**List of similarity measures in collaborative filtering**

# Abstract

Recommendation system (RS) is the system which recommends items to users among many existing items/products in database. Two main approaches for RS are content-based filtering (CBF) and collaborative filtering (CF). Recommendation system is a system which recommends items to users among many existing items in database. A popular algorithm in CF is nearest neighbors (NN) algorithm which is to find out nearest neighbors of an active user and then to recommend active user items that these neighbors may like. The essence of NN algorithm is to calculate similarity measures like cosine and Pearson for determining nearest neighbors of an active rating vector. This report focuses on descripting similarity measures which are implemented in SIM framework which is available at http://sim.locnguyen.net. Note that SIM framework provides recommendation algorithms and other machine learning algorithms along with tools to evaluate and deploy them.

# 1. Introduction

Before descripting popular similarity measures, we need to concern some important conventions and notations. Let ***U*** = {*u*1, *u*2,…, *um*} be the set of users and let ***V*** = {*v*1, *v*2,…, *vn*} be the set of items. User-based rating matrix is the matrix in which rows indicate users and columns indicate items and each cell is a rating which a user gave to an item. In other words, each row in user-based rating matrix is a rating vector of a specified user. Rating vector of active user is called active user vector. Missing values (missing ratings) are denoted as question mask “?”. For example, we have (user) rating vectors *u*1 = (*r*11=1, *r*12=2, *r*13=1, *r*14=5), *u*2 = (*r*21=2, *r*22=1, *r*23=2, *r*24=4), *u*3 = (*r*31=4, *r*32=1, *r*33=5, *r*34=5), and *u*4 = (*r*41=1, *r*42=2, *r*43=?, *r*44=?). User-based rating matrix can be transposed into item-based rating matrix in which each row is a rating vector of a specified item. For example, we have (item) rating vectors *v*3 = (*r*13=1, *r*23=2, *r*33=5, *r*43=?) and *v*4 = (*r*14=5, *r*24=4, *r*34=5, *r*44=?). In general, nearest neighbors (NN) algorithm includes two steps (Torres Júnior, 2004, pp. 17-18):

1. Find out nearest neighbors of the active user by calculating similarities between active vector and other vectors. The more the similarity is, the nearer two users are. Given a threshold, users whose similarities between them and active user are equal to or larger than a threshold are considered as nearest neighbors of active user.
2. Compute predictive values for missing ratings of active vector. The computation is based on ratings of nearest neighbors and similarities calculated in step 1. The items whose predictive values are high enough are recommended to active user.

Similarity measures are calculated in step 1. Given two rating vectors *u*1 = (*r*11, *r*12,…, *r*1*n*) and *u*2 = (*r*21, *r*22,…, *r*2*n*) of user 1 and user 2, in which user 1 is considered as active user and some *rij* can be missing (empty). Let *I*1 and *I*2 be set of indices of items that user 1 and user 2 rated, respectively. Let denote intersection set of *I*1 and *I*2 and let denotes union set of *I*1 and *I*2. All items whose indices belong to are rated by both user 1 and user 2. In other words, all items whose indices belong to co-exist in vectors *u*1 and *u*2. All items whose indices belong to are rated by user 1 or user 2. Notation |x| indicates absolute value of number, length of vector, length of geometric segment, or cardinality of set, which depends on context. Let sim(*u*1, *u*2) denote the similarity of *u*1 and *u*2, which is the general notation for all similarity measures, for example, sim(*u*1, *u*2) = cos(*u*1, *u*2) denotes cosine measure. Before describing similarity measures, we should consider how to compute predictive values for missing values in step 2 of NN algorithm. Given the similarity of *u*1 and *u*2 denoted sim(*u*1, *u*2), the larger sim(*u*1, *u*2) is, the more the user 2 is near to active user 1. Hence, sim(*u*1, *u*2) is used to determine the list of neighbors of active user. Suppose NN algorithm finds out *k* neighbors of *u*1, let *N* be set of indices of *k* neighbors of *u*1. Of course, we have |*N*| = *k*. A missing value *r*1*j* of *u*1 is computed (predicted) based on ratings of nearest neighbors and similarities according to step 2 of NN algorithm (Torres Júnior, 2004, p. 18).

|  |  |
| --- | --- |
|  |  |

Where and are mean values of *u*1 and *ui*, respectively. Equation 2 above is called prediction formula or estimation formula.

Where *Ii* is the set of indices of items that user *i* rated. The missing value *r*1*j* of *u*1 can be predicted more simply as follows:

|  |  |
| --- | --- |
|  |  |

In general, similarity measure is the heart of NN algorithm because prediction formulas are based on similarity measures. Next sections mention similarity measures built in SIM framework.

# 2. Jaccard

The first measure which is described here is Jaccard because of its special feature when it does not concern magnitude of numeric rating values. Jaccard measure is ratio of cardinality of common set to cardinality of union set . It measures how much common items both users rated, which is defined as follows (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 158):

|  |  |
| --- | --- |
|  |  |

Although Jaccard is simple, it is effective within spare rating matrix which has many missing values. Another version of Jaccard is (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 158):

|  |  |
| --- | --- |
|  |  |

# 3. Cosine

Cosine which is the most popular similarity measure is specified as follows:

|  |  |
| --- | --- |
|  |  |

Where |*u*1| and |*u*2| are lengths of *u*1 and *u*2, respectively whereas *u*1•*u*2 is dot product (scalar product) of *u*1 and *u*2, respectively. If all ratings are non-negative, range of cosine measure is from 0 to 1. If it is equal to 0, two users are totally different. If it is equal to 1, two users are identical. By following the ideology of Jaccard measure, cosine measure is modified as follows:

|  |  |
| --- | --- |
|  |  |

Let *rm* be median of rating values, for example, if rating values range from 1 to 5, the median is *rm* = (1+5) / 2 = 3. The normalized cosine measure (CON) [4, p. 158] is defined as follows:

Obviously, CON measure is constrained Pearson correlation (CPC) mentioned later.

Let *vj* = (*r*1*j*, *r*2*j*,…, *rmj*) be vector of rating values that item *j* receives from *m* users, for example. The mean of *vj* is:

Adjusted cosine measure (COD) is defined as follows:

|  |  |
| --- | --- |
|  |  |

Jaccard can be combined with any measure. For instance, CosineJ is combinations of Jaccard and cosine as follows:

|  |  |
| --- | --- |
|  |  |

# 4. Pearson

Pearson correlation is another popular similarity measure besides cosine, which is defined as follows (Sarwar, Karypis, Konstan, & Riedl, 2001, p. 290):

|  |  |
| --- | --- |
|  |  |

Where and are mean values of *u*1 and *u*2, respectively.

The range of Pearson measure is from –1 to 1. If it is equal to –1, two users are totally opposite. If it is equal to 1, two users are identical. Pearson measure is sample correlation coefficient in statistics. Pearson measure has some variants. Constrained Pearson correlation (CPC) measure considers impact of positive and negative ratings by using median *rm* instead of using the means; for example, if rating values range from 1 to 5, the median is *rm* = (1+5) / 2 = 3. CPC measure is defined as follows (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 158):

|  |  |
| --- | --- |
|  |  |

The similarity will be significant if both users rated more common items. Weight Pearson correlation (WPC) measure and sigmoid Pearson correlation (SPC) measure concern how much common items are. WPC is defined as follows (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 158):

|  |  |
| --- | --- |
|  |  |

SPC is defined as follows (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 158):

|  |  |
| --- | --- |
|  |  |

Where *H* is a threshold and it is often set to be 50 (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 158).

PearsonJ is combinations of Jaccard and Pearson as follows:

|  |  |
| --- | --- |
|  |  |

Choi and Suh (Choi & Suh, 2013) proposed a so-called PC measure which is Pearson measure weighted by similarities of items. In other words, PC measure combines similarities of users and items (Patra, Launonen, Ollikainen, & Nandi, 2015, p. 4). The ideology is excellent. PC measure can be applied into any foundation measures. Each factor in PC measure is weighted by a similarity of active item and another item. Suppose it is necessary to estimate rating values of active item *k*, PC measure (Choi & Suh, 2013, p. 148) is defined as follows:

|  |  |
| --- | --- |
|  |  |

Where sim(*vk*, *vj*) is similarity of the active item *k* and item *j*. Note, sim(*vk*, *vj*) can be calculated by any measures here. The and are mean values of *u*1 and *u*2, respectively.

Experimental results proved that PC is an effective measure.

# 5. MSD

Mean squared difference (MSD) is defined as inverse of distance between two vectors. Let MAX be maximum value of ratings, MSD is calculated as follows (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 158):

|  |  |
| --- | --- |
|  |  |

Another variant of MSD is specified by some authors as follows:

|  |  |
| --- | --- |
|  |  |

MSD measure combines with Jaccard measure, which derives MSDJ measure as follows (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 158):

|  |  |
| --- | --- |
|  |  |

# 6. SRC

When rating values are converted into ranks, Spearman’s Rank Correlation (SRC) is defined as follows (Hyung, 2008, p. 39):

|  |  |
| --- | --- |
|  |  |

Where *dj* is difference between two ranks on item *j* given by user 1 and user 2.

Note, it is easy to convert ratings values to ranks. For example, suppose rating values (bins) are 5, 6, 7, 8, 9 then, we have rank 1 (for value 9), rank 2 (for value 8) , rank 3 (for value 7), rank 4 (for value 6), and rank 5 (for value 5). If user 1 rates value 9 to item *j*, we have *rank*1*j* = 1. The larger the value is, the smaller (higher) the rank is.

# 7. PIP

Ahn (Hyung, 2008) proposed a heuristic measure to solve cold-starting problem which relates to missing data in which there is not enough information to calculate similarities between rating vectors (Hyung, 2008, p. 39). The measure called PIP measure based on concept of “agreement” in rating. If both user 1 and user 2 like or dislike the same item, it is called that they have a rating “agreement” on such item. Let *r*1*j* and *r*2*j* be ratings of user 1 and user 2 on item *j*, respectively, the agreement (Hyung, 2008, p. 43) of them is defined as follows:

Note, *rm* be median of rating values, for example, if rating values range from 1 to 5, the median is *rm* = (1+5) / 2 = 3. PIP measure (Hyung, 2008, p. 42) is sum of products of triples Proximity, Impact, and Popularity.

|  |  |
| --- | --- |
|  |  |

Proximity (Hyung, 2008, p. 43) indicates similarity of two ratings, based on agreement and distance between them. The distance is increased twice as a penalty if such two ratings are not agreed.

Where *rmin* and *rmax* are minimum rating value and maximum rating value, respectively. If two ratings are agreed, their impact (Hyung, 2008, p. 43) is proportional to difference between them and rating median. If two ratings are disagreed, their impact is inverse of such difference.

Popularity (Hyung, 2008, p. 43) indicates difference between ratings given by active users and the average rating.

Note, *μj* is average rating of item *j*, which is same mean of rating values of item *j*. Experimental results proved that cold-starting problem is solved well by PIP measure (Hyung, 2008, p. 47).

# 8. PSS

Liu et al. (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 156) proposed a new similarity measure called NHMS to improve recommendation task in which only few ratings are available. Their NHMS measure (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 160) is based on sigmoid function and the improved PIP measure as PSS (*Proximity* – *Significance* – *Singularity*). PSS similarity is calculated as follows (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 160):

|  |  |
| --- | --- |
|  |  |

Where, is intersection set of *I*1 and *I*2. The proximity factor determines similarity of two ratings, based on distance between them; such distance is as less as better. The significance factor determines similarity of two ratings, based on distance from them to rating median; such distance is as more as better. The significance factor determines similarity of two ratings, based on difference between them and other ratings; such difference is as less as better. Followings are equations of these factors based on sigmoid function (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 161).

Note, *rm* be median of rating values, for example, if rating values range from 1 to 5, the median is *rm* = (1+5) / 2 = 3 whereas *μj* is rating mean of item *j*. Liu et al. (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 161) also considered the similarity between two users via URP measure as follows:

|  |  |
| --- | --- |
|  |  |

Where *μ*1 and *μ*2 are rating means of user 1 and user 2, respectively and *σ*1 and *σ*2 are rating standard deviations of user 1 and user 2, respectively.

Liu et al. (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 161) proposed a new heuristic similarity model (NHSM) as triple product of PSS measure, URP measure, and Jaccard2 measure.

|  |  |
| --- | --- |
|  |  |

In general, Liu et al. (Liu, Hu, Mian, Tian, & Zhu, 2013) aim to alleviate the problem of few rated common items via their NHSM measure. From experimental result, NHSM gave out excellent estimation.

# 9. BCF

Patra et al. (Patra, Launonen, Ollikainen, & Nandi, 2015, p. 143) proposed a new similarity measure called BCF for CF, which uses all ratings made by a pair of users. Proposed measure finds importance of each pair of rated items by exploiting Bhattacharyya (BC) similarity. The BC similarity, which is core of their own measure, measures the similarity between two distributions. So, these distributions are estimated as the number of uses rated on given item. In general, Patra et al. (Patra, Launonen, Ollikainen, & Nandi, 2015, p. 5) combined BC similarity and the local similarity where the local similarity relates to Pearson correlation. It is necessary to survey BC similarity. Bin is a terminology indicating domain of rating values, for example, if rating values range from 1 to 5, we have bins: 1, 2, 3, 4, 5. Let *m* be the number of bins, given items *i* and *j*, item BC coefficient for items is calculated as follows (Patra, Launonen, Ollikainen, & Nandi, 2015, p. 5):

|  |  |
| --- | --- |
|  |  |

Note, #*i* and #*j* are the numbers of users who rated items *i* and *j*, respectively whereas #*hi* and #*hj* are numbers of users who gave rating value *h* on items *i* and *j*, respectively. So, item BC coefficient concerns two items. In table 1.2, rating vectors of item 3 and item 4 are *v*3 = (1, 2, 5, ?) and *v*4 = (5, 4, 5, ?), respectively with note that rating values range from 1 to 5 and so we have:

According to Patra et al. (Patra, Launonen, Ollikainen, & Nandi, 2015, p. 5), user BC similarity is sum of products of item BC coefficients and local similarities as follows:

|  |  |
| --- | --- |
|  |  |

The local similarity is calculated as a part of constrained Pearson coefficient (CPC) as follows:

Note, *rm* be median of rating values, for example, if rating values range from 1 to 5, the median is *rm* = (1+5) / 2 = 3. Patra et al. (Patra, Launonen, Ollikainen, & Nandi, 2015, p. 5) proposed Bhattacharyya similarity in CF (BCF) as sum of user BC similarity and Jaccard measure as follows:

|  |  |
| --- | --- |
|  |  |

# 10. MMD

Suryakant and Mahara (Suryakant & Mahara, 2016) proposed a so-called Cosine-Jaccard-Mean Measure of Divergence (CjacMD) based on Mean Measure of Divergence (MMD) to solve the problem of sparse rating matrix. Because MMD measure takes advantages of statistical aspects, it can alleviate sparsity. MMD focuses on personal habits which are ignored by nonstatistical measures (Suryakant & Mahara, 2016, p. 453). Recall that bin is a terminology indicating domain of rating values, for example, if rating values range from 1 to 5, we have bins: 1, 2, 3, 4, 5. Let *X* = (*x*1, *x*2,…, *xb*) and *Y* = (*y*1, *y*2,…, *yb*) be count vectors of user 1 and user 2, respectively where *xj* (*yj*) is the number of items to which user 1 (user 2) gives bin *j* with note that *b* is the number of bins. For example, rating vectors of user 1 and user 2 in table 1.1 are *u*1 = (1, 2, 1, 5) and *u*2 = (2, 1, 2, 4), respectively with note that rating values ranges from 1 to 5. We have:

MMD measure is defined as follows (Suryakant & Mahara, 2016, p. 453), (Harris & Sjøvold, 2018, p. 87):

|  |  |
| --- | --- |
|  |  |

Where *θ*1\* and *θ*2\* are Grewal’s transformations (Harris & Sjøvold, 2018, p. 85) of *X* and *Y*, respectively.

In fact, CjacMD (Suryakant & Mahara, 2016, p. 453) combines three other measures such as cosine, Jaccard, and MMD together.

|  |  |
| --- | --- |
|  |  |

Experimental result proved that CjacMD model is effective similarity model.

# References

Choi, K., & Suh, Y. (2013, January). A new similarity function for selecting neighbors for each target item in collaborative filtering. (H. Fujita, & J. Lu, Eds.) *Knowledge-Based Systems, 37*(2013), 146-153. doi:10.1016/j.knosys.2012.07.019

Harris, E. F., & Sjøvold, T. (2018, September 7). Calculation of Smith's Mean Measure of Divergence for Intergroup Comparisons Using Nonmetric Data. (E. F. Harris, Ed.) *Dental Anthropology: A Publication of the Dental Anthropology Association, 17*(3), 83-93. doi:10.26575/daj.v17i3.152

Hyung, J. A. (2008, January 2). A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem. (W. Pedrycz, Ed.) *Information Sciences, 178*(1), 37-51. doi:10.1016/j.ins.2007.07.024

Liu, H., Hu, Z., Mian, A., Tian, H., & Zhu, X. (2013, November 20). A new user similarity model to improve the accuracy of collaborative filtering. (H. Fujita, & J. Lu, Eds.) *Knowledge-Based Systems, 56*(2014), 156-166. doi:10.1016/j.knosys.2013.11.006

Patra, B. K., Launonen, R., Ollikainen, V., & Nandi, S. (2015, March 1). A new similarity measure using Bhattacharyya coefficient for collaborative filtering in sparse data. (H. Fujita, & J. Lu, Eds.) *Knowledge-Based Systems, 82*, 163-177. doi:10.1016/j.knosys.2015.03.001

Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001). Item-based Collaborative Filtering Recommendation Algorithms. *Proceedings of the 10th international conference on World Wide Web* (pp. 285-295). Hong Kong: ACM. doi:10.1145/371920.372071

Suryakant, & Mahara, T. (2016, January). A New Similarity Measure Based on Mean Measure of Divergence for Collaborative Filtering in Sparse Environment. (R. K. Venugopal, R. Buyya, & M. L. Patnaik, Eds.) *Procedia Computer Science, 89*(2016), 450-456. doi:10.1016/j.procs.2016.06.099

Torres Júnior, R. (2004). *Combining Collaborative and Content-based Filtering to Recommend Research Paper.* Universidade Federal do Rio Grande do Sul, Programa de Pós Graduação em Educação. Porto Alegre: Universidade Federal do Rio Grande do Sul. Retrieved from http://www.lume.ufrgs.br/bitstream/handle/10183/5887/000432990.pdf;sequence=1