Recommender System

IDS 572- DATA MINING ASSIGNMENT 5

ASHUTHOSH J GOWDA

AKSHAY MERCHANT

LAVANYA GANESH

GOURAV CHOUDHARY

Assignment 5

1. (a)Explore the data to obtain an understanding of users, movies and how users have rated movies.
- what is the overall distribution of ratings

- on average, how do users rate movies; what ratings do movies have on average ? (you may want to plot the distribution of average ratings for users, movie. Can you show this on a single plot?)

- how many movies do users rate, and how many ratings do movies get? (consider the distribution of rating counts)

- how are rating levels distributed, do many people have high/low ratings?

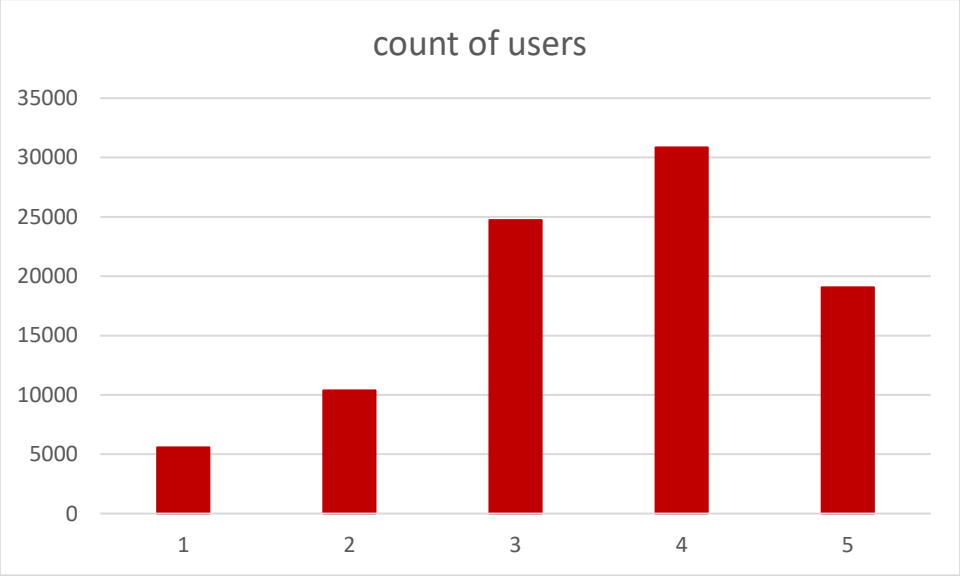
Answer 1.a.

The dataset includes variables such as:
User: Id of the user
Item: Id of the movie
Rating: Rating given by the user for the specific Item, with 1 being the lowest and 5 the highest.
Distribution of Variables

Variable	Range	Average	Standard Deviation	Mode
User	1-943	461.494	±266.003	405
Item	1-1682	428.105	±333.086	50
Rating	1-5	3.524	±1.126	4

The mode here indicates the most frequently occurring value in the variables.
For user, it means that UserId 405 has seen the most movies.
For item, it means that the movie with ItemId 50 has been seen the most number of times.
For rating, it means that a high proportion of users gave a rating of 4 for the movies that they watched.

The chart below shows the number of times each rating has been given by a user.



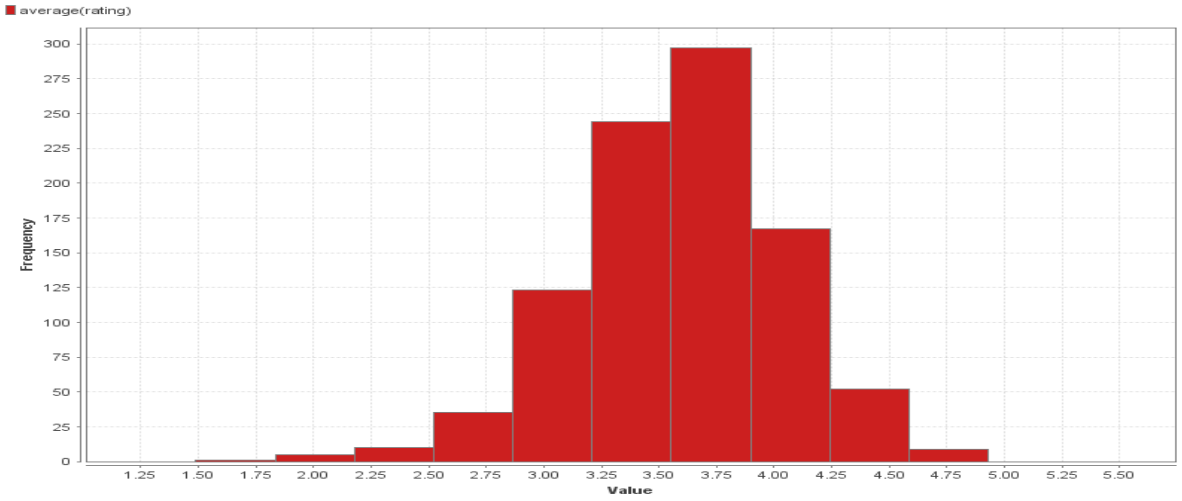
As you can see, most of the ratings are between 3 and 5.

Variable	Range	Average	Standard Deviation	Mode
Rating	1-5	3.524	±1.126	4

To find the ratings each user has given, we grouped by user and then took the average of rating.
We got the following

Variable	Minimum	Maximum	Average
Average(rating)	1.489	4.929	3.588

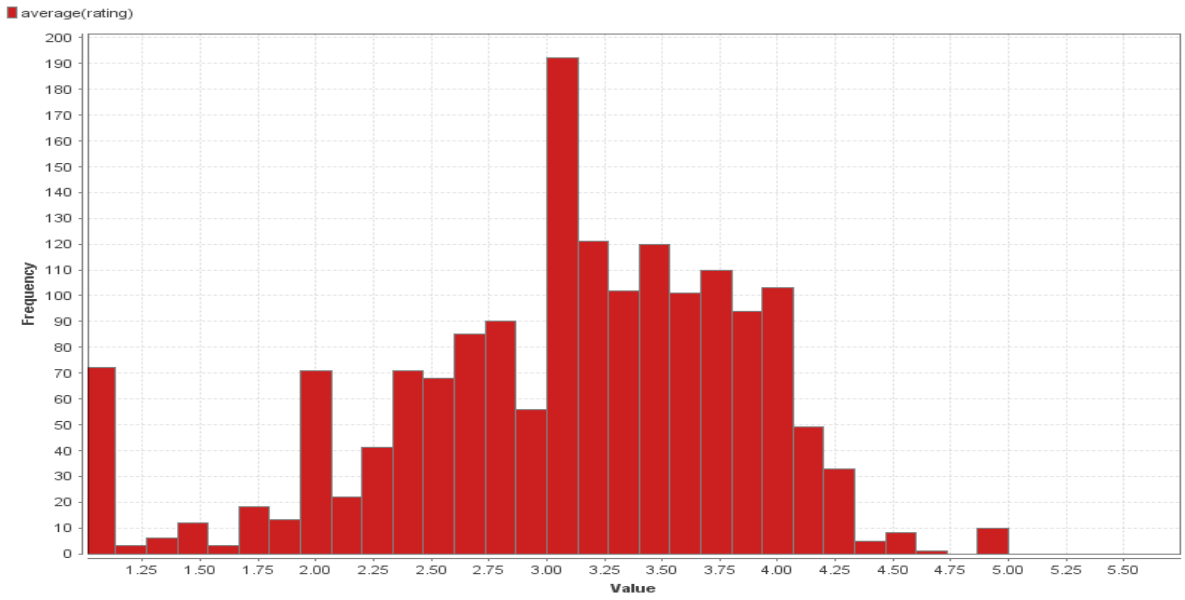
This showed that on an average, the 943 users gave a rating of 3.588



To find out what ratings movies had on average, we took the average of rating and then grouped by item.
We got the following

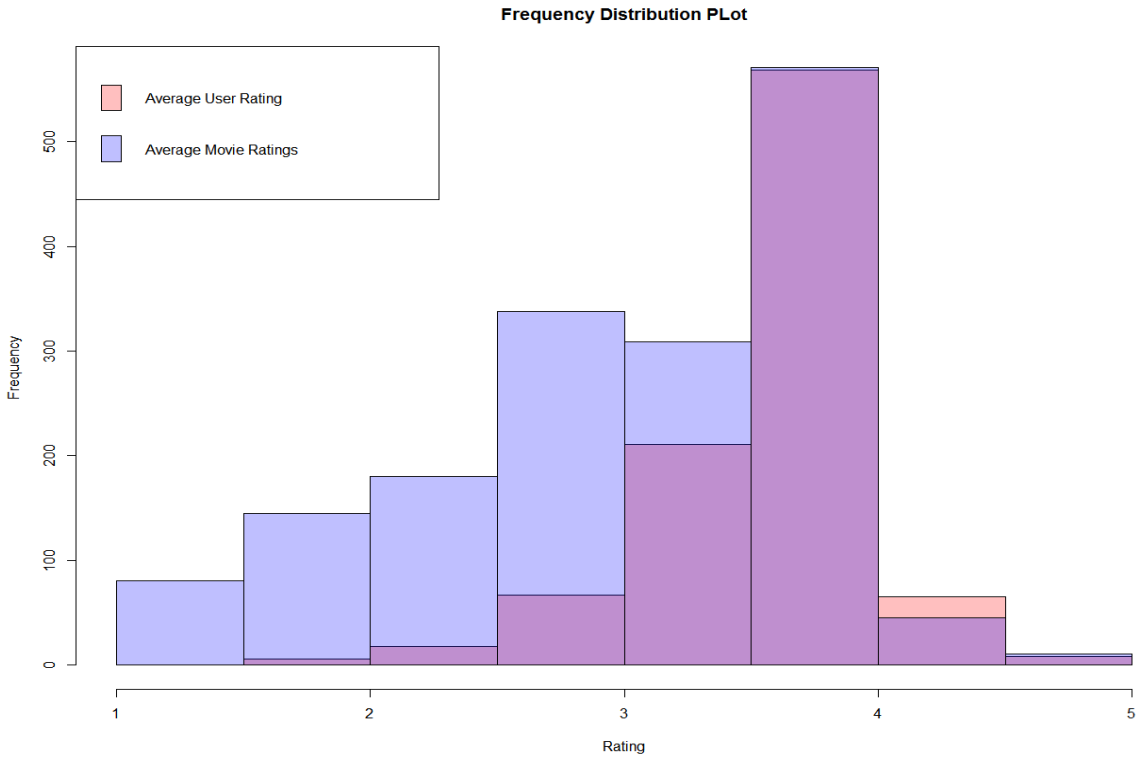
Variable	Minimum	Maximum	Average
Average(rating)	1	5	3.067

This showed that while some movies were consistently rated bad(1) by all users, and some were consistently rated good(5) by all users, the average rating for all movies was 3.067.



For items that had an average rating of either 1 or 5, we took the count of the ratings to find out how many users rated them that low or high.
We found out that those movies which had average ratings of either 1 or 5 were rated by a very small number of users.

We then combined the average ratings given by users and average ratings for movies into one histogram and got the following

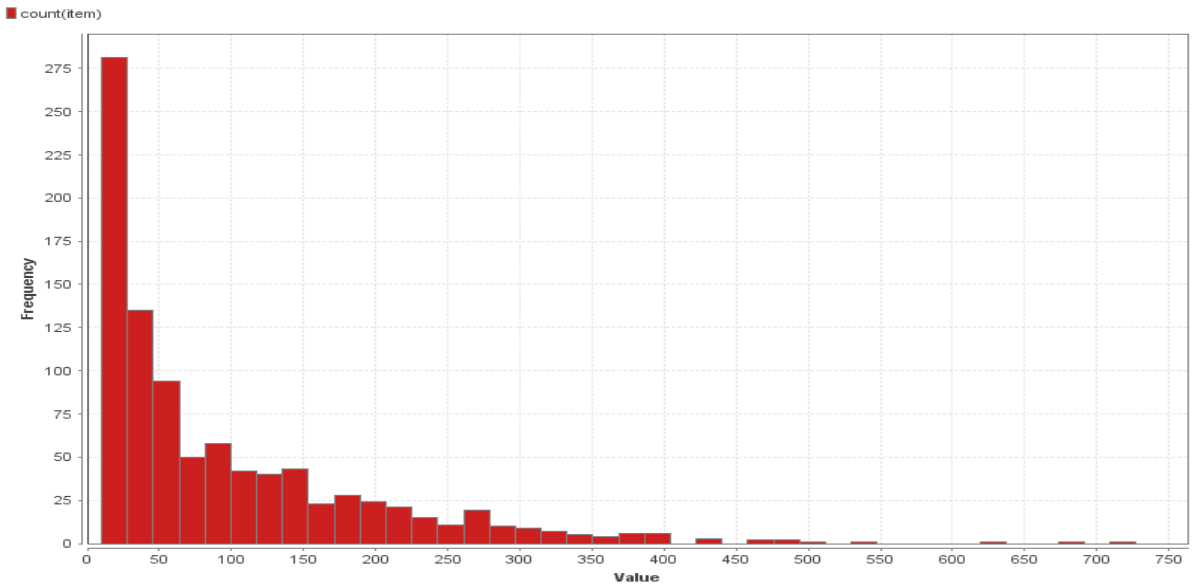


The average rating given by a user(red) and the average rating given for a movie(blue) have been plotted. The regions where there is overlap is shown in purple. The plot above shows that the spread of rated movies(blue) is greater than the spread of user ratings(red).

To find the number of movies each user had rated, we grouped by user and then counted item. We got the following

Variable	Minimum	Maximum	Average
Count(item)	10	727	96.045

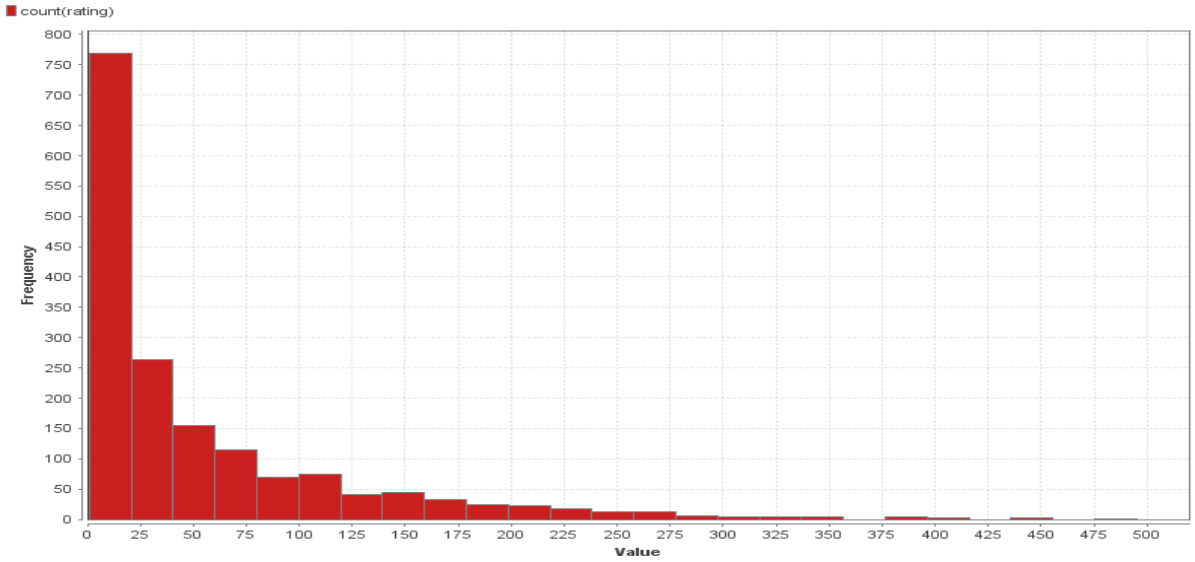
This showed that each user rated at least 10 movies, and on an average the 943 users rated 96 movies.



To find out how many ratings movies get we took the count of rating and grouped them by item. We got the following

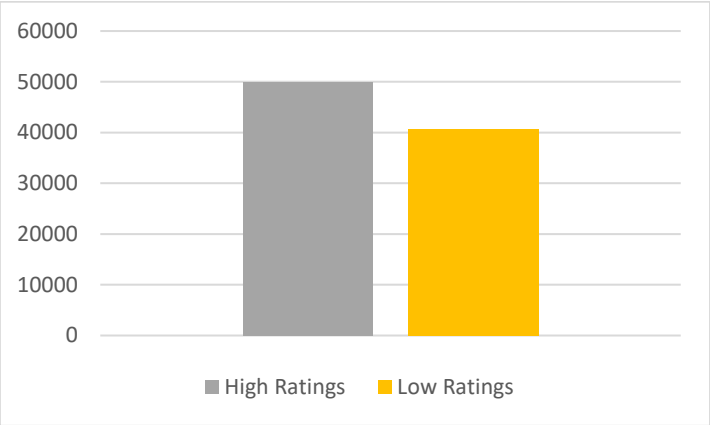
Variable	Minimum	Maximum	Average
Count(rating)	1	495	53.911

This showed that on average, movies were rated 54 times.



Rating	Proportion
1	6.148
2	11.455
3	27.295
4	34.071
5	21.031

We then divided ratings into Low and High. With ratings of 1,2,3 being labelled as low ratings and 4,5 being labelled as high ratings. We can see that the number of high ratings (even though it has less #ratings i.e. 4,5 compared to 1,2,3) is higher than the number of low ratings. This shows that more users gave movies high ratings as compared to low ratings.



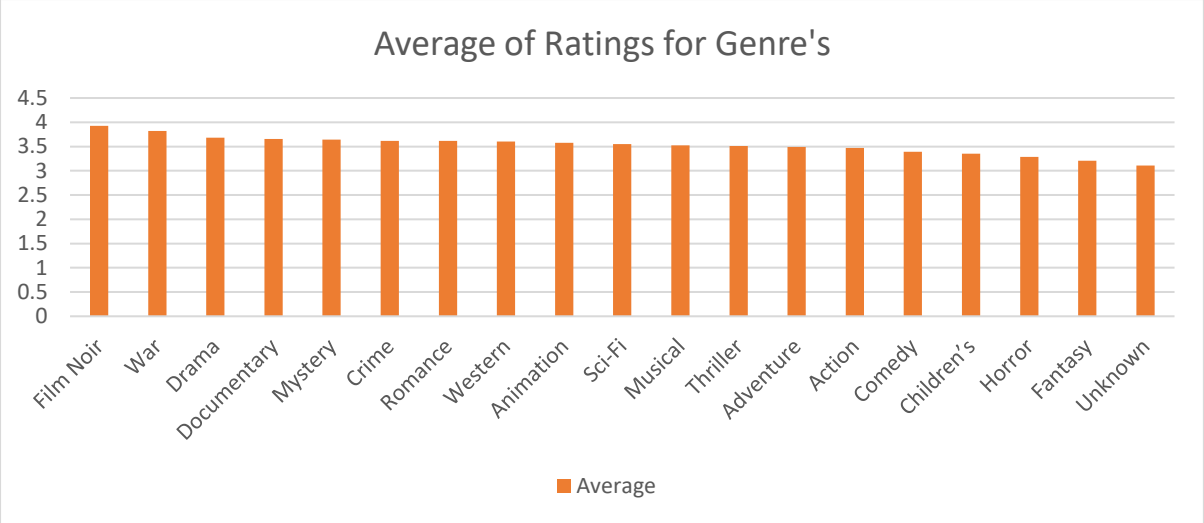
(b) Consider the movie attributes in the file u_item.csv and the user attributes in the file u_user.csv. How do ratings differ by genre, by user age (group) , gender and occupation? You can analyze this in various ways – please describe what you do and any interesting findings.

Answer 1.b.

We merged the uaTrg dataset with the u_user and u_item datasets.
Then we grouped by the different genres to get the following data

Genre	Count	Average	Standard Deviation	Mode
Action	23124	3.473	1.127	4
Adventure	12508	3.493	1.129	4
Animation	3328	3.575	1.094	4
Children's	6568	3.349	1.143	3
Comedy	27177	3.389	1.142	4
Crime	7259	3.619	1.118	4
Documentary	686	3.659	1.186	4
Drama	36011	3.682	1.080	4
Fantasy	1244	3.208	1.121	3
Film Noir	1578	3.928	0.988	4
Horror	4841	3.285	1.191	4
Musical	4560	3.525	1.111	4
Mystery	4636	3.640	1.096	4
Romance	17513	3.617	1.094	4
Sci-Fi	11487	3.553	1.135	4
Thriller	19568	3.510	1.106	4
War	8469	3.819	1.077	4
Western	1728	3.605	1.043	4
Unknown	9	3.111	1.364	4

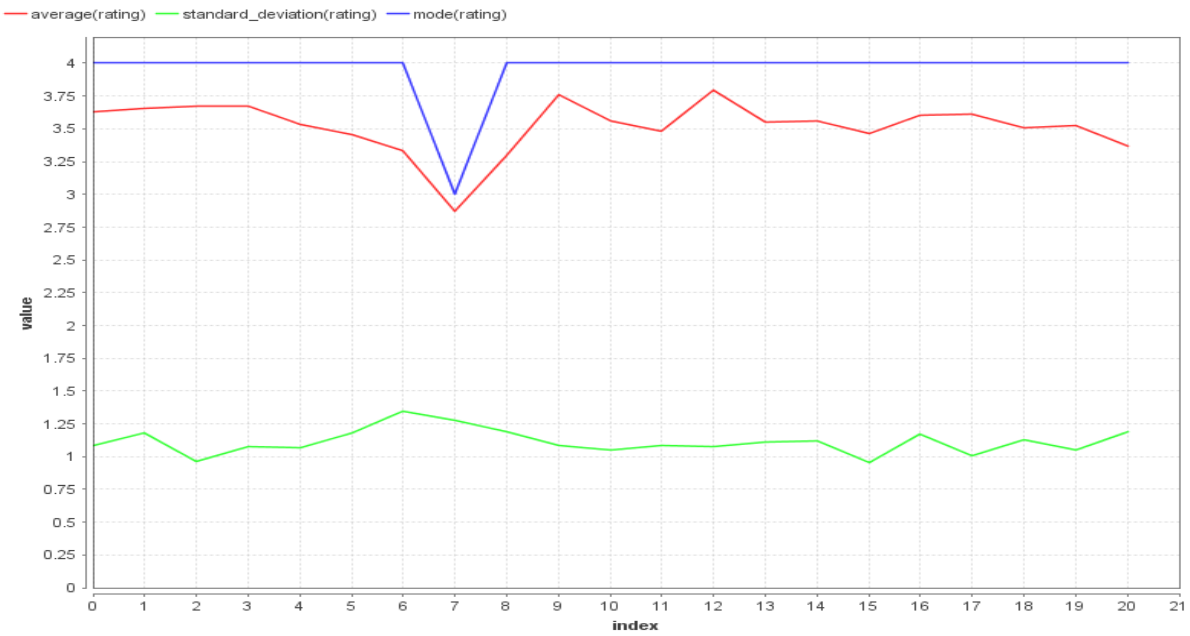
Here we saw that while most genres were similar in terms of average rating, their deviation and mode, 2 genres were consistently rated lower than the others by the users. Those were Children's movies and Fantasy movies.



We then grouped by occupation and got the following

Occupation	Count	Average	Standard Deviation	Mode	Age
Administrator	6689	3.631	1.088	4	39.168
Artist	2028	3.651	1.176	4	30.482
Doctor	470	3.676	0.965	4	34.404
Educator	8492	3.673	1.073	4	42.876
Engineer	7505	3.530	1.071	4	34.175
Entertainment	1915	3.452	1.183	4	28.723
Executive	3083	3.336	1.342	4	36.396
Healthcare	2644	2.868	1.276	3	38.723
Homemaker	229	3.301	1.192	4	32.310
Lawyer	1225	3.756	1.082	4	34.341
Librarian	4763	3.562	1.050	4	37.075
Marketing	1690	3.479	1.085	4	36.304
None	811	3.791	1.078	4	24.836
Other	9613	3.548	1.110	4	32.355
Programmer	7141	3.561	1.116	4	32.443
Retired	1469	3.463	0.954	4	61.630
Salesman	736	3.603	1.167	4	33.113
Scientist	1748	3.609	1.006	4	35.307
Student	19997	3.509	1.125	4	22.149
Technician	3236	3.521	1.046	4	31.593
Writer	5086	3.363	1.189	4	34.150

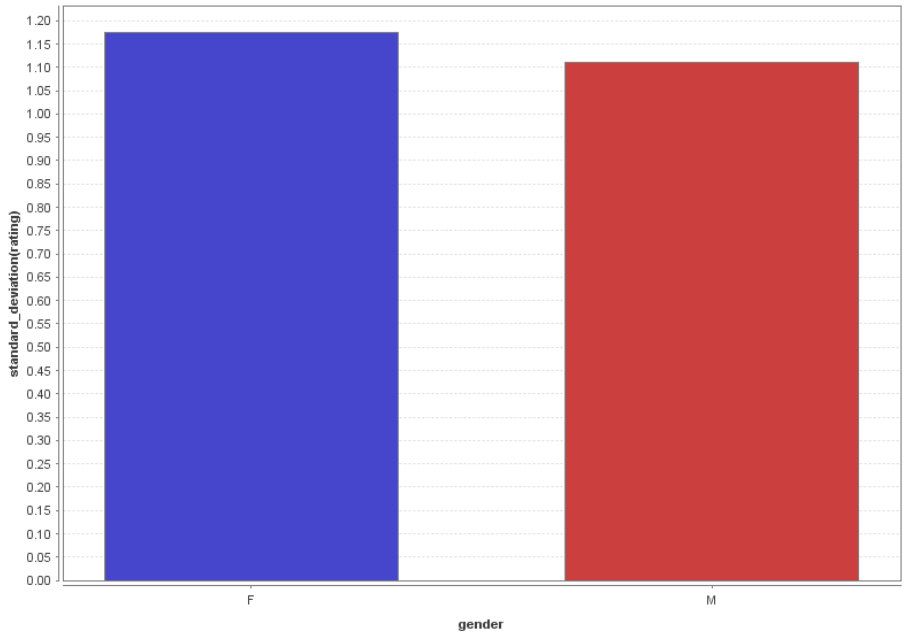
Most of the users from different occupations are similar on an average, but users from Healthcare consistently rate movies lower than users from other occupations. Users from healthcare also have one of the largest spreads amongst all the populations.



When we compared age with ratings we got the following

Gender	Count	Average	Standard Deviation	Mode	Age
F	23010	3.525	1.174	4	32.137
M	67560	3.524	1.109	4	33.102

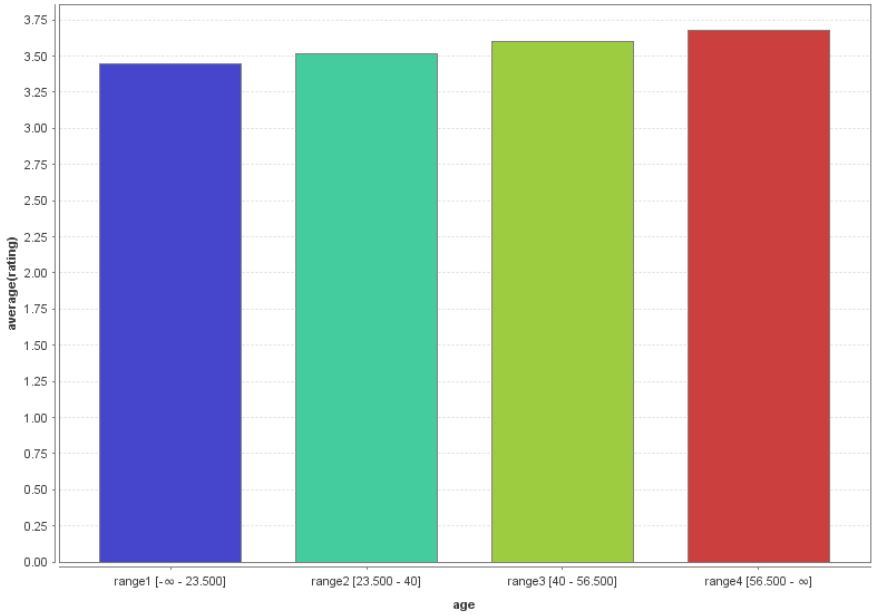
Approximately three-fourth of the users who rated movies were male.



The chart above shows that range of men's ratings was smaller than the range of women's ratings. This means that men tended to rate movies more similarly than women did.

We then took the average of ratings and grouped by age. We first binned the values in age into 4 bins.

Age	Category
0-23.5	Range 1
23.5-40	Range 2
40-56.5	Range 3
56.5-∞	Range 4



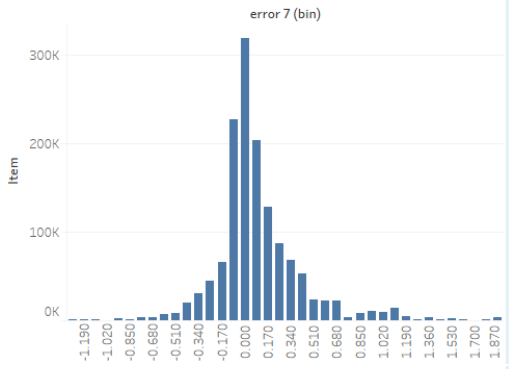
As you can see the average rating increases as users get older.

Age	Count	Average	Standard Deviation	Mode
Range 1	19985	3.445	1.2	4
Range 2	48601	3.516	1.116	4
Range 3	18162	3.6	1.083	4
Range 4	3822	3.675	1.021	4

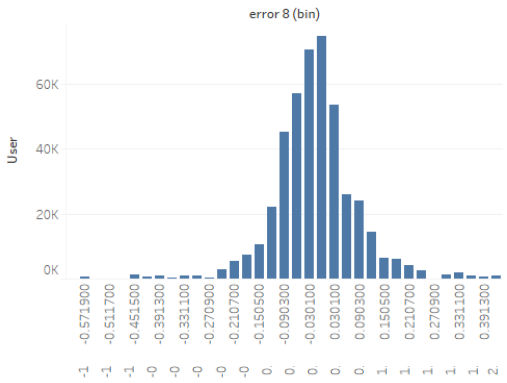
Answer 2

The error Ratings across movies and users are distributed as follows:Here, we can observe that the Error is Normally Distributed, with a low standard deviation mostly concentrated around 0. Hence we can say that the error is equally distributed across movies and users.

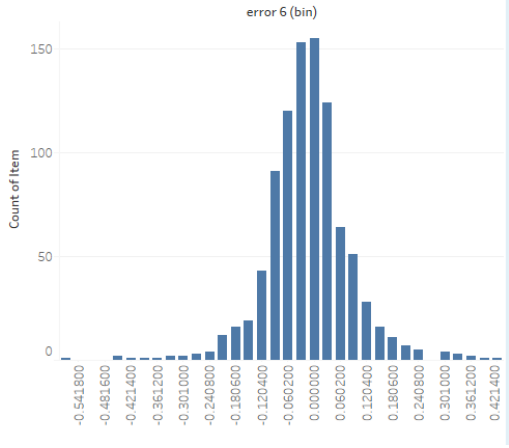
matrix factorization with train data group by item



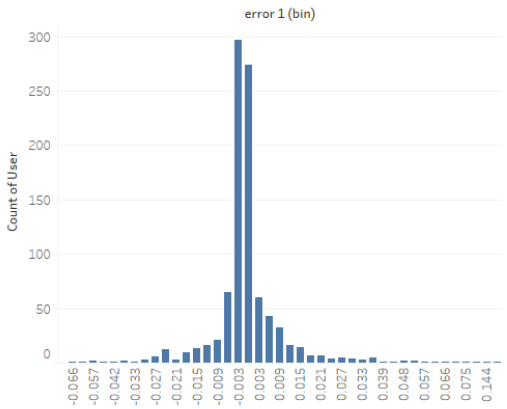
matrix factorization with train data group by user



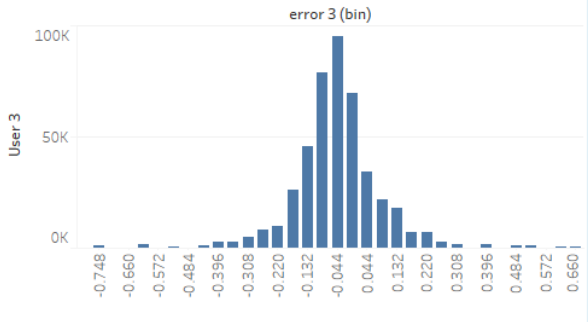
baseline train data group by item



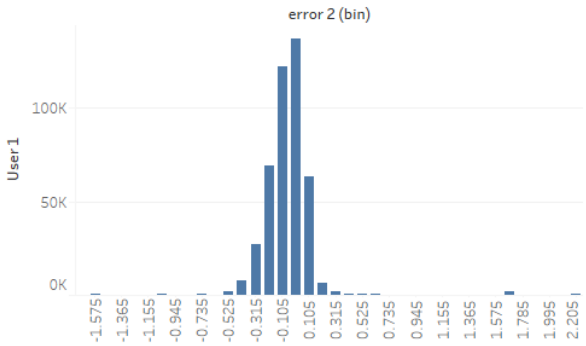
Baseline train data group by user



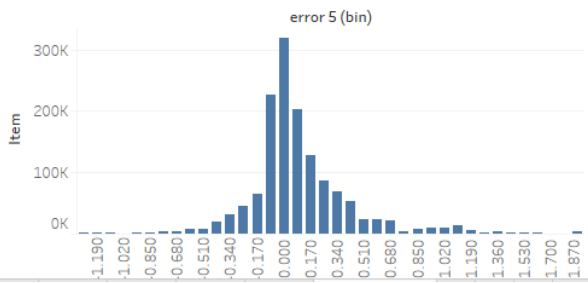
User knn train data group by user



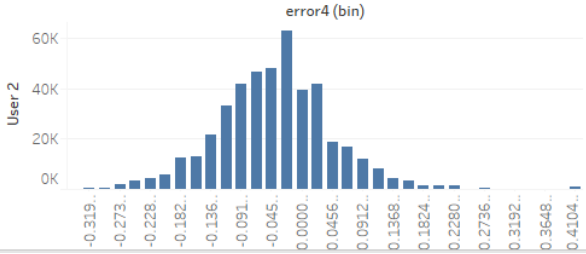
User KNN train data group by item



Item knn train data group by item



Item knn - train data group by user



To evaluate the performances of different approaches we used the Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Normalized Mean Absolute Error (NMAE) since they provide an estimate of how close the predicted ratings will be close to the actual ratings.

a) **GLOBAL AVERAGE**

Min Rating, Range	Training RMSE	Training MAE	Training NMAE	Testing RMSE	Testing MAE	Testing NMAE
1,4	1.126	0.945	0.236	1.122	0.945	0.236

Since our minimum and maximum rating is 1 and 5 respectively for any user, we are not making any changes to the parameters of global average model. Range is always 4.

The predicted rating is the average of all the user ratings and is assigned to every user. Thus it does not change and this is why the errors are comparatively high.

USER ITEM BASELINE

(Min Rate=1, Range=4)

RegU, RegI	Training RMSE	Training MAE	Training NMAE	Testing RMSE	Testing MAE	Testing NMAE
0,0	0.916	0.72	0.18	0.975	0.761	0.19
1.5,1.5	0.91	0.718	0.179	0.961	0.756	0.189
3,2	0.911	0.719	0.180	0.959	0.757	0.189
15,15	0.924	0.733	0.183	0.964	0.766	0.192

The changes in number of iterations does not make any difference to the RMSE or MAE. As we increase the regularization parameters from 0,0, the error reduces first and then it starts increasing.

For low regularization parameters, the bias is low and the model is taking most users for predicting the movie rating. As regularization increases, the bias also increases and hence less number of users are taken into consideration to predict the movie rating.

Since the error values are less for training and test data when compared to global average, we chose the User Item Baseline Operator over the Global Average Operator.

b) **MATRIX FACTORIZATION**

(Min Rating=1, Range=4, Iteration Number =30, Regularization= 0.015, Initial Mean=0, Standard Deviation= 0.1)

Number of factors	Training RMSE	Training MAE	Training NMAE	Testing RMSE	Testing MAE	Testing NMAE
5	0.815	0.639	0.16	0.955	0.753	0.188
10	0.745	0.584	0.146	0.975	0.767	0.192

20	0.635	0.497	0.124	1.003	0.785	0.196

As we increase the number of factors, the RMSE and MAE for training and test data decreases up to a certain point (10). Beyond the certain point the RMSE and MAE for training data decreases and for testing data increases.

Learn Rate	Training RMSE	Training MAE	Training NMAE	Testing RMSE	Testing MAE	Testing NMAE
0.01	0.745	0.584	0.146	0.975	0.767	0.192
0.1	0.794	0.611	0.153	1.09	0.844	0.211
Regularization	**Training RMSE**	**Training MAE**	**Training NMAE**	**Testing RMSE**	**Testing MAE**	**Testing NMAE**
0.05	0.771	0.543	0.166	0.95	0.711	0.175
0.035	0.757	0.592	0.172	0.958	0.762	0.171

The lower the learning rate, lower are RMSE and MAE for the model. As the learning rate increases, the RMSE and MAE for training and test data increases. The lower the regularization, the lower is the RMSE and MAE for the model.

c) USER K-NN

k Cosine Similarity	Training RMSE	Training MAE	Training NMAE	Testing RMSE	Testing MAE	Testing NMAE
10	0.939	0.736	0.184	0.979	0.771	0.193
11	0.937	0.735	0.184	0.975	0.768	0.192
20	0.926	0.726	0.182	0.96	0.757	0.189
21	0.925	0.726	0.181	0.96	0.756	0.189
51	0.922	0.724	0.181	0.957	0.754	0.188
55	0.922	0.724	0.181	0.957	0.754	0.188

k Pearson Similarity	Training RMSE	Training MAE	Training NMAE	Testing RMSE	Testing MAE	Testing NMAE
10	0.758	0.587	0.147	0.963	0.756	0.189
11	0.758	0.587	0.147	0.961	0.755	0.189
20	0.765	0.593	0.148	0.952	0.748	0.187
21	0.765	0.594	0.148	0.951	0.747	0.187
51	0.79	0.615	0.154	0.949	0.745	0.186
55	0.793	0.617	0.154	0.949	0.745	0.186
60	0.795	0.619	0.155	0.949	0.745	0.186

On running the optimized parameter grid, we got the optimal k for Cosine similarity to be 51 and for Pearson similarity to be around 11. On varying the number of nearest neighbours, we observe that as we increase the k in combination with cosine similarity, the RMSE and MAE for training and testing data decrease up to the optimal value of k=51 and remain constant for some values after it and increase thereafter. On varying the number of nearest neighbours, we observe that as we increase the k in combination with Pearson similarity, the RMSE and MAE for training and testing data decrease

up to the optimal value of k=11 and almost constant for some values after it. Thereafter the RMSE and MAE for training increase and testing continues to decrease.

Comparing the values of k (51 & 11) that give the optimal performance for cosine and Pearson similarity, we find that Pearson similarity gives lesser RMSE and MAE values. Hence we choose Pearson similarity over Cosine similarity for User K-nn

ITEM K-NN

k Cosine Similarity	Training RMSE	Training MAE	Training NMAE	Testing RMSE	Testing MAE	Testing NMAE
10	0.902	0.703	0.176	0.946	0.739	0.185
11	0.901	0.702	0.175	0.945	0.739	0.185
20	0.895	0.699	0.175	0.939	0.736	0.184
21	0.894	0.699	0.175	0.939	0.737	0.184
51	0.898	0.705	0.176	0.943	0.741	0.185
55	0.899	0.705	0.176	0.944	0.742	0.185

k Pearson Similarity	Training RMSE	Training MAE	Training NMAE	Testing RMSE	Testing MAE	Testing NMAE
10	0.674	0.523	0.131	0.946	0.74	0.185
11	0.674	0.524	0.131	0.931	0.739	0.185
20	0.688	0.535	0.134	0.937	0.735	0.184
21	0.689	0.536	0.134	0.937	0.735	0.184
51	0.725	0.566	0.141	0.937	0.736	0.184
55	0.728	0.569	0.142	0.937	0.736	0.184

On running the optimized parameter grid, we got the optimal k for Cosine similarity to be 21 and for Pearson similarity to be around 11. On varying the number of nearest neighbours, we observe that as we increase the k in combination with cosine similarity, the RMSE and MAE for training and testing data decrease up to the optimal value of k=21 and remain constant for some values after it and increase thereafter. On varying the number of nearest neighbours, we observe that as we increase the k in combination with Pearson similarity, the RMSE and MAE for training and testing data decrease up to the optimal value of k=11 and remain almost constant for some values after it. Thereafter the RMSE and MAE for training increasing and testing continues to decrease.

Comparing the values of k (21 & 11) that give the optimal performance for cosine and Pearson similarity, we get Pearson similarity gives lesser RMSE and MAE measures. Hence we choose Pearson similarity over Cosine similarity for Item K-nn.

On comparing the optimal performances (k=11) of Pearson similarity for User k-nn and Item k-nn, we observe that Pearson similarity for item k-nn gives a lesser RMSE and MAE. Hence, we choose Item K-nn with Pearson similarity.

d) COMBINER MODEL

Model 1	Model 2	RMSE		MAE		NMAE	
		Train	Test	Train	Test	Train	Test
User KNN	Matrix Factorization	0.739	0.932	0.582	0.738	0.146	0.185
User KNN	Global Average	0.871	0.983	0.701	0.794	0.175	0.198
Global Average	Matrix Factorization	0.902	0.997	0.740	0.812	0.185	0.203

After running the Model combiner on a combination of all our best models, we observed that the combination of User K-NN and Matrix Factorization gives a lower RMSE and MAE as compared to the combinations of User K-NN & Global Average, Global Average & Matrix Factorization.

Overall when we compared our best models from Global Average, User Item Baseline, User K-NN, Item K-NN and Matrix Factorization, we observed that the best performance on the basis of RMSE and MAE was given by Item K-NN (k=11 and Pearson).

Answer 3

The data below shows different cut-off values for different operators. For this we have assumed that all rating values above the cut-off as high, and those below the cut-off as low.

User K-NN

Cut-off	Accuracy	Recall	Precision
3.1	69.02%	92.63%	66.80%
3.2	69.85%	89.83%	68.24%
3.3	70.76%	86.41%	70.12%
3.4	71.36%	82.23%	72.23%
3.5	71.01%	76.92%	74.08%

Cut-Off=3.4	Actual(0)	Actual(1)	Precision
Prediction(0)	2232	972	
Prediction(1)	1729	4497	72.23%
Recall		82.23%	**Accuracy=71.36%**

As we increase the cut-off value, the accuracy and precision increase initially, while the recall decreases. Therefore, we have taken a cut-off value of 3.4 since it gives the best accuracy while having high values of precision and recall.

User Baseline

Cut-off	Accuracy	Recall	Precision
3.1	68.78%	90.64%	67.09%
3.2	69.49%	87.62%	68.54%
3.3	70.15%	83.78%	70.38%
3.4	70.32%	79.56%	72.13%
3.5	69.35%	73.71%	73.52%

Cut-Off=3.4	Actual(0)	Actual(1)	Precision
Prediction(0)	2280	1118	
Prediction(1)	1681	4351	72.13%
Recall		79.56%	**Accuracy=70.32%**

As we increase the cut-off value, the accuracy and precision increase initially, while the recall decreases. Therefore, we have taken a cut-off value of 3.4 since it gives the best accuracy while having high values of precision and recall.

Global Average

Cut-Off=3.5	Actual(0)	Actual(1)	Precision
Prediction(0)	0	0	
Prediction(1)	3961	5469	58%
Recall		100%	**Accuracy=58.00%**

In this operator, since all the predicted ratings are the same, it makes no sense to try different cut-offs. This is because after the cut-off crosses the average prediction value, the model will have 0 recall and precision.

Item K-NN

Cut-off	Accuracy	Recall	Precision
3.1	72.65%	92.14%	70.10%
3.2	74.29%	90.11%	72.35%
3.3	75.65%	87.60%	74.75%
3.5	76.88%	80.36%	79.89%
3.6	76.09%	75.10%	82.14%

Cut-Off=3.5	Actual(0)	Actual(1)	Precision
Prediction(0)	2855	1074	
Prediction(1)	1106	4395	79.89%
Recall		80.36%	**Accuracy=76.88%**

As we increase the cut-off value, the accuracy and precision increase, while the recall decreases. Therefore, we have taken a cut-off value of 3.5 since it gives the best accuracy while having high values of precision and recall.

Matrix Factorization

Cut-off	Accuracy	Recall	Precision
3.3	74.61%	98.57%	69.95%
3.4	78.93%	96.32%	74.68%
3.5	83.20%	91.00%	82.01%
3.6	82.51%	78.55%	90.03%
3.7	74.97%	60.63%	94.12%

Cut-Off=3.5	Actual(0)	Actual(1)	Precision
Prediction(0)	2869	492	
Prediction(1)	1092	4977	82.01%
Recall		91.00%	**Accuracy=83.20%**

As we increase the cut-off value, the accuracy and precision increase initially, while the recall decreases. Therefore, we have taken a cut-off value of 3.5 since it gives the best accuracy while having high values of precision and recall.

The objective of the decision support is to recommend movies with high ratings to users. Therefore we have tried different cut-off values to get more optimal rating predictions.

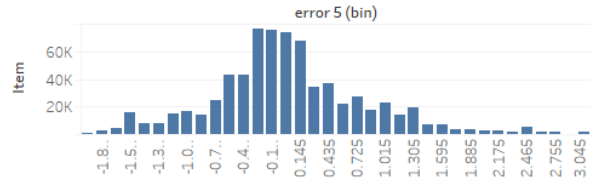
Model	Cut-off	Accuracy	Recall	Precision	True Positives
User K-NN	3.4	71.36%	82.23%	72.23%	4497
User Baseline	3.4	70.32%	79.56%	72.13%	4351
Global Average	3.5	58%	100%	58%	5469
Matrix Factorization	3.5	76.88%	80.36%	79.89%	4395
Item K-NN	3.5	83.20%	91.00%	82.01%	4977

As you can see, Item K-NN has by far the better accuracy, recall and precision. Item K-NN also has a comparatively large number of True positives (actual 1's predicted as 1's). This is why we think Item K-NN is the better model for recommending movies to users.

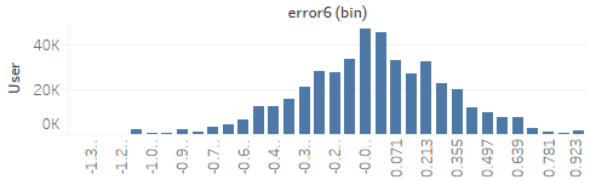
The distribution of error across movies and users are as follows:

Here, we can observe that the Error is Normally Distributed, with a high standard deviation. Hence we can say that the error is not equally distributed across movies and users.

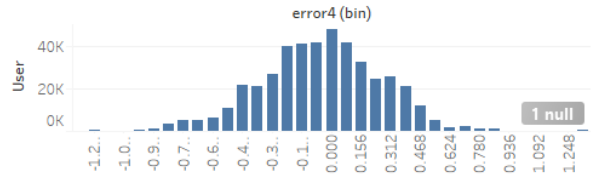
matrix factorization with test data group by item



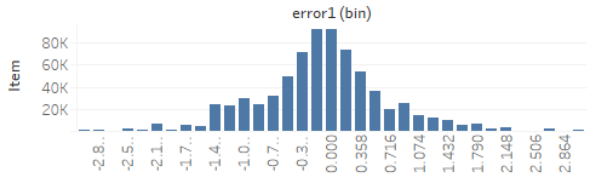
matrix factorization with test data group by user



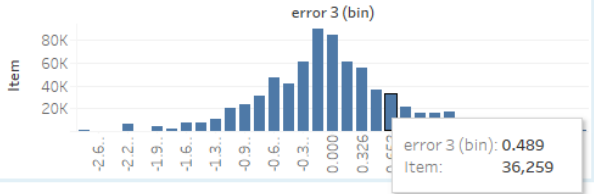
item knn with test data group by user



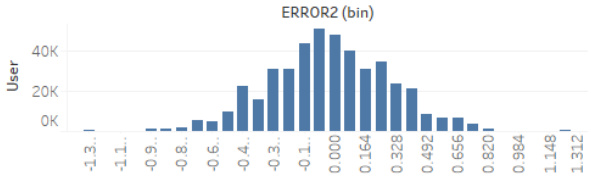
Base line with test data group by item



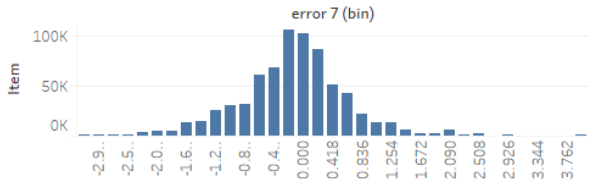
item knn with test data group by item



base line with test data group by user



user knn with test data group by item



user knn with test data group by user