02525: Movie recommendation with collaborative filtering (Week 2)

Deena Francis

Postdoc, Section for Cognitive Systems (CogSys),

DTU Compute, Technical University of Denmark (DTU)

email: dfra@dtu.dk

Overview

- Recap
- Collaborative filtering - why it works, issues
- Item based filtering
- Evaluating the performance
- Analysis of top 50 IMDB movies recommendation
- Information about report- deadline November 9, 2022.
- Exercise 2

Last week ...

- Collaborative filtering:
  - Collection of algorithms that **predict** ratings based on **similarities**.
  - The “intelligence" is in having the computer **automatically** come up with the recommendation.
  - In addition, it would get better and better at making recommendations the **more data** it sees.
  - Assumptions:
    - Users like movies that other similar users like (user-based)
    - Users like movies that are similar to movies they already like (item-based)

Last week ...

- The data
  - Ratings matrix: N = Number of unique users/individuals, M = Number of unique movies

$$\mathbf{R} = \begin{bmatrix} R_{11} & R_{12} & \dots & R_{1M} \\ R_{21} & R_{22} & \dots & R_{2M} \\ \vdots & \vdots & \vdots & \vdots \\ R_{N1} & R_{N2} & \dots & R_{NM} \end{bmatrix}$$

- Contains values in some ratings scheme range. For eg: $0 - 5$.
- Can have missing values.

Last week ...

- Similarity measures
  - Manhattan similarity:

$$S_{Manhattan}(a,b) = \frac{1}{1 + \sum_{i=1}^{M} |R_{a,i} - R_{b,i}|}$$

  - Euclidean similarity:

$$S_{Euclidean}(a,b) = \frac{1}{1 + \sqrt{\sum_{i=1}^{M} (R_{a,i} - R_{b,i})^2}}$$

  - Pearson's similarity:

$$S_{Pearson}(a,b) = \frac{\frac{1}{M}\sum_{i=1}^{M} (R_{a,i} - \bar{R}_a)(R_{b,i} - \bar{R}_b)}{\sqrt{\frac{1}{M}\sum_{i=1}^{M} (R_{a,i} - \bar{R}_a)^2} \cdot \sqrt{\frac{1}{M}\sum_{i=1}^{M} (R_{b,i} - \bar{R}_b)^2}}$$

Last week ...

Similarity matrix

$$\mathbf{S} = \begin{bmatrix} S_{11} & S_{12} & \dots & S_{1N} \\ S_{21} & S_{22} & \dots & S_{2N} \\ \vdots & \vdots & \vdots & \vdots \\ S_{N1} & S_{N2} & \dots & S_{NN} \end{bmatrix}$$

- Is a symmetric matrix
- Diagonals are 1s.
- In the case of Euclidean and Manhattan similarity
    - entries will be in range $[0, 1]$.
    - always defined.
- In the case of Pearson similarity
    - the entries will be in the range $[-1, 1]$
    - may be *undefined* when one user has constant rating.

Last week ...

Clarification in notation:

$$\sum_{b \in \mathit{KNN(a)}} R_{b,m}$$

- denotes the sum of ratings of user b (varying) for movie m (fixed).
- b is a variable denoting a nearest neighbor (KNN) of user a,

$$b = \{b_1, b_2, ..., b_k\}$$

  where k is the number of neighbors we chose to use, and $b_1, b_2, ..., b_k$ are the k neighbors of user a.
- The above sum is in fact...

$$R_{b_1,m} + R_{b_2,m} + R_{b_3,m} + ...R_{b_k,m}$$

Last week ...

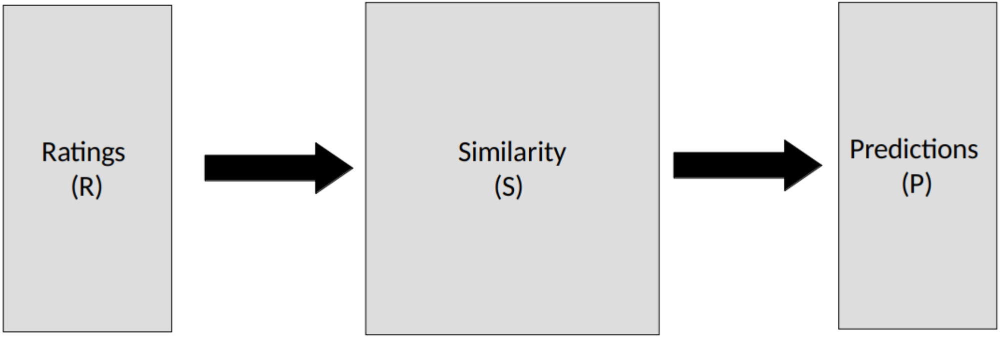
Prediction method	Meaning
$P_{\textsf{AVG}}(a,m) = \frac{1}{K}\sum_{b\in \textsf{KNN}(a)} R_{b,m}$	Average value of ratings of user a's k nearest neighbors
$P_{\textsf{WAVG}}(a,m) = \sum_{b\in \textsf{KNN}(a)} w_{a,b}R_{b,m}$	Average value of ratings of user a's k nearest neighbors, each value multiplied by an importance score
$P_{\textsf{WAC}}(a,m) = \bar{R}_a + \sum_{b\in \textsf{KNN}(a)} w_{a,b}(R_{b,m} - \bar{R}_b)$	Weighted average value of ratings of user a's k nearest neighbors with adjustments

Table: Prediction

where

$$w_{a,b} = \frac{S_{a,b}}{\sum_{b\in \textsf{KNN}(a)} S_{a,b}}$$

Last week ...



Why does collaborative filtering work?

- "Wisdom of the crowds"
- "Law of large numbers"

Why does collaborative filtering work?

- "Wisdom of the crowds"
  - Diversity of opinions
  - Independent assessments
  - Aggregation of knowledge (using the average rating)

Random variables

- Random variable: A function that maps values arising from a *random* experiment to a number.
  - Eg: Experiment: roll of a fair six-sided die, random variable: value of die obtained, X with possible values: $\{1, 2, 3, 4, 5, 6\}$.
  - Eg: Experiment: Toss of a fair coin, Random variable: $X = 1$ if HEADS, $X = 0$ if tails.
- Probability distribution: It assigns a probability to all values of the random variable.
  - Eg: $p(X = 1) = \frac{1}{6}$, $p(X = 2) = \frac{1}{6}$, $p(X = 3) = \frac{1}{6}$, $p(X = 4) = \frac{1}{6}$, $p(X = 5) = \frac{1}{6}$, $p(X = 6) = \frac{1}{6}$.

Random variables and expectation

- Expectation: Weighted average of the values taken by the random variable, where weight is the probability of the occurrence of the value.

$$\mathbb{E}[X] = \sum_i p(X_i) X_i$$

where $p(X_i)$ is the probability of the occurrence of X_i, where $X = [X_1, X_2, ...]$.

- X = outcome of a roll of a fair six-sided die $= \{1, 2, 3, 4, 5, 6\}$
- $p(X=1) = \frac{1}{6}$, $p(X=2) = \frac{1}{6}$, $p(X=3) = \frac{1}{6}$, $p(X=4) = \frac{1}{6}$, $p(X=5) = \frac{1}{6}$, $p(X=6) = \frac{1}{6}$
- $\mathbb{E}[X] = 1 \cdot \frac{1}{6} + 2 \cdot \frac{1}{6} + 3 \cdot \frac{1}{6} + 4 \cdot \frac{1}{6} + 5 \cdot \frac{1}{6} + 6 \cdot \frac{1}{6} = 3.5$

Sample mean and population mean (Expectation)

- *Population*: A potentially infinite set of items or events from an experiment.
  - Eg: The set of all possible outcomes of throw of a fair die.
- *Sample*: A random selection of elements of a population.
  - Eg: The outcomes of 5 throws of a fair die.
- Population mean or expectation:

$$\mathbb{E}[X] = \sum_i p(X_i) X_i$$

- Sample mean or average:

$$\bar{X} = \frac{1}{N} \sum_{i=1}^{N} X_i$$

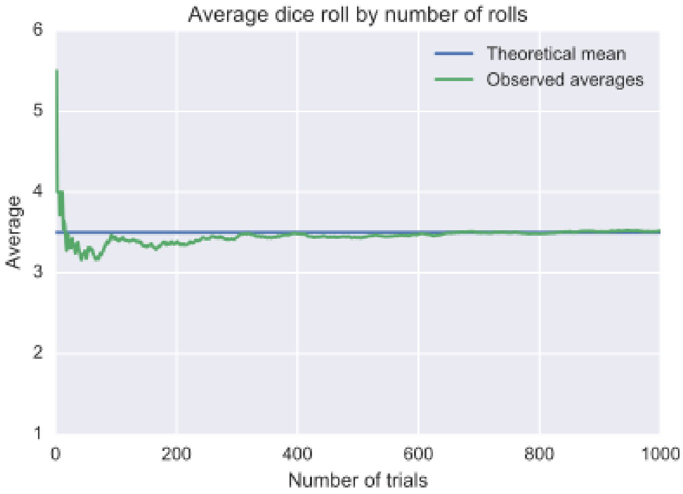
Law of large numbers

- Repeating an experiment many times will result in the sample mean moving towards the population average / expected value.

$$\lim_{N \to \infty} \frac{1}{N} \sum_{i=1}^{N} X_i = \mathbb{E}[X]$$

Sample mean and expectation

- $\mathbb{E}[X] = 1 \cdot \frac{1}{6} + 2 \cdot \frac{1}{6} + 3 \cdot \frac{1}{6} + 4 \cdot \frac{1}{6} + 5 \cdot \frac{1}{6} + 6 \cdot \frac{1}{6} = 3.5$
- Sample: The outcomes of 5 throws of a fair die, $\{3, 3, 3, 4, 5\}$
- $\bar{X} = \frac{3+3+3+4+5}{5} = 3.6$



Connecting law of large numbers and collaborative filtering

- Random variable: Ratings of the **true KNN** of user a.
- Probability distribution: The true distribution of this variable is "unknown".
- Expectation or true mean: The mean of the ratings of the true KNN.
- Sample mean: The average rating of **estimated KNN** of user a.
- By law of large numbers, when the number of samples (users/movies) increases, the predicted (estimated) rating (average of neighbor's ratings) becomes close to the true mean (expectation).

$$\text{As } N \to \infty, \text{Average of ratings of estimated KNN} \to \text{Mean of ratings of true KNN}$$

A 5 minute discussion with neighbors

- If there are few ratings per user, user-based collaborative filtering will be poor because ...
- It is more expensive in terms of computing power to use user-based collaborative filtering because ...
- We would have to (re)calculate $\mathbf{S}$ more often, because...

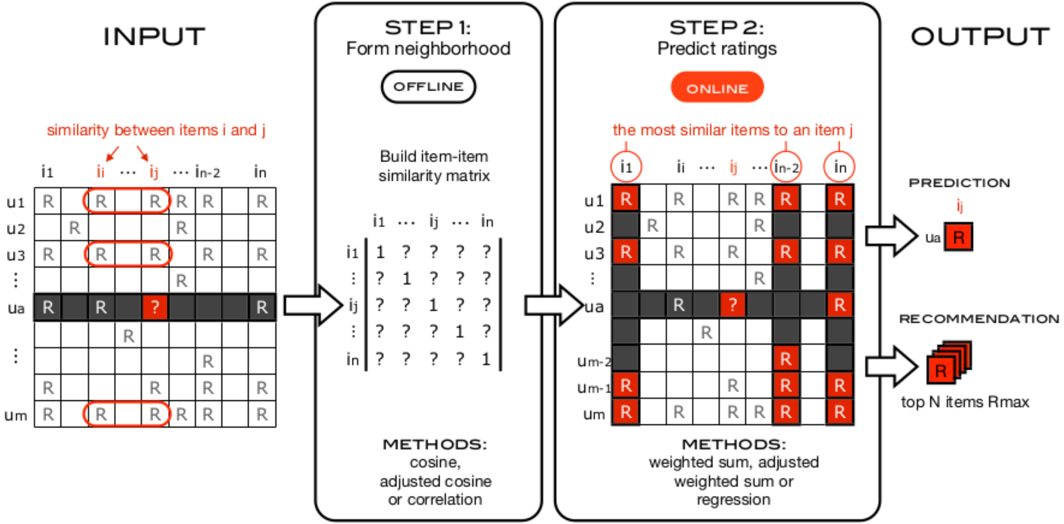
A 5 minute discussion with neighbors

- If there are few ratings per user, user-based collaborative filtering will be poor because ...
  - there will be fewer points to compare 2 users. Typically there are several ratings per film than per user.
- It is more expensive in terms of computing power to use user-based collaborative filtering because ...
  - there are typically more users than movies.
- We would have to (re)calculate $\mathbf{S}$ more often, because...
  - new users come more often than new movies

Item-based collaborative filtering

- Compute the similarity between movies instead of users.
    - $\mathbf{S}$ needs to be recomputed less frequently.
    - $\mathbf{S}$ is a smaller matrix ($M \times M$).

Item-based collaborative filtering



Evaluating the recommender system

- Measuring performance
    - How do you quantify the performance of your recommender system?
- Practical considerations:
    - How do you choose the number of neighbors, the similarity measure and the prediction method? (*Choosing the parameters*)

5 minute discussion with neighbors

Measuring performance

- How do you quantify (measure) the performance of your recommender system?

Hint: Remember that we have the ratings matrix (ground truth) and the output of the recommender system, a prediction matrix:

$$\mathbf{R} = \begin{bmatrix} R_{11} & R_{12} & \dots & R_{1M} \\ R_{21} & R_{22} & \dots & R_{2M} \\ \vdots & \vdots & \vdots & \vdots \\ R_{N1} & R_{N2} & \dots & R_{NM} \end{bmatrix} \qquad \mathbf{P} = \begin{bmatrix} P_{11} & P_{12} & \dots & P_{1M} \\ P_{21} & P_{22} & \dots & P_{2M} \\ \vdots & \vdots & \vdots & \vdots \\ P_{N1} & P_{N2} & \dots & P_{NM} \end{bmatrix}$$

Quantifying performance - Error of prediction

- How do you quantify (measure) the performance of your recommender system?
    - An error measure

$$e = \textit{Predicted value} - \textit{Ground truth}$$
$$e_{i,j} = P_{i,j} - R_{i,j}$$

    - Compute this error for all values of i and j and *aggregate* it.

Quantifying performance - Error of prediction

- Mean Absolute Error (MAE):

$$\begin{aligned} MAE &= \frac{1}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}|e_{i,j}| \\ &= \frac{1}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}|P_{ij} - R_{ij}| \end{aligned}$$

- Here $| \quad |$ denotes the absolute value (value without the sign).
- Double sum $\sum_{i=1}^{N}\sum_{j=1}^{M}$: first evaluate the inner summation, then the outer summation.

It's simply the average error between $\mathbf{R}$ and $\mathbf{P}$.

$$
\begin{aligned}
\mathit{MAE} &= \frac{1}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}|P_{ij} - R_{ij}| \\
&= \frac{1}{NM}\sum_{i=1}^{N}(|P_{i,1} - R_{i,1}| + |P_{i,2} - R_{i,2}| + ... + |P_{i,M} - R_{i,M}|) \\
&= \frac{1}{NM}(|P_{1,1} - R_{1,1}| + |P_{1,2} - R_{1,2}| + ... + |P_{1,M} - R_{1,M}|) \\
&+ (|P_{2,1} - R_{2,1}| + |P_{2,2} - R_{2,2}| + ... + |P_{2,M} - R_{2,M}|) \\
&+ ... \\
&+ (|P_{N,1} - R_{N,1}| + |P_{N,2} - R_{N,2}| + ... + |P_{N,M} - R_{N,M}|)
\end{aligned}
$$

Results of top 50 IMDB movie ratings

Summary

- Total number of users (with ratings) $= 43$
- Total number of (rated) movies $= 48$
- Similarity measure: Pearson
- $K = 6$
- Prediction method: Weighted average with corrections P_{WAC}

Results - Your ratings for 50 films

User	MAE	Recommended film 1:	P_{WAC}
Würtz, Fabian Schiøler	0.0	Casablanca (1942)	4.0
Fan, Ivan	0.3	The Intouchables (2011)	5.0
Knudsen, Jacob Brünnich	0.48	The Intouchables (2011)	3.73
Henriksen, Rasmus Niels	0.49	Spirited Away (2001)	4.39
Nielsen, Aske Funch Schrøder	0.5	The Green Mile (1999)	3.78
Kjærager, Charlie Christian	0.5	Forrest Gump (1994)	4.4
Nickel, Philip Korsager	0.54	Spirited Away (2001)	3.78
Larsen, Anton Hermann Daugaard	0.56	The Shawshank Redemption (1994)	4.49
Vølund, Gertrud	0.58	The Shawshank Redemption (1994)	4.15
Nielsen, Mathilde Philbert	0.59	Psycho (1960)	4.38
Ross-Rønnow, Erik Jonathan	0.61	The Departed (2006)	4.09
Hadzimahovic, Harris Nielsen	0.62	Inception (2010)	4.77
Tauris, Sofie Louise	0.64	Parasite (2019)	4.61
Pilehave, August Ørnstrup	0.64	Parasite (2019)	4.14
Bjerregaard, Christian Lundgaard	0.65	Fight Club (1999)	4.52

Table: Top 15 performance (P_{WAC}, Pearson similarity, $K = 6$) error measured using MAE.

5 minute discussion with neighbors

Practical considerations

- How do you choose the number of neighbors, the similarity measure and the prediction method? (Assume you have do not have any prior knowledge.)

Choosing the parameters

- How do you choose the number of neighbors, the similarity measure and the prediction method? (Assume you do not have any prior knowledge.)

- **Answer**: Grid-search.
  - Decide on a set of prediction method, similarity measures and the number of neighbors.
  - For *each combination* of the above compute MAE.
  - Choose the combination that gives least error (MAE).

Choosing the parameters

- The number of tests we have to do is therefore:

$$N_P \cdot N_S \cdot N_K$$

  where N_P is the number of prediction algorithms, N_S is the number of similarity measures and N_K is the number of neighbors.

Prediction method	Similarity measure	$K=1$	$K=2$	$K=3$	$K=4$	$K=5$	$K=6$	$K=7$	$K=8$	$K=9$
P_A	M	1.19	1.06	1.03	1.0	0.99	0.97	0.94	0.95	0.95
	E	1.27	1.05	1.01	0.97	1.0	0.98	0.94	0.93	0.93
	P	0.72	0.76	0.75	0.78	0.81	0.8	0.78	0.79	0.8
P_{WA}	M	3.42	3.36	3.33	3.31	3.31	3.29	3.29	3.28	3.27
	E	3.43	3.33	3.31	3.25	3.23	3.23	3.22	3.22	3.21
	P	2.08	2.02	2.02	2.0	1.99	1.98	1.97	1.97	1.97
P_{WAC}	M	0.76	0.75	0.78	0.77	0.77	0.77	0.77	0.77	0.77
	E	0.77	0.76	0.78	0.77	0.77	0.77	0.77	0.77	0.77
	P	0.75	0.71	0.7	0.72	0.72	0.72	0.72	0.73	0.73

Table: Grid search: MAE for various prediction methods and similarities. M = Manhattan, E = Euclidean, P = Pearson, $K = 1, ..., 9$, user-based collaborative filtering.

Summary

- Collaborative filtering - why and how.
- User-based and Item-based.
- Similarity measures.
- Ratings predictions based on similarities.
- Evaluating prediction using error measure.

Report

- Instructions for making the report on Learn: *02525-Movie-recommendations-Report_instructions.pdf*
- Deadline: 9 November 2022 at 23:59.

- Complete Exercise 2 (*02525-Movie-recommendations-Exercise2.pdf* on Learn) and begin the report.
- Hope you think it was exciting :-) Good luck with the report and exam!

Thank you!