set, Books 1-5	4.77
Brandon Sanderson	Words of Radiance (The Stormlight ...	4.77
Francine Rivers	Mark of the Lion Trilogy	4.76
Anonymous ...	ESV Study Bible	4.76
Bill Watterson	It's a Magical World: A Calvin and ...	4.75
Bill Watterson	There's Treasure Everywhere: A Calvin ...	4.74
J.K. Rowling	Harry Potter Boxset (Harry Potter, #1-7)	4.74
J.K. Rowling	Harry Potter Collection (Harry Potter, #1-6)	4.73
Bill Watterson	The Indispensable Calvin and Hobbes	4.73

Table 2: Calvin & Hobbes and Harry Potter dominate the average ratings.

authors	year	title
Anonymous...	-1750.0	The Epic of Gilgamesh
Homer, Robert Fagles ...	-762.0	The Iliad/The Odyssey
Homer, Robert Fagles	-750.0	The Iliad
Anonymous ...	-750.0	The I Ching or Book of Changes
Homer, Robert Fagles ...	-720.0	The Odyssey
Aesop, Laura Harris ...	-560.0	Aesop's Fables
Anonymous, Juan Mascaró	-500.0	The Upanishads: Translations from the Sanskrit
Sun Tzu, Thomas Cleary	-500.0	The Art of War
Anonymous ...	-500.0	The Dhammapada
Confucius, D.C. Lau	-476.0	The Analects

Table 3: The oldest books in the dataset.

authors	title	avg	count	ratio
Cynthia Hand, Brodi Ashton, ...	My Lady Jane (The Lady ...	4.12	12794	0.274
Amie Kaufman, Jay Kristoff, ...	Gemina (The Illuminae ...	4.56	10960	0.265
Amie Kaufman, Jay Kristoff	Illuminae (The Illuminae ...	4.32	44500	0.264
Angie Thomas	The Hate U Give	4.62	32610	0.236
Stephanie Garber	Caraval	3.97	30975	0.233
Marissa Meyer	Heartless	4.06	33348	0.233
Sarah Pinborough	Behind Her Eyes	3.77	17944	0.231
Julianne Donaldson	Edenbrooke (Edenbrooke ...	4.34	28536	0.229
Pam Muñoz Ryan	Echo	4.36	14864	0.225
Victoria Schwab	This Savage Song (Monsters ...	4.14	17210	0.225

Table 4: The ratings ratio is `work_text_reviews_count` divided by `work_ratings_count`. The majority of the greatest ratings ratio books are romance novels.

authors	title	avg	count	ratio
Cynthia J. McGean	Henry & Ramona	4.14	11106	0.000270
John D. Rateliff, J.R.R. Tolkien	The History of the Hobbit, Part One...	3.81	108399	0.000424
Frank Miller	Sin City: Una Dura Despedida ...	4.21	9115	0.000439
Janet Evanovich	Janet Evanovich Three and Four	4.34	63691	0.000612
Dean Koontz, Leigh Nichols	Cold Fire / Hideaway / The Key to ...	4.16	17581	0.000626
Mark Cotta Vaz	The Twilight Saga Breaking Dawn ...	4.30	188136	0.000712
Richard Lancelyn Green, ...	The Further Adventures of Sherlock ...	4.40	36863	0.000976
Amazon	Kindle Paperwhite User's Guide	3.72	15002	0.001037
John Williams	Harry Potter and the Chamber of ...	4.61	29409	0.001054
Jenö Barcsay	Anatomy for the Artist	3.97	21640	0.001107

Table 5: Books with the least ratings ratio.

Table 6: `ratings.csv` and `to_read.csv`

user_id	book_id	rating
1	258	5
2	4081	4
2	260	5
2	9296	5
2	2318	3

(a) `ratings.csv` consists of 5,976,479 entries, 53,424 users, and 10,000 books.

user_id	book_id
9	8
15	398
15	275
37	7173
34	380

(b) `to_read.csv` consists of 912,705 entries, 48,871 unique `user_ids`, and 9,986 unique `book_ids`.

2.2 Ratings

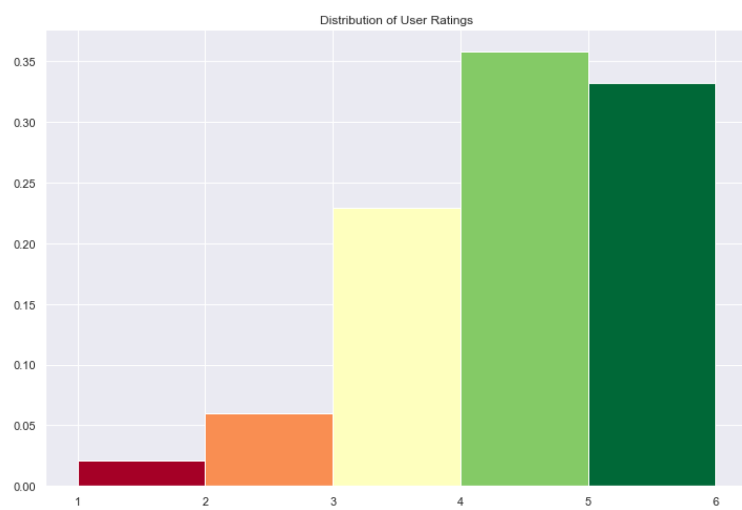

Figure 4: Ratings of 4 or 5 are by far most common.

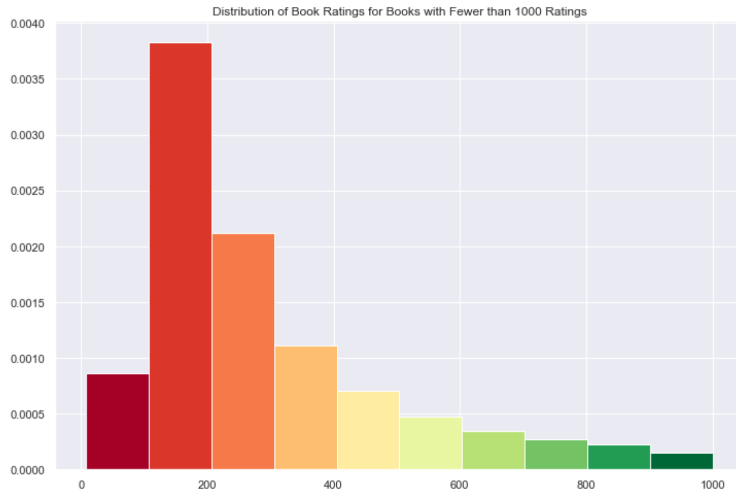

Figure 5: The distribution of ratings by book in `ratings.csv` is left skew. The range is 8–22806 though the interquartile range is 155–503. Since the tail is long we plot the distribution for books with fewer than 1000 ratings.

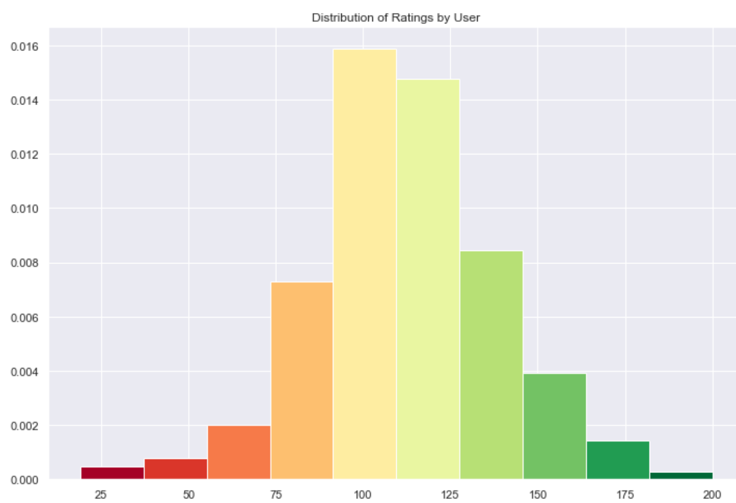

Figure 6: The range of reviews by user is 19–200.

2.3 Tags

We use the user-generated book tags to assist in interpretation of the model. `book_tags.csv` has the top 100 user-generated tags for each book along with the tag counts by book. Most users tag at least one book to-read and almost all books are tagged to-read by some user. We can optionally let users decide against recommendations of books tagged to-read.

3 Models

3.1 Ratings Matrix

First collect the user-book ratings into a matrix V with rows indexed by the ordered set of users U, ordered by `user_id`, and columns indexed by the ordered set of books B, ordered by `book_id`. As the set of ratings R are integers $1-5$, we consider no rating to be a 0 in this representation. Given the matrix V, a baseline model we consider is the mean book rating, that is, the mean along columns, as a recommendation value; these recommendations are identical across users.

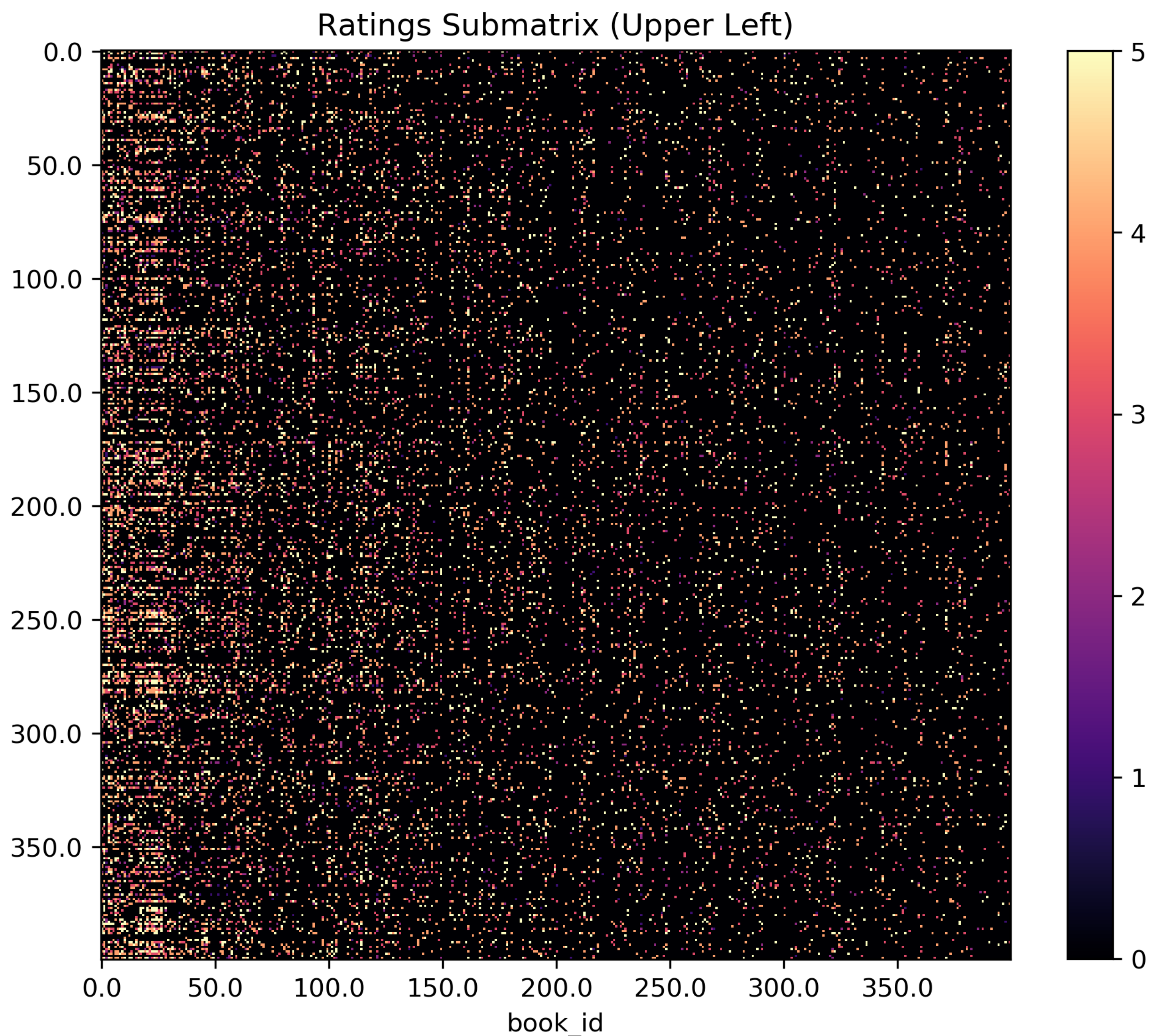

Figure 7: The first 400 rows and columns of the ratings matrix. $\|V\|_F = 9884.39$.

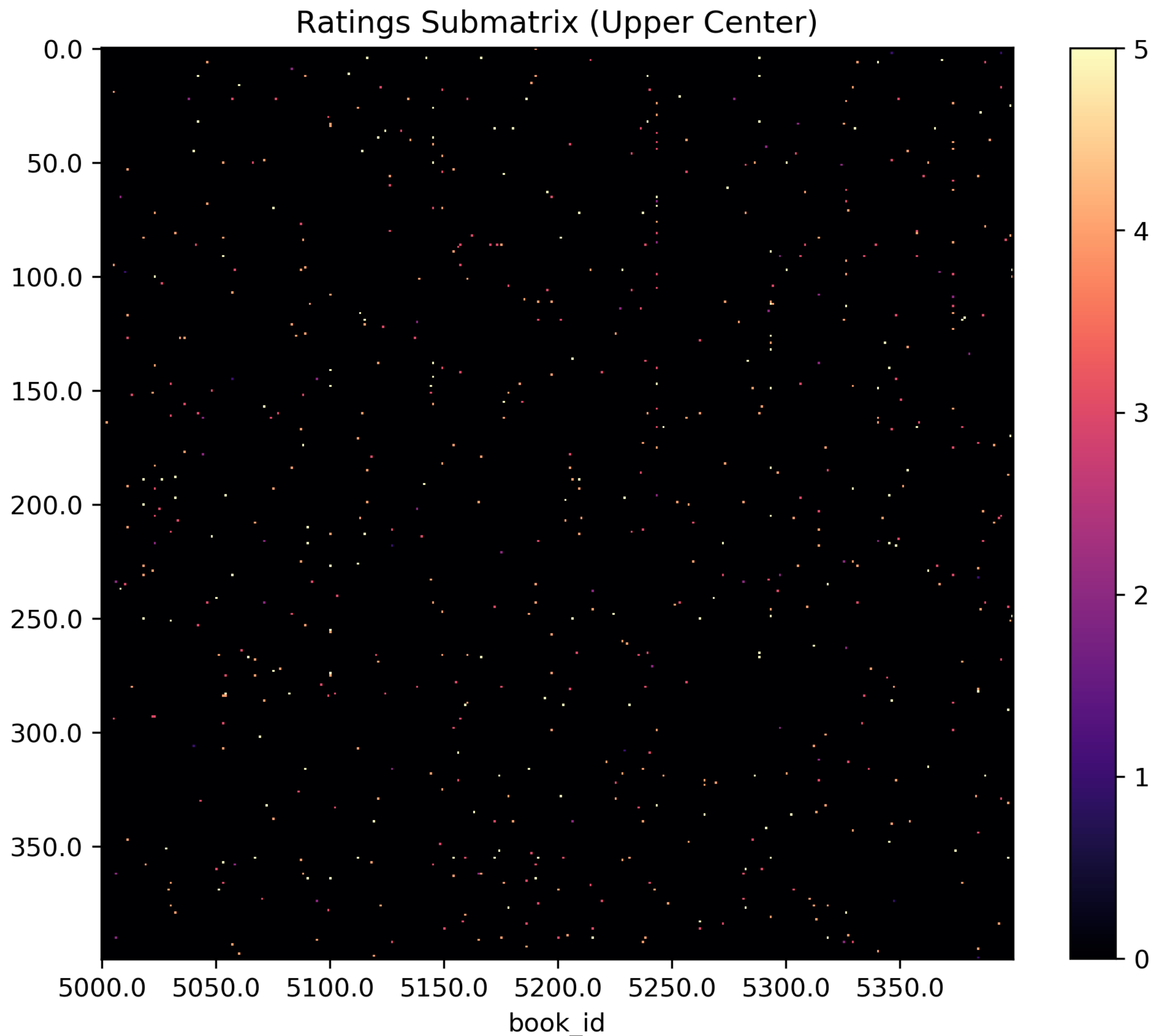

Figure 8: The upper center of the ratings matrix. `book_id` is ordered by overall ratings. The density of the matrix is 1.12%.

Denote the number of users and books by $n_u = |U|$ and $n_b = |B|$ respectively. The models we construct are matrix factorizations of $V \in \mathsf{M}_{n_u \times n_b}(R)$, where R is the set of rating values. A choice of the number of latent factors, k, as well as hyperparameter choices, determine a *matrix factorization model*, which is a factorization of V into a matrix $W \in \mathsf{M}_{n_u \times k}(\mathbb{R}_{\geq 0})$ and a matrix $H \in \mathsf{M}_{k \times n_b}(\mathbb{R}_{\geq 0})$ such that $V \approx WH$.

To be more explicit, we represent $V \in \mathsf{M}_{n_u \times n_b}(\mathbb{R}_{\geq 0})$ and use the non-negative matrix factorization implementation of scikit-learn, sklearn.decomposition.NMF, to return two matrices W and H minimizing the loss function

$$\mathcal{L} = \frac{1}{2}\|V - WH\|_F^2, \tag{1}$$

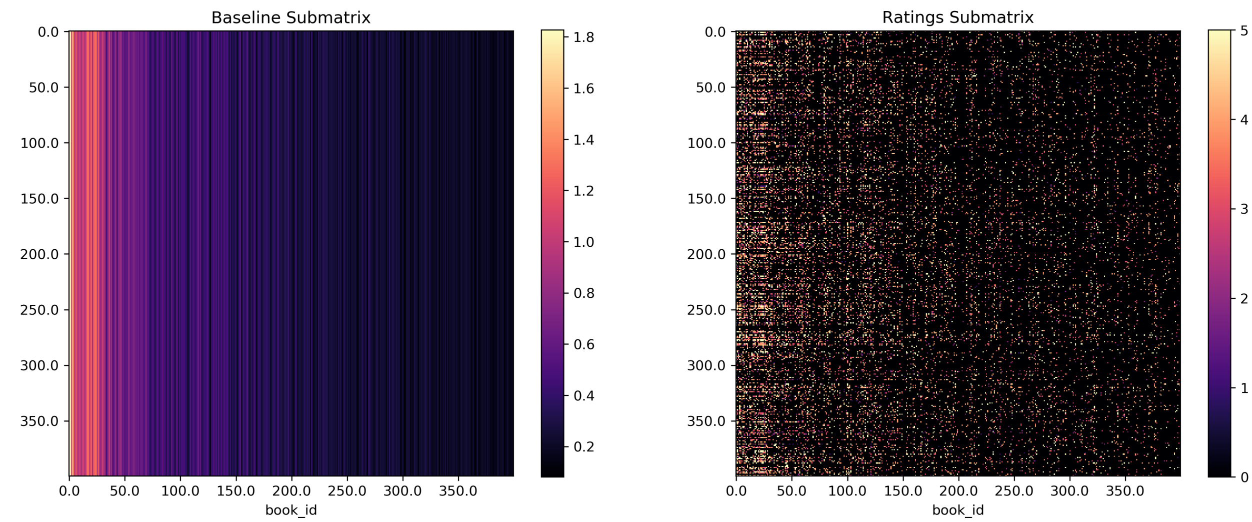

Figure 9: The mean book rating along each column including no ratings. The RMSE is 9584.66.

where $\|\bullet\|_F$ is the Frobenius norm

$$\|X\|_F = \sqrt{\sum_{i,j} X_{ij}^2},$$

that is, the $L2$-norm. [LS01] describes the algorithms used by scikit-learn for matrix factorization.

In minimizing $\mathcal{L}$, we are minimizing the root-mean-square error (RMSE) between V and WH. In analogy to ElasticNet, we also explore $L1$- and $L2$-regularization in hyperparameters, in which we minimize the loss function

$$\mathcal{L}_{\text{reg}} = \frac{1}{2}\|V - WH\|_F^2 + \lambda_1(\|W\|_1 + \|H\|_1) + \lambda_2 \cdot \frac{1}{2}(\|W\|_F^2 + \|H\|_F^2), \tag{2}$$

where $\|\bullet\|_1$ is the $L1$-norm

$$\|X\|_1 = \sum_{i,j} |X_{ij}|$$

and $\lambda_1, \lambda_2 \in \mathbb{R}_{\geq 0}$ are hyperparameters.

[RYL+18] is a helpful survey of matrix factorization techniques for recommendation which suggests that $L2$-regularization is helpful to prevent overfitting while $L1$-regularization can control density. We examine some effects of regularization in 3.4.

3.2 Baseline Model

As a baseline model, take the mean rating for each book as the value along each column of the ratings matrix V. This yields a vector of length n_b which makes uniform predictions across users. A plot of the baseline model alongside the ratings matrix is in Figure 9.

The RMSE of the baseline model is 9584.66, which is a markedly lower score than $\|V\|_F = 9884.39$. A lower score does mean that the baseline model does in fact approximate V. Scores are lower when we subtract matrices from V which have entries closer to V's entires (in the $L2$ sense).

3.3 Factorizations

We can think of W as a matrix of *user preferences* for book profiles. A row w_u of W is a vector of length k which describes the degree to which each of the k latent factors influences user preferences for books. Similarly, we can think of H as a matrix of *books preferenced* by user profiles; a column h_b of H describes the degree to which each of the k latent factors influences that book's preferences by users. The dot product $a_{ub} = w_u h_b^T$ captures the correlation between user u and and book b; we consider the matrix $A = WH$ to be a "completion" of V.

There are a few advantages and interpretations of matrix factorization.

- While a dense matrix of size $n_u \times n_b$ is a few GB, for low k, H and W are only a few MB.
- Since H and W are (for $k < n_u, n_b$) low-rank matrices relative to the size of V, matrix factorization can be considered a dimensionality reduction technique.
- Matrix factorization has various equivalencies with K-means clustering, as described in [DHS05].
- We interpret the latent factors as clusterings, which is the justification for topic modeling in section 4.
- The latent factors, H and W, can be used as training vectors for other models besides matrix completions.

3.3.1 $k = 1$

Choosing a number of latent factors $k = 1$ acts as a sort of baseline model as well. In this case H is a vector of book ratings aggregated by user – while W gives the component of each user in the H 'direction'. In other words, W describes how 'close' each user's ratings are to the aggregated book ratings. The vector H is highly correlated to the column means of the baseline model, so recommendations from these two models are very similar, but the $k = 1$ model scores better since it also takes into account how much each user's ratings are correlated to the aggregated book ratings. We now have not only a row vector of book rating means but a column vector of user correlations to the aggregated book ratings.

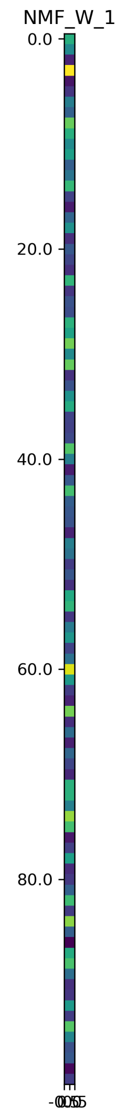

Figure 10

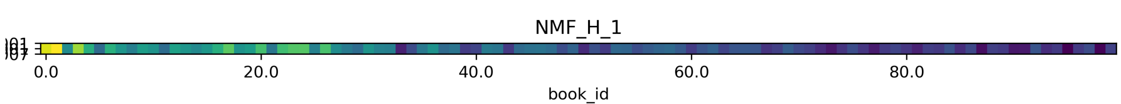

Figure 11: A portion of the book preferenced matrix ($k = 1$).

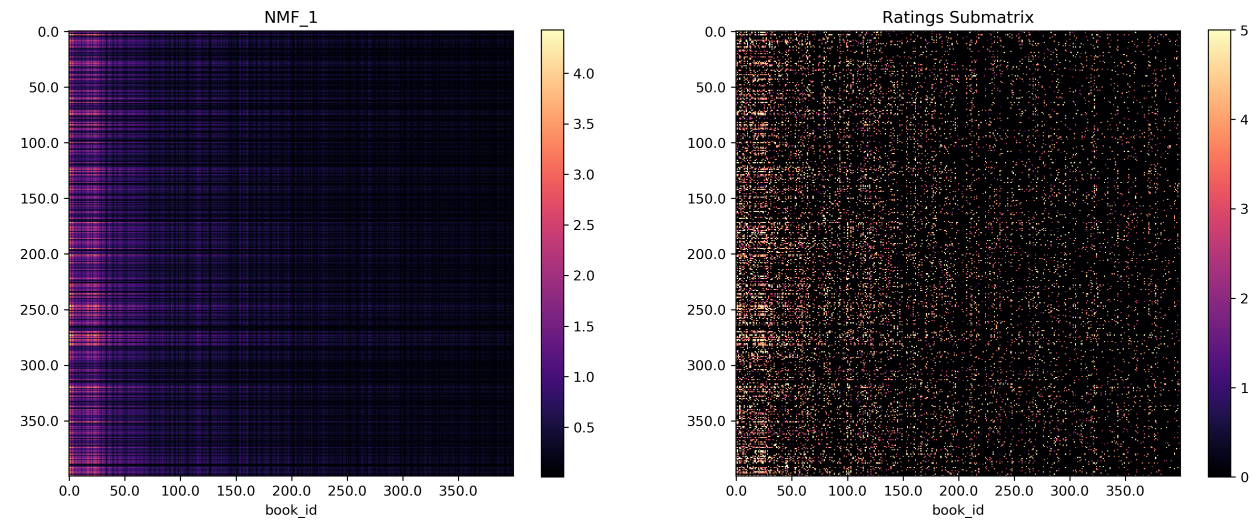

Figure 12: The matrix reconstruction compared aside the ratings matrix ($k = 1$).

3.3.2 $k > 1$

We have trained models for $k \in \{1, 10, 25, 50, 100, 250\}$. Lesser values of k give more interpretable models, as we describe in section 4, whereas greater values of k give more accurate models, as in Figure 20. Upon viewing the matrices, we can also see that the matrix reconstruction becomes finer and makes more 'confident' recommendations for less popular books. For lower values of k, the rows exhibit strong correlations between each other. No regularization is applied to any of the models below, but we discuss and demonstrate the effects in section 3.4.

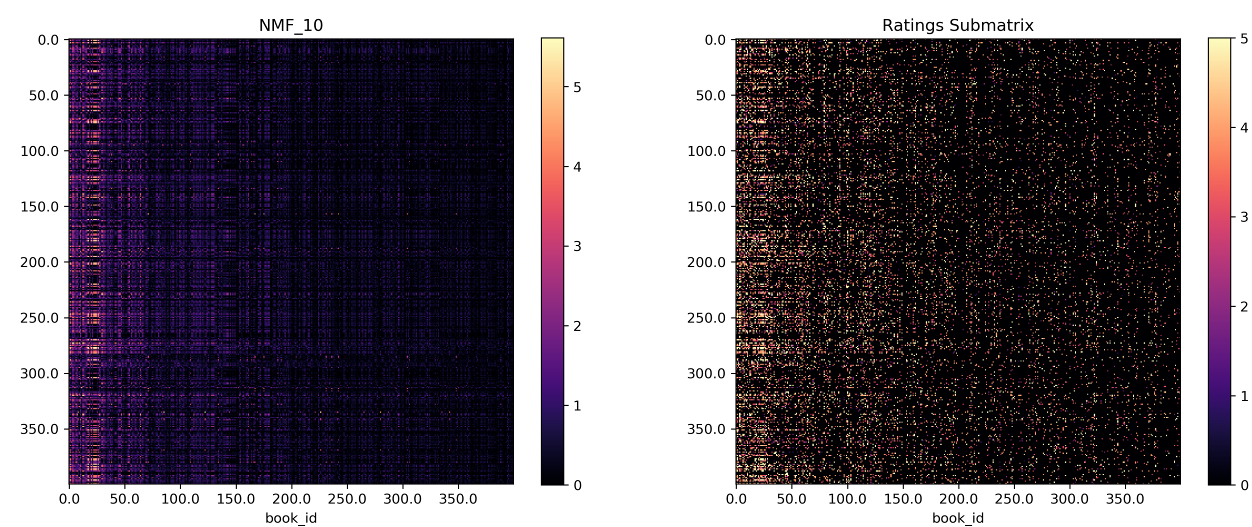

Figure 13: Now the reconstruction clearly captures features of the ratings matrix.

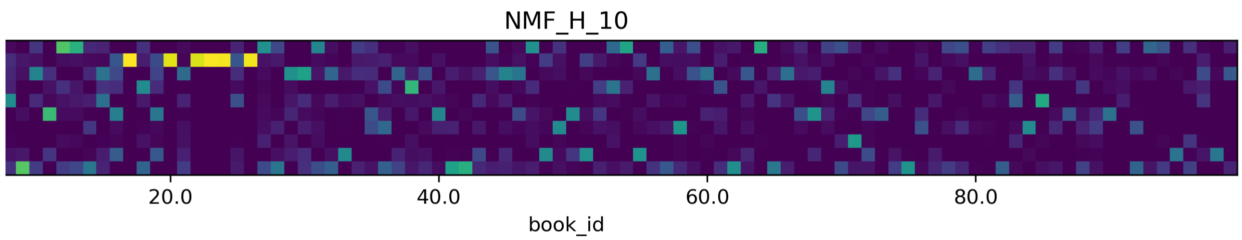

Figure 14: The size of the $k = 10$ model is 5MB.

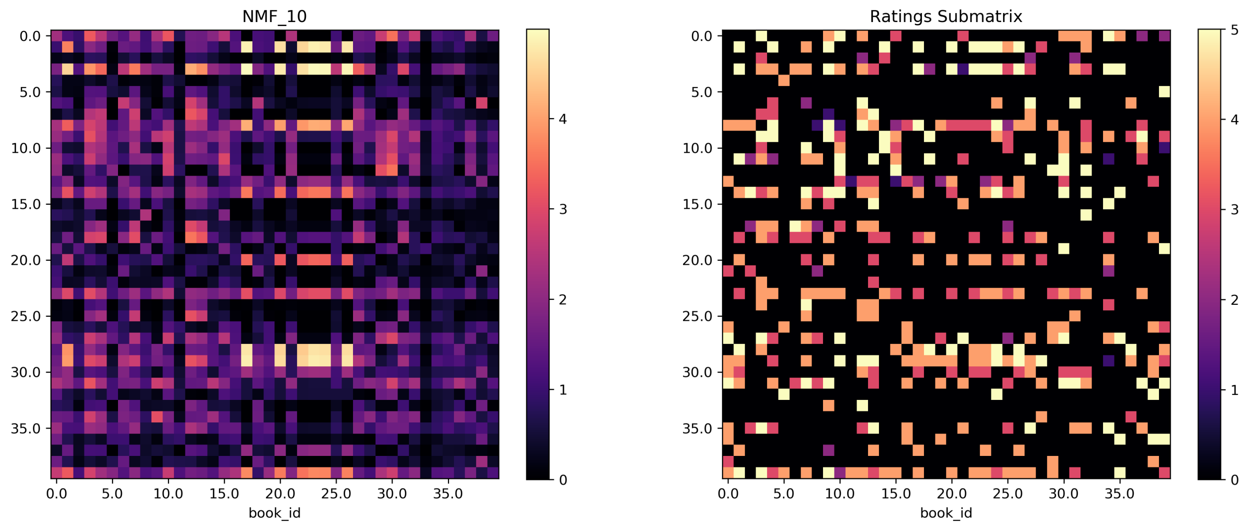

Figure 15: Looking closer, we can see that the reconstruction fills in entries with no rating in the ratings matrix.

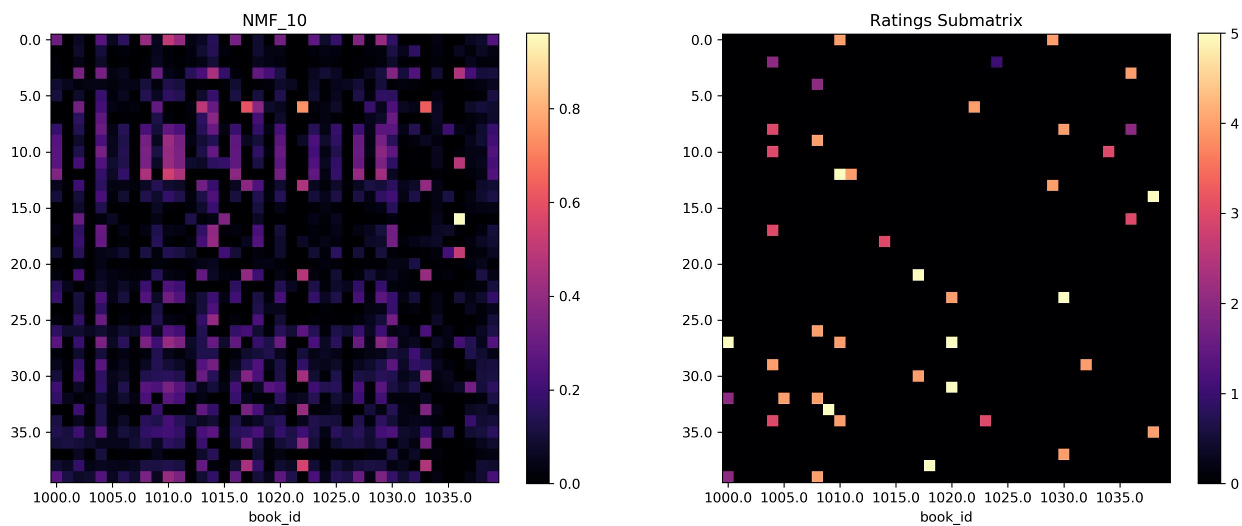

Figure 16: Moving 1000 columns to the right, we see a sparser portion of the matrix.

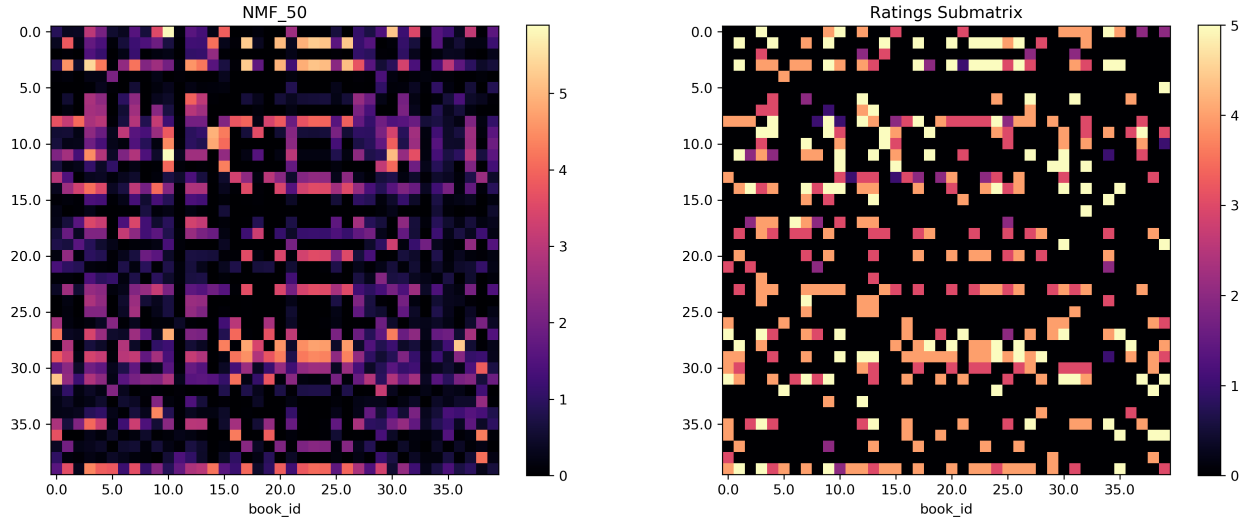

Figure 17: Visually, at $k = 50$, we see the rows are less correlated; each row is a linear combination of 50 vectors rather than 10 as in the $k = 10$ case.

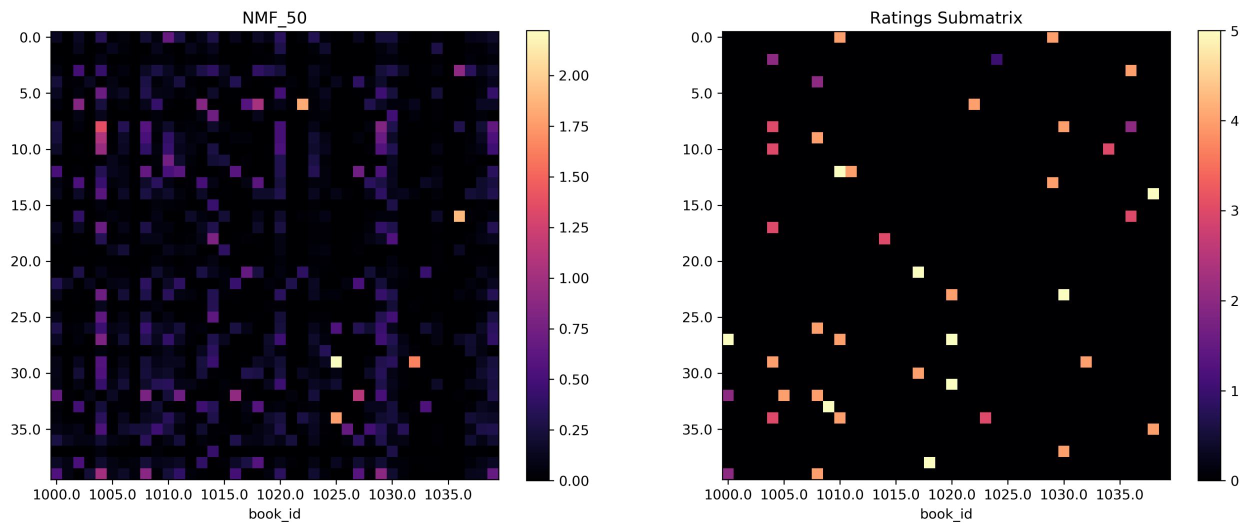

Figure 18: For $k = 50$ the correlations in sparser areas of the matrix reconstruction are stronger than for $k = 10$.

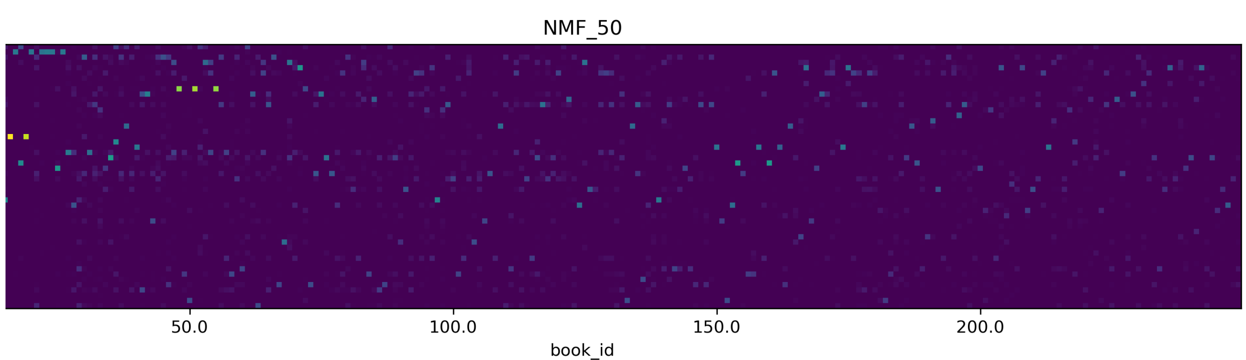

Figure 19: The size of the $k = 50$ model is 25MB. The first few hundred columns of H are shown.

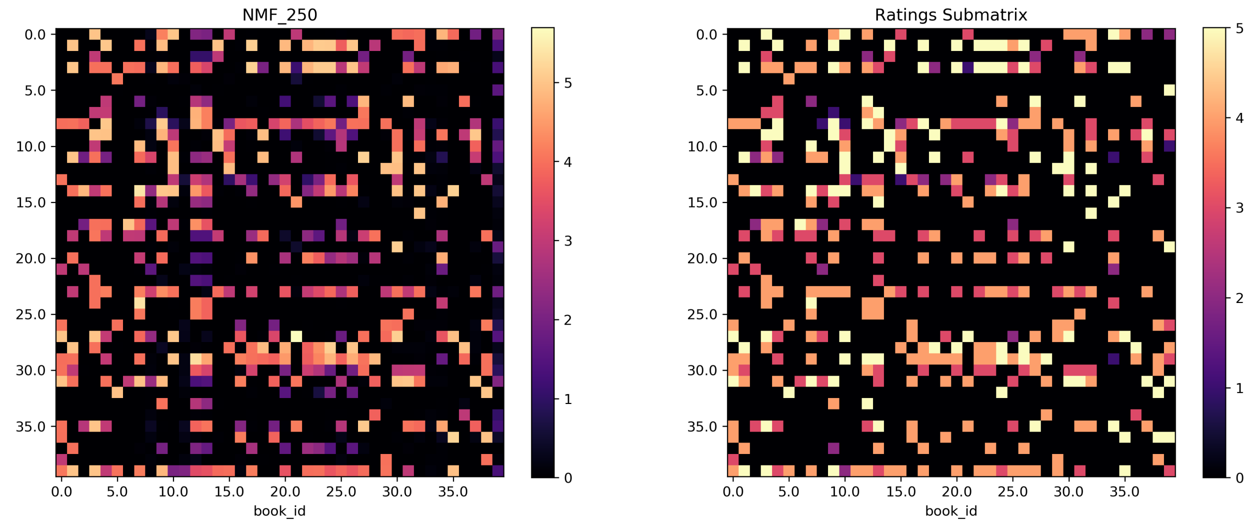

Figure 20: $k = 250$ uses roughly 125MB. While in the upper left 400×400, there are some points such that the reconstruction value is as great as 14, it is notable that none of the values shown is greater than 6.

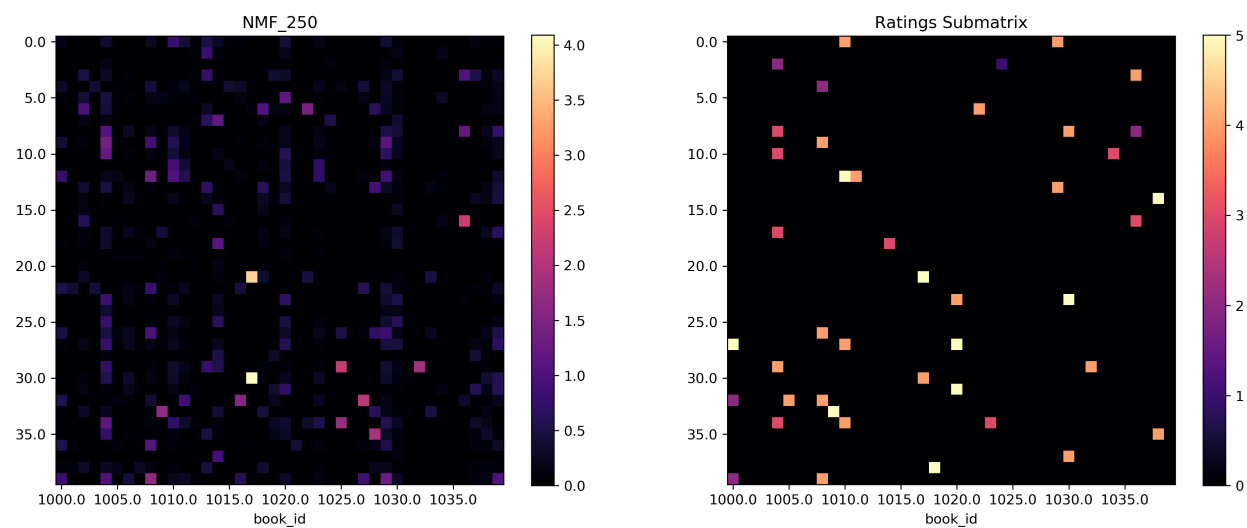

Figure 21: For $k = 250$ the correlations in sparser areas of the matrix reconstruction are considerably greater than in $k = 50$ or $k = 10$.

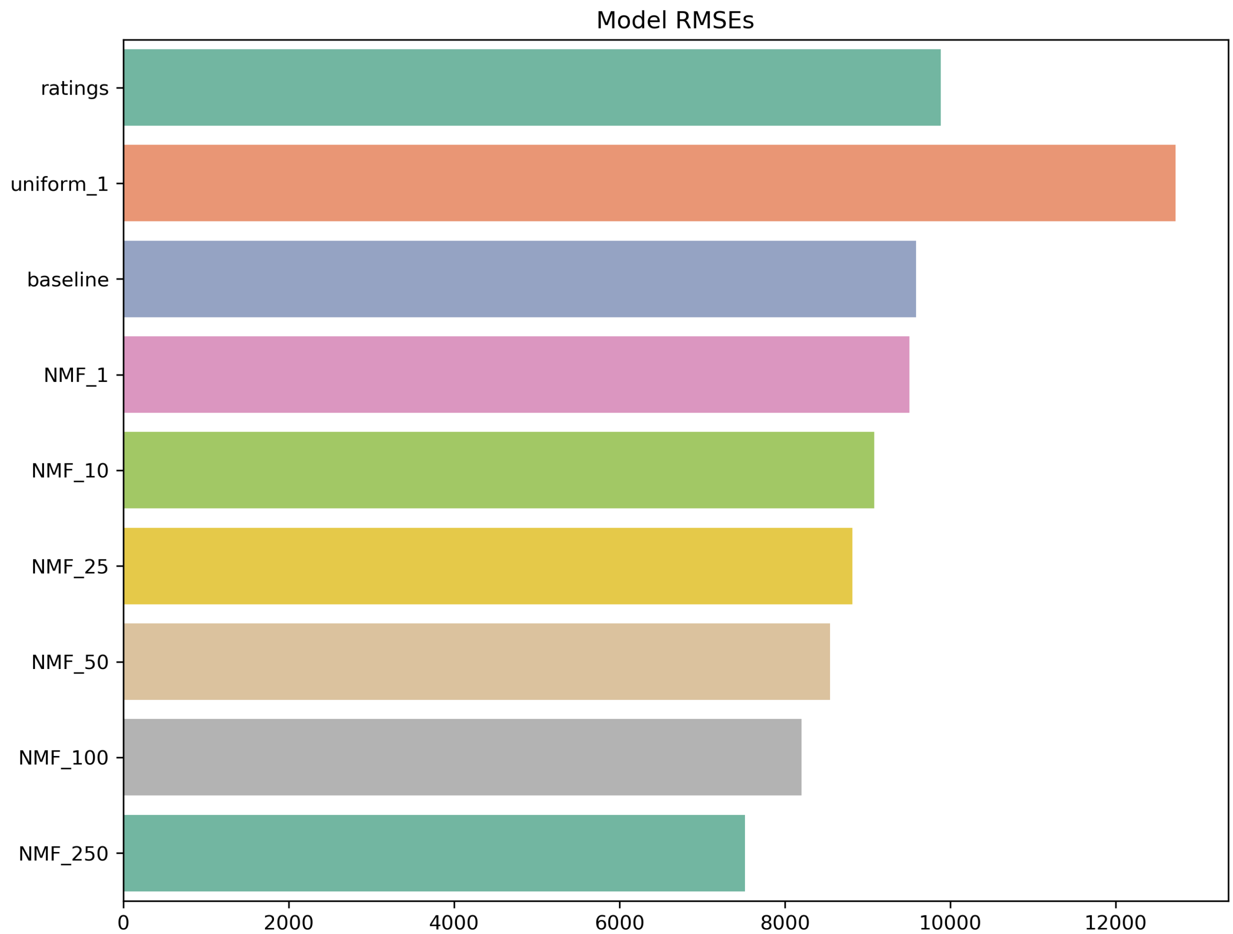

Figure 22: The $k = 250$ model factorization requires roughly 45 minutes to compute on Kaggle's platform; it may be reasonable to consider larger factorizations for recommendations.

3.4 Hyperparameters

The hyperparameters of `sklearn.decomposition.NMF` are similar to those of `sklearn.linear_models.ElasticNet` in that we can tune the magnitude $\alpha := \lambda_1 + \lambda_2$ and the `l1_ratio`, $\rho = \frac{\lambda_1}{\lambda_1+\lambda_2}$, where λ_1 and λ_2 are as in equation (2). Figure 23 shows the effects of $L1$- and $L2$-regularization for $k = 50$. $L2$-regularization does not seem to help RMSE much except for some of the largest values of k we examine, and even then only marginally, so none of the models above include any $L2$-regularization.

$L1$-regularization is useful in controlling sparsity. This lowers RMSE but could still be advantageous as it 'zeros out' entries with very low correlation which would not be a factor in our recommendations anyway. A dense matrix of size $n_u \times n_b$ is a few GB, which is quite tractable, so we do not apply any $L1$-regularization to the models above. In a more realistic situation, we may expect $n_b, n_u \approx 10^6$ rather than $n_b, n_u \approx 10^4$, so that a model with sparse reconstructions may be advantageous at a deployment or engineering phase.

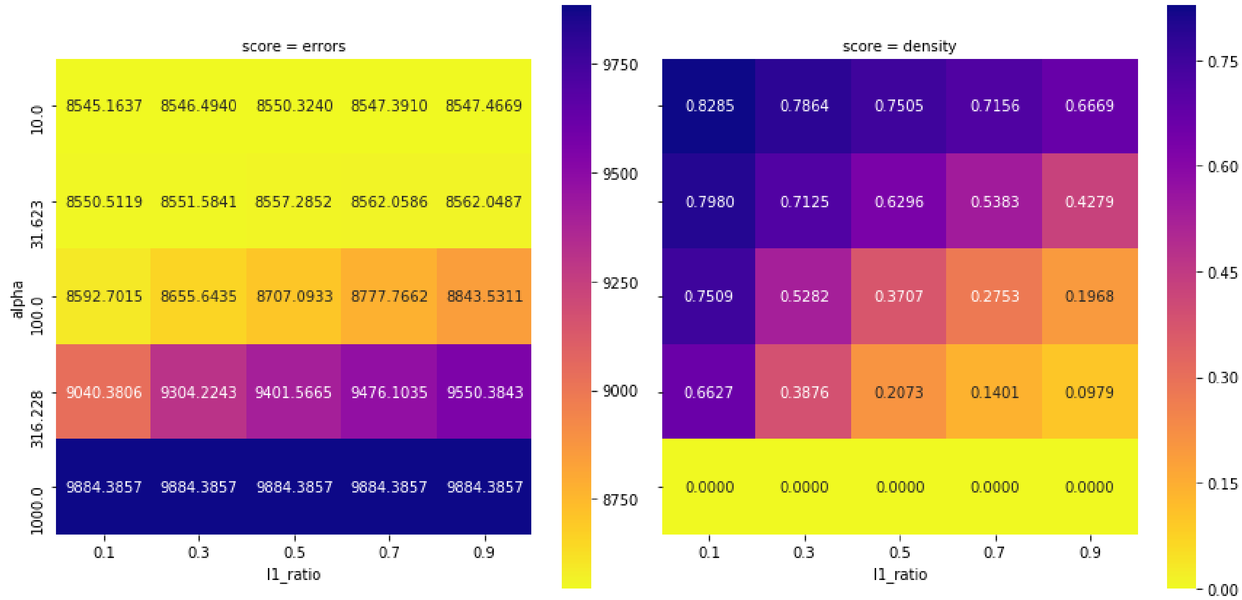

Figure 23: $k = 50$. We can see an accuracy/sparsity trade-off between $L1$- and $L2$-regularization.

4 Topic Extraction & Recommendations

Once we have constructed the model we can use the latent factors to cluster books, make user profiles based on that clustering, and recommend books based on user components in each cluster. The book preferenced matrix H contains a column vector h_b of length k for each book $b \in B$. The components of h_b indicate the correlation between book b and the ℓ^{th} latent factor.

We use two approaches to interpret each latent factor in order to name book topics. The first is to sort the values within each row h^ℓ, for $1 \leq \ell \leq k$, of H in descending order and list the associated top books indexing those values. The top values of h^7 are shown in Table 9.

	9
topic_name	
Harry Potter	0.29
Modern Classics	0.22
Fiction	0.22
Twilight & Fifty Shades	0.12
Austen & Brontës	0.06
Thrillers	0.02
Stephen King	0.00
Children's	0.00
Young Adult	0.00
Fantasy & Sci-Fi	0.00

Table 7: The vector w_9 describes `user_id` 9's preferences for each of the $k = 10$ book topics.

The second is to collect top user-generated tags for each latent factor: The file `book_tags.csv` contains a count of user-generated tags $n_{b,t}$ for each book b for the 100 most used tags t per book. For each latent factor ℓ, $1 \leq \ell \leq k$, use as weights the ℓ^{th} row of H, h^ℓ, which scores how much each book is correlated with latent factor ℓ. Then compute the weighted sum of tag counts $n_{b,t}$ over all tags t associated to all books b. The resulting *tag rank* r_t^ℓ for tag t in latent factor ℓ is defined as

$$r_t^\ell = \sum_b h_b^\ell n_{b,t}.$$

In Table 10, we show the tag ranking for each latent factor for $k = 10$.

For making recommendations, we should use our most complex model ($k = 250$). To make recommendations for a given user u, multiply row u of W, w_u, by H to obtain the completion vector $a_u = w_u \cdot H$, then select the top (unrated) values. This is advantageous in that we do not need to reconstruct the entire completion matrix $A = WH$ in order to create recommendations for a single user.

In addition, we can use the book topics we named in the previous section ($k = 10$) to make a user profile, showing the user's preferences for each category. This is merely the vector w_u, the u^{th} row of W.

Table 8: `book_tags.csv` and `tags.csv`

goodreads_book_id	tag_id	count
1	30574	167697
1	11305	37174
1	11557	34173
1	8717	12986
1	33114	12716

(a) `book_tags.csv`

tag_id	tag_name
0	-
1	–1-
2	–10-
3	–12-
4	–122-

(b) `tags.csv`

authors	title
Stephen King	It
Stephen King, Bernie Wrightson	The Stand
Stephen King	The Shining (The Shining #1)
Stephen King	Misery
Stephen King	Carrie
Stephen King	Pet Sematary
Stephen King	'Salem's Lot
Stephen King	Needful Things
Stephen King	The Gunslinger (The Dark Tower, #1)
Stephen King	The Green Mile
Stephen King	The Dead Zone
Stephen King	Firestarter

Table 9: The top-scoring books in topic 7 ($k = 10$) are all Stephen King novels.

Table 10: The top tags in each topic computed via sum of tags for each book in topic, weighted by component of book in topic.

Modern Classics	Harry Potter	Fiction	Fantasy & Sci-Fi	Thrillers
classics	fantasy	historical-fiction	fantasy	mystery
classic	young-adult	young-adult	science-fiction	thriller
science-fiction	harry-potter	book-club	sci-fi	fantasy
sci-fi	ya	classics	classics	young-adult
literature	series	mystery	young-adult	crime
fantasy	magic	fantasy	series	suspense
school	childrens	contemporary	sci-fi-fantasy	series
novels	re-read	ya	adventure	science-fiction
dystopian	adventure	romance	epic-fantasy	john-grisham
historical-fiction	children	non-fiction	ya	default
dystopia	children-s	dystopia	kindle	mystery-thriller
young-adult	all-time-favorites	kindle	audiobook	sci-fi

Young Adult	Children's	Stephen King	Twilight & Fifty Shades	Austen & Brontës
young-adult	childrens	horror	young-adult	classics
fantasy	classics	stephen-king	fantasy	fantasy
ya	children	fantasy	romance	classic
dystopian	children-s-books	thriller	vampires	young-adult
romance	fantasy	king	ya	romance
series	children-s	science-fiction	paranormal	historical-fiction
dystopia	young-adult	default	series	literature
science-fiction	picture-books	classics	dystopian	school
sci-fi	childhood	sci-fi	dystopia	historical
paranormal	kids	mystery	vampire	novels
adventure	poetry	supernatural	paranormal-romance	clàssics
teen	childrens-books	suspense	urban-fantasy	ya

3 authors	title
Douglas Adams	The Hitchhiker's Guide to the Galaxy (Hitchhik...
Orson Scott Card	Ender's Game (Ender's Saga, #1)
Frank Herbert	Dune (Dune Chronicles #1)
Isaac Asimov	Foundation (Foundation #1)
Ray Bradbury	Fahrenheit 451
George Orwell, Erich Fromm, Celâl Üster	1984
Neil Gaiman	American Gods (American Gods, #1)
Aldous Huxley	Brave New World
Alan Moore, Dave Gibbons, John Higgins	Watchmen
Isaac Asimov	I, Robot (Robot #0.1)
Philip K. Dick, Roger Zelazny	Do Androids Dream of Electric Sheep?
Michael Crichton	Jurassic Park (Jurassic Park, #1)

Table 11: Further clustering leads to further refinements in topics. For example, in $k = 25$ there is a single sci-fi topic while $k = 10$ had a Sci-Fi and Fantasy topic; there are many fantasy topics centered around more specific fantasy series for $k = 25$.

4 title	7 title
The Shining (The Shining #1)	The Drawing of the Three (The Dark Tower, #2)
It	The Waste Lands (The Dark Tower, #3)
Misery	Wizard and Glass (The Dark Tower, #4)
The Stand	Wolves of the Calla (The Dark Tower, #5)
Carrie	The Dark Tower (The Dark Tower, #7)
Pet Sematary	The Gunslinger (The Dark Tower, #1)
'Salem's Lot	Song of Susannah (The Dark Tower, #6)
Needful Things	The Wind Through the Keyhole (The Dark Tower, ...
Cujo	The Eyes of the Dragon
Firestarter	The Talisman (The Talisman, #1)
The Dead Zone	Hearts in Atlantis
Christine	Different Seasons

(a) For $k = 25$ there are two Stephen King topics (thus author columns excluded); but there is a clear distinction between the type of books in each topic.

tag_name	4	tag_name	7
horror	764504.41	fantasy	247250.45
stephen-king	330394.24	stephen-king	102357.72
fantasy	138616.48	horror	98633.96
thriller	100242.94	science-fiction	26623.75
king	72494.46	king	23965.85
classics	67286.23	series	23063.93
science-fiction	55149.26	sci-fi	22892.69
default	54161.39	default	18704.06
mystery	43744.66	western	14983.82
sci-fi	43039.65	dark-tower	14601.85

(b) While Stephen King titles may be familiar, the tag rankings for each topic assist in distinguishing topics especially if we are unfamiliar with the authors and titles.

authors	title	ratings_count	model	rating
Truman Capote	In Cold Blood	381652	5.21	5.00
Jane Austen	Pride and Prejudice	2035490	5.08	5.00
David Sedaris	Me Talk Pretty One Day	495736	5.05	5.00
F. Scott Fitzgerald	The Great Gatsby	2683664	5.05	5.00
Mark Haddon	The Curious Incident of the Dog in the Night-Time	867553	5.02	5.00
George Orwell, Erich Fromm, Celâl Üster	1984	1956832	4.79	5.00
Sylvia Plath	The Bell Jar	401605	4.54	4.00
Stephenie Meyer	Eclipse (Twilight, #3)	1134511	4.41	4.00
Jeffrey Eugenides	Middlesex	488243	4.32	4.00
Stephenie Meyer	Breaking Dawn (Twilight, #4)	1070245	4.27	5.00
Stephenie Meyer	New Moon (Twilight, #2)	1149630	4.22	4.00
George Orwell	Animal Farm	1881700	4.20	4.00
J.K. Rowling, Mary GrandPré	Harry Potter and the Deathly Hallows (Harry Po...	1746574	4.08	5.00
Frank McCourt	Angela's Ashes (Frank McCourt, #1)	392103	4.02	4.00
Gillian Flynn	Gone Girl	512475	4.01	4.00
J.K. Rowling, Mary GrandPré	Harry Potter and the Half-Blood Prince (Harry ...	1678823	4.00	4.00
J.K. Rowling, Mary GrandPré	Harry Potter and the Sorcerer's Stone (Harry P...	4602479	3.99	4.00
Carlos Ruiz Zafón, Lucia Graves	The Shadow of the Wind (The Cemetery of Forgot...	263685	3.99	5.00
William Golding	Lord of the Flies	1605019	3.98	4.00
Suzanne Collins	The Hunger Games (The Hunger Games, #1)	4780653	3.96	4.00

Table 13: The dot product of vector w_9 with H describes `user_id` 9's preferences for books via each of the $k = 10$ book topics.

authors	title	ratings_count	model	to-read
David Sedaris	When You Are Engulfed in Flames	150898	2.72	NaN
Gabriel García Márquez, Edith Grossman	Love in the Time of Cholera	283806	2.30	True
Jane Austen, James Kinsley, Deidre Shauna Lynch	Persuasion	365425	2.21	True
Dave Eggers	A Heartbreaking Work of Staggering Genius	145459	2.02	NaN
Michael Chabon	The Amazing Adventures of Kavalier & Clay	147717	2.01	NaN
Jonathan Safran Foer	Extremely Loud and Incredibly Close	294726	1.94	NaN
Sue Monk Kidd	The Secret Life of Bees	916189	1.60	NaN
Kazuo Ishiguro	Never Let Me Go	294123	1.55	NaN
Jeffrey Eugenides	The Virgin Suicides	159249	1.49	NaN
Stieg Larsson, Reg Keeland	The Girl Who Kicked the Hornet's Nest (Millenn...	443951	1.41	NaN
Jhumpa Lahiri	The Namesake	184211	1.40	NaN
John Kennedy Toole, Walker Percy	A Confederacy of Dunces	170776	1.40	NaN
Jhumpa Lahiri	Interpreter of Maladies	110651	1.38	NaN
John Berendt	Midnight in the Garden of Good and Evil	167997	1.36	NaN
Arundhati Roy	The God of Small Things	165378	1.34	NaN
Diane Setterfield	The Thirteenth Tale	213200	1.34	NaN
Jonathan Franzen	Freedom	119213	1.31	NaN
John Grisham	The Chamber	102715	1.29	NaN
Jane Austen, Alfred MacAdam	Northanger Abbey	205167	1.26	NaN
Elizabeth Kostova	The Historian	190473	1.24	NaN

Table 14: The list of unrated books sorted by model scores serve as recomendations.

References

[DHS05] Chris Ding, Xiaofeng He, and Horst D. Simon. On the Equivalence of Nonnegative Matrix Factorization and Spectral Clustering. In *Proceedings of the 2005 SIAM International Conference on Data Mining*, pages 606–610. Society for Industrial and Applied Mathematics, April 2005. URL: https://epubs.siam.org/doi/10.1137/1.9781611972757.70, doi:10.1137/1.9781611972757.70.

[LS01] Daniel D. Lee and H. Sebastian Seung. Algorithms for Non-negative Matrix Factorization. In T. K. Leen, T. G. Dietterich, and V. Tresp, editors, *Advances in Neural Information Processing Systems 13*, pages 556–562. MIT Press, 2001. URL: http://papers.nips.cc/paper/1861-algorithms-for-non-negative-matrix-factorization.pdf.

[RYL+18] A. Ramlatchan, M. Yang, Q. Liu, M. Li, J. Wang, and Y. Li. A survey of matrix completion methods for recommendation systems. *Big Data Mining and Analytics*, 1(4):308–323, December 2018. URL: https://ieeexplore.ieee.org/document/8400447, doi:10.26599/BDMA.2018.9020008.

[TLLL18] Thanh Tran, Kyumin Lee, Yiming Liao, and Dongwon Lee. Regularizing Matrix Factorization with User and Item Embeddings for Recommendation. *Proceedings of the 27th ACM International Conference on Information and Knowledge Management - CIKM '18*, pages 687–696, 2018. URL: http://arxiv.org/abs/1809.00979, arXiv:1809.00979, doi:10.1145/3269206.3271730.