
Recommendation Systems

Pulkit Gopalani
180564

Achint Agrawal
180028

Abstract

Recommendation systems are used by many platforms to suggest items to their users. The present implementation is an attempt at recommending songs. The recommendations are based on collaborative filtering using cosine similarity as well as Pearson Correlation Coefficient (Using SVD Matrix completion method). The first method is more of a naive approach, calculating similarities of all users, while the second one relies on Matrix factorization based on SVD. The dataset used for the project is the Million Song Dataset, from Kaggle.com .

GitHub link for the project : <https://github.com/Pulkit-Go/PT-Project>

1. Background

1.1. Introduction to Recommender systems

Recommender systems are being used by many product based platforms like Amazon, Netflix, Spotify etc where user needs to be "recommended" some items, as the total number of items is too large to be explored fully by the user. Out of billions of songs and products, the most relevant are suggested to the user based on his past preferences, and his similarity to other users based on these past preferences. This is based on the basic assumption that if 2 people have listened to similar songs in the past, they will also like similar songs in the future.

These platforms use sophisticated ML based algorithms like K-Nearest neighbors, Clustering etc. and a mixture of Collaborative filtering based and content based recommendations.

1.2. Similarity measures

Similarity measures are basically the different type of methods used to calculate how similar two users or items are, based on the difference between the specific ratings. Two measures are quite popular - Cosine based similarity and Pearson Correlation Coefficient. One major difference between the two is that cosine similarity is based on raw data, whereas Pearson correlation coefficient is more of a "normalized" measure, meaning that the raw data is not directly used, instead the standardized data (subtracting the mean from each entry of a given row/column) is used. Therefore it can (and does) lead to negative correlation, which signifies that the two objects are far from being similar.

1.3. Evaluation Metrics

The evaluation is an important part of the recommender system design, telling us how accurate the prediction model actually is. The evaluation is done using MAP (Mean Average Precision) score, in which some of the songs listened by the user are masked, and based on the remaining history, the predictions and the hidden songs are compared. More the matching songs, better the MAP score. Basically,

$$\text{MAP score} = \frac{\text{Number of matching predictions}}{\text{Total number of hidden songs}}$$

2. Problem Overview and Approach to solution

2.1. Dataset used for recommendations

The data used for calculating similarities, making recommendations and evaluation is the Million Song Dataset from Kaggle, which has the data of a million users and their usage history. In the present project we have used the kaggle-song-evaluation-triplets file which has the User ID of each user, corresponding to which the songs which have been listened to by the user have been listed, along with the number of times the specific song has been listened.

One issue with such kind of dataset is that people don't listen to as many songs as there are available. The average user has listened to only 30-35 songs, while the data is available for thousands of songs. This leads to a sparse data matrix, with most of the elements empty (the matrix in question here has a sparsity >0.95 , i.e. in a 10 x 10 sub-matrix of the matrix, there are high chances that only 5-6

elements are actually filled). There are many users who have not listened to many songs (< 5). So to address these issues, the data has been filtered out to only the top 2000 users with the most number of songs listened. The matrix is then effectively converted to a denser matrix with comparatively much more entries per user.

2.2. Calculating similarities

2.2.1. Cosine based similarity

Cosine similarity is based on the property of vectors that the cosine of angle between them is the dot product of the normalized vectors (unit norm vectors pointing along the original vectors). More specifically,

$$\cos \theta = \frac{\mathbf{x} \cdot \mathbf{y}}{\|\mathbf{x}\| \|\mathbf{y}\|}$$

This means the similarity will lie between -1 and 1. For this particular case, the ratings are non-negative, so the cosine based similarity will also be non-negative, i.e. in the interval $[0, 1]$. We take the product of a specific user's ratings with that of another user (the vector here has a dimension = number of songs). Similarity therefore can be written as:

$$\text{sim}(i, j) = \frac{\sum_{s \in S_{ij}} r_{i,s} * r_{j,s}}{\sqrt{\sum_{s \in S_{ij}} r_{i,s}^2} \sqrt{\sum_{s \in S_{ij}} r_{j,s}^2}}$$

where S_{ij} is the set of songs for which both users have rated it a non-zero number. $r_{i,s}$ and $r_{j,s}$ is the rating given by the i-th and j-th user.

2.2.2. Pearson Correlation Coefficient

Pearson correlation is very much similar to cosine based similarity, differing only in the fact that it considers the standardized or mean-relative ratings for calculating correlations. In this case the correlation coefficients can be both positive or negative based on whether the items being compared are alike or different.

Therefore mathematically PCC can be written as

$$\text{corr}(i, j) = \frac{\sum_{s \in S_{ij}} (r_{i,s} - \bar{r}_i) * (r_{j,s} - \bar{r}_j)}{\sqrt{\sum_{s \in S_{ij}} (r_{i,s} - \bar{r}_i)^2} \sqrt{\sum_{s \in S_{ij}} (r_{j,s} - \bar{r}_j)^2}}$$

The calculation of predicted ratings in this case is done using SVD and matrix completion methods.

2.3. Predicting the ratings and recommending songs

2.3.1. Rating based on user-user similarity

In this part, weights assigned to the users are their similarities with the user in question. As can be seen,

$$r_{c,s} = \frac{\sum_{c' \in C} \text{sim}(c, c') * r_{c',s}}{\sum_{c' \in C} \text{sim}(c, c')}$$

is the predicted rating for user c and song s . Note that more the similarity between c and c' , more the weightage given to c' rating for predicting the rating of c .

Now, using `argsort` and `argsort` functions of `numpy` library the top- k songs (here $k = 50$) based on predicted ratings, not listened to earlier were recommended to the user.

2.3.2. Rating based on matrix completion

The `TruncateSVD` method of `scikit-learn` library was used to carry out the matrix decomposition in this project. It reduces the dimensionality of data being used and then completes the matrix. Based on this matrix the correlation coefficients of all songs vs themselves were calculated using `Numpy's corrcoef()` function. The higher the correlation, more similar the songs were. Based on these the highest correlated songs (w.r.t. the song rated best by the given user) were recommended to the user.

3. Implementation and evaluation

The input was read from the file "Kaggle-evaluation-triplets" using `.strip` and `.read` functions, used for file handling in python.

3.1. Recommendations based on user-user similarity

The program was run for 100 and 700 users subsequently, and the snapshots are appended below.

As can be observed, the time taken by this method is too large for higher number of users, a problem which is solved to some extent in the next implementation.

3.2. Recommendations based on song-song similarity (SVD matrix completion)

This implementation is faster than the above implementation, taking almost one-fourth of the time for 100 users than in the first case. As in the previous case the numbers indicate the song indices.

This program too was run for 100 and 700 users subsequently. The output snapshots are appended below.

As can be seen, this method is much faster than the previous one in case of recommending songs.

```

File Edit View Search Terminal Help
3158 2803 2815 3058 463 2401 2397 174 1877 2399 2807 3680 2796 2793
2802 2799 3068 2789 3704 501 2227 711]
User 91 : [3024 3001 2246 2999 3014 215 2259 3162 2998 3018 2995 609 284 3015
2250 1053 2255 3032 3023 2247 288 882 2244 1857 1272 3031 1976 2263
2260 2243 1977 825 2241 2240 2234 2235 1991 409 3019 2245 407 1978
1247 3757 2248 2236 2238 285 2252 2264]
User 92 : [ 320 1177 2857 2836 1735 1194 569 1592 1857 2835 174 1602 1584 588
1596 2830 1217 2854 1589 1869 1587 1601 1605 1597 1276 1118 1859 1579
1853 1610 1846 583 1844 1121 570 1581 1850 671 1383 1840 1604 1583
2822 1874 1195 1097 2846 1211 565 1096]
User 93 : [1194 320 1177 322 3115 2481 3139 3143 1766 1388 3148 3135 3120 3137
3131 1976 1977 1991 3140 3147 1978 1688 3146 3142 1979 1217 3141 3117
2727 3136 3753 3125 3132 3133 3118 665 2738 1756 3119 3122 3116 1726
2277 60 3124 3134 3128 143 2733 2730]
User 94 : [1396 1373 2296 2246 2259 4148 4159 498 480 1372 1389 4155 1379 1357
1388 1384 1390 2289 489 1370 2250 2255 1362 1378 1354 2247 474 890
1360 4146 4168 1272 74 502 2244 1857 473 4150 964 1377 462 1385
482 4165 4156 4162 2263 488 4158 4157]
User 95 : [1396 1373 2019 2030 489 498 2028 480 2027 2026 1372 2044 1389 1384
1388 1379 1357 1390 1362 1370 1572 1378 174 253 1354 1360 2227 1385
74 462 1857 502 493 473 1377 464 469 467 1364 482 488 500
1374 1381 491 501 485 1387 486 1383]
User 96 : [3658 3654 3667 3675 3642 3659 3660 3661 3663 3664 3665 3678 3668 3669
3670 3671 3672 3673 3674 3666 3657 3655 3676 3679 3637 3638 3639 3640
3641 3643 3644 3645 3646 3647 3648 3649 3650 3651 3652 3653 3656 3677]
User 98 : [2019 2030 2028 2027 2026 2044 890 1857 2035 174 2018 2025 2005 2008
885 2029 1194 1276 3753 2034 1846 164 1859 981 976 1853 1869 1844
1172 1162 2277 1850 1840 160 968 888 1874 29 2009 4010 1845 2038
984 1858 2043 1855 1158 605 1132 2016]
User 99 : [ 489 909 908 907 1895 1899 1880 915 905 174 936 898 1572 925
939 933 921 938 944 1890 1902 926 912 930 1412 409 934 1891
268 1879 1549 1906 1430 1896 945 906 931 407 1905 30 917 366
1571 3409 927 940 1912 941 662 1413]
MAP score = 0.05
Time taken: 10.524
Number of recommendations made: 96
achint@achint-Lenovo-G570:~/Documents/project track/PT-Project-master$ █

```

Figure 1: 100 users, first approach

```

File Edit View Search Terminal Help
1666 407 366 716 9232 14558 6447 730 1844 1846 10583 1859
7974 15791 377 1704 1194 1853 5055 387 1276 467 1874 3181
8702 12435]
User 694 : [ 4027 6800 13310 1477 15742 15743 174 716 7171 9322 2093 4049
7160 7159 4032 7173 7170 7164 15740 17625 14576 409 5283 7681
13306 10199 10139 366 852 4174 5794 407 12681 1566 387 12685
662 3052 2707 2266 7222 9045 4056 4055 1667 8195 15749 4040
377 4028]
User 695 : [19086 6814 1326 463 12256 2227 7099 1272 2692 1407 1857 1726
174 9284 17893 2701 9297 3115 3159 2818 1075 8819 19076 501
120 1853 1276 1023 2673 1859 1846 4335 1194 3629 19079 882
662 1876 3148 6339 15115 1688 1840 5081 268 3180 826 15740
4275 830]
User 696 : [ 3193 3211 3207 3202 7326 3205 239 15392 1917 3902 7977 249
7980 3894 2979 6413 3629 7971 8056 1053 3899 787 7985 1354
7976 2984 7978 3905 2790 1577 234 4447 7605 8036 7983 10358
4565 1312 3900 1845 7989 7972 5770 1364 7982 7990 7975 7979
5992 7984]
User 697 : [16619 13000 18233 1056 5486 18235 1039 4175 6268 3 13403 954
7727 2019 18238 6274 6265 16178 3530 19086 1837 2030 1069 2227
6276 1060 1850 18241 4317 662 463 6279 1407 11813 18230 882
11506 14576 1053 1063 7681 473 1071 1074 1041 1051 1045 6734
19079 1023]
User 698 : [ 6092 4499 4509 4488 1462 4511 4671 6088 6078 6060 13115 7977
1017 1577 4476 4478 6066 6082 1688 381 4514 3490 4493 6091
11288 6073 15352 11757 5383 4477 4490 3146 4482 4518 4489 4513
5171 4485 3304 3319 917 2668 13325 5077 6055 3899 6059 6061
12008 6074]
User 699 : [ 186 1015 1952 6401 3866 1949 10569 10574 16105 18211 18229 16285
1955 18212 171 201 196 13353 999 18207 13346 773 18227 760
10566 2257 202 305 4603 1928 204 18213 911 10817 3181 18220
8438 10563 18201 192 1362 904 10578 6162 18228 10580 5241 18202
18210 18214]
MAP score = 0.05
Time taken: 452.58
Number of recommendations made: 700
achint@achint-Lenovo-G570:~/Documents/project track/PT-Project-master$

```

Figure 2: 700 users, first approach

```

File Edit View Search Terminal Help
3898 3900 3901 3902 3903 3905 3906 3907 3908 3909 3910 3880 3911 3887
3883 3881 3888 3882 3886 3889 3885 3884 3914 3894 3904 3899 2267 1062
2624 1064 1801 367 2869 2863 2890 2884]
User 92 : [ 254 237 234 236 227 244 239 221 231 226 223 222 220 219
218 216 213 212 211 210 208 209 232 217 228 225 256 235
240 241 242 230 243 245 246 238 248 249 251 252 247 255
362 250 2618 2281 2282 2283 2284 2285]
User 93 : [1212 882 1439 2227 1293 3278 777 880 875 333 2610 2412 2615 2594
2593 2592 2590 2589 2588 2587 2586 2619 2625 2626 2627 2595 2616 2601
2607 2598 2600 2621 2602 2603 2614 2604 2605 2606 2617 2613 2609 2611
2612 2597 2623 2608 2591 2620 2596 2622]
User 94 : [3753 1879 1895 1897 1898 1899 1900 1902 1903 1913 1907 1908 1909 1910
1911 1912 1914 1904 1894 1887 1883 1884 1885 1893 1886 1882 1888 1881
1890 1880 1889 1901 1896 1892 1905 1906 1878 1406 1891 665 2272 1603
3691 1272 2935 2950 2934 2906 2925 2938]
User 95 : [4039 4035 4036 4037 4040 4041 4042 4043 4044 4046 4034 4047 4049 4059
4058 4057 4056 4055 4054 4053 4052 4048 4051 4033 4031 4021 4032 4023
4024 4025 4022 4027 4026 4030 4029 4050 4028 1986 3120 3121 3148 3129
3130 3135 3138 3139 3147 3127 3149 3116]
User 96 : [3101 3100 3112 3103 3088 3098 3092 3082 3081 3080 3079 3078 3074 3075
3083 3109 3110 3111 3114 3107 3077 3108 3113 3085 3084 3105 3104 3102
3099 3097 3096 3106 3094 3093 3090 3089 3087 3095 3086 2398 3532 3531
3528 3526 3525 3520 3516 3518 3517 3515]
User 97 : [4113 4122 4121 4120 4119 4116 4115 4114 4112 4111 4110 4107 4106 4104
4103 4102 4126 4127 4118 4139 4128 4138 4137 4141 4136 4135 4142 4134
4132 4130 4143 4144 4133 4117 4108 4105 4131 4125 4109 4140 4129 4123
4124 1613 4038 2381 2410 2400 2390 2404]
User 98 : [4181 4163 4164 4165 4166 4167 4168 4180 4169 4171 4172 4175 4176 4177
4179 4170 4160 4162 4149 4145 4146 4147 4159 4150 4151 4182 4154 4155
4156 4157 4158 4153 4148 4173 4174 4178 4152 4161 1156 1163 1283 4007
60 671 743 300 2689 833 830 831]
User 99 : [4210 4198 4183 4215 4194 4195 4216 4196 4197 4199 4200 4201 4202 4203
4204 4205 4206 4219 4208 4209 4211 4212 4213 4192 4191 4193 4189 4218
4190 4217 4184 4214 4186 4188 4187 4185 4207 1480 1464 3038 3039 3040
3041 3042 3043 3045 3047 3048 3049 3051]
2.201 sec
achint@achint-Lenovo-G570:~/Documents/project track/PT-Project-master$ █

```

Figure 3: 100 users, second approach

```
File Edit View Search Terminal Help
User 693 : [19418 19423 19424 19425 19426 19427 19428 19429 19430 19431 19420 19417
19433 19434 19435 19436 19437 19416 19438 19439 19440 19441 19442 19443
19422 19421 19419 19432 5935 5955 5925 5926 5922 5927 5928 5929
5930 5933 5934 5936 5939 5941 5942 5944 5945 5946 5947 5948
5949 5951]
User 694 : [19448 19452 19463 19462 19461 19460 19459 19458 19457 19456 19455 19454
19453 19451 19450 19449 19447 19446 19445 19465 19464 19467 19466 19474
19473 19472 19444 19471 19470 19469 19468 18511 18512 18513 18508 18514
18515 18516 18517 18522 18521 18509 18497 18496 18518 18498 18506 18505
18504 18499]
User 695 : [ 2688 15276 17898 19496 19495 19494 19493 19492 19491 19490 19489 19488
19487 19486 19485 19484 19483 19482 19481 19480 19479 19478 19477 19476
19475 19497 19498 11644 17897 15827 15834 2613 5560 7276 8862 1728
7238 15845 4379 15005 7320 14999 15010 15004 15003 15002 15012 15006
15001 15013]
User 696 : [ 4305 3214 3657 3656 3655 3654 3653 3652 3651 3648 3679 3646
3645 3642 3641 3658 3659 3660 3661 3663 3664 3665 3666 3668
3669 3670 3672 3673 3675 3676 3677 3678 3640 3639 3647 3637
18555 19509 19513 19512 19511 19510 19519 19508 19507 19506 19505 19518
19504 19516]
User 697 : [16860 19520 19527 19521 19522 19526 19524 19525 19523 15469 17456 1547
451 7034 15188 1836 4696 13283 13270 13281 13266 13288 13293 13287
13285 13284 13282 13289 13294 13279 13278 13280 13267 13268 13269 13271
13265 13273 13274 13276 13277 13272 8265 578 1564 1056 7855 2019
4784 2006]
User 698 : [19537 19543 19529 19530 19531 19532 19533 19534 19535 19536 19528 19538
19540 19541 19542 19544 19545 19546 19547 19548 19549 19539 19550 19551
12473 17789 8540 5335 18294 13323 9471 3686 17783 17781 17785 17786
17787 17769 17771 17784 17772 17773 17774 17775 17776 17788 17782 17780
17779 17777]
User 699 : [ 1936 5987 18243 8231 9039 3866 760 773 10565 10566 10576 10569
10574 10580 10579 10578 10577 10575 10564 10568 10561 10563 10562 10572
10571 10570 1015 10573 6143 3411 10882 5268 8701 6762 12739 16546
16545 16544 16529 16542 16541 16543 16539 16530 16531 16532 16540 16533
16538 16534]
228.718 sec
achint@achint-Lenovo-G570:~/Documents/project track/PT-Project-master$
```

Figure 4: 700 users, second approach