

# 특이값 분해 시의 코사인 유사도 변화에 대한 추정

유 찬 우\*

## Estimation of Cosine Similarity Change in Singular Value Decomposition

Chanwoo Yoo\*

### 요 약

많은 개인화 추천을 위한 머신러닝 모델이 추천 후보를 만들어내기 위해 근접 이웃 기법을 사용한다. 이 경우 사용자 또는 품목의 차원이 보통 매우 크기 때문에, 데이터를 나타내는 벡터를 그대로 사용하기는 어렵고 차원이 축소된 임베딩을 만들어낼 필요가 있다. 특이값 분해와 같은 행렬 분해 기법을 이용해 차원을 줄이게 되면 본래의 데이터 사이의 코사인 유사도에 손실이 일어나게 되는데, 그 경우 상대적인 유사도의 순위가 바뀌어서 추천 목록이 부정확해질 수 있다. 본 논문에서는 특이값 분해를 이용한 차원 축소 시 코사인 유사도의 상대적인 순위가 보존되는 정도를 측정할 수 있는 지표를 상대 순위 정확도로 정의하고, 특이값을 이용하여 상대 순위 정확도를 추정할 수 있는 방법을 제안하였다.

### Abstract

Many machine learning models for personalized recommendation use nearest neighbor methods to generate recommendation candidates. In this case, the dimension of users or items is usually very large, it is hard to use user and item vectors as they are, so it needs to generate embeddings of which the dimension reduced. When dimensionality reduction is done by matrix factorization methods like singular value decomposition, there is loss in cosine similarity between data points. The loss may results in inaccurate recommendation candidates because relatively farther neighbor may be regarded as closer one. In this paper, I defined a metric that measures the degree of preservation of relative rankings of cosine similarity as relative ranking precision. I also proposed a formula to estimate relative ranking precision with singular values.

### Keywords

personalized recommendation, dimensionality reduction, singular value decomposition, cosine similarity, relative ranking precision

---

\* 라인플러스 Data Science Dev Lead  
- ORCID: <http://orcid.org/0000-0003-2474-7355>

• Received: Feb. 02, 2020, Revised: Mar. 06, 2020, Accepted: Mar. 09, 2020  
• Corresponding Author: Chanwoo Yoo  
LINE Plus, Data Science Dev, 11th floor, 42, Hwangsaeul-ro 360 beon-gil,  
Bundang-gu, Seongnam-si, Gyeonggi-do, Republic of Korea  
Tel.: +82-1544-9430, Email: [chanwoo.yoo@gmail.com](mailto:chanwoo.yoo@gmail.com)

## 1. 서 론

Netflix Prize[1]에서의 성공 이후, 행렬 분해(Matrix factorization) 기법[2]은 개인화 추천(Personalized recommendation) 머신러닝 알고리즘의 베이스라인으로 널리 사용되고 있다. 특히 산업계에서는 사용자(User)-품목(Item)이 각각 행과 열이 되고, 품목에 대한 평점(Rating)이 행렬의 원소가 되는 계획 행렬(Design matrix)을 분해해서 얻어진 사용자 또는 품목 임베딩을 가지고, 사용자가 선호한 품목과 유사한 품목을 추천하거나, 목표 사용자와 유사한 사용자를 찾아내어 유사 사용자가 선호한 품목을 추천하는 근접 이웃(Nearest neighbor) 방식으로 추천의 후보군을 생성하는 경우가 많다[3]-[5]. 이 경우에 임베딩 사이의 유사성을 계산하기 위해 개별 임베딩의 크기에 영향을 받지 않는 코사인 유사도(Cosine similarity)[6]를 주로 사용하게 된다.

원본 계획 행렬에서의 사용자 벡터 또는 품목 벡터의 차원은 매우 크기 때문에 그대로 유사도 비교에 사용하기는 어렵고, 행렬 분해 기법을 사용할 때 차원을 축소하여 임베딩의 차원을 줄이게 된다. 예를 들어 특이값 분해(Singular Value Decomposition, SVD)를 사용하여 행렬을 분해하는 경우, 상대적으로 큰 특이값에 대응되는 특이벡터(Singular vector)를 남기고 나머지를 버림으로써 차원을 축소할 수 있다.

특이값 분해를 통해 차원을 축소할 경우에 생기는 행렬의 프로비니어스 놈(Frobenius norm)의 차이는 에카르트-영 정리(Eckart-Young theorem)[7]나 존슨-린든스트라우스 보조정리(Johnson-Lindenstrauss lemma)[8]를 통해 알려져 있다. 하지만, 개인화 추천 머신러닝 모델을 생성 시 특이값 분해를 사용하는 경우의 관심사는 프로비니어스 놈의 차이보다는 원본 계획 행렬의 개별 데이터(사용자 또는 품목) 사이의 코사인 유사도에 얼마나 손실이 일어나느냐는 것이다. 예를 들어, 원본 계획 행렬에서 A 품목이 상대적으로 C 보다 B 품목과 코사인 유사도가 높았을 때, 차원 축소 후에 C와의 유사도가 B 보다 높아졌다면, 본래의 관계를 왜곡하는 추천을 하게 되기 때문에, 좋은 추천을 위해서는 차원 축소 과정에서 원본 계획 행렬의 개별 데이터 사이의 코사인 유사도

를 최대한 보존하는 것이 바람직하다. 하지만 차원이 축소되는 특이값 분해(Reduced SVD)시 데이터 사이의 코사인 유사도가 어느 정도 보존되는지는 기존 정리들로 파악이 어렵다.

본 논문에서는 차원이 축소되는 특이값 분해시 데이터 사이의 코사인 유사도와 특이값 사이에 상관관계가 있다는 것을 실험을 통해 보이고, 특이값을 이용해 코사인 유사도 손실에 대해 추정하는 방법을 제시하고자 한다. 본 논문의 구성은 다음과 같다. 2장에서는 특이값 분해를 이용한 차원 축소시의 오차에 대해 알려진 기존 내용들을 정리하고, 3장에서는 차원 축소 전후의 코사인 유사도 보존 정도를 측정하기 위한 지표를 정의한다. 4장에서는 특이값을 이용하여 코사인 유사도 보존 정도를 추정할 수 있는 방법을 제안하고 개인화 추천 머신러닝을 위한 공개 데이터셋을 이용하여 해당 방법의 정확도를 평가한 후, 5장에서 결론을 맺는다.

## II. 특이값 분해를 이용한 차원 축소시의 오차

에카르트-영 정리는  $B$ 가 계수  $r$ 의 행렬이고,  $A_r$ 이  $A_r = \sum_{i=1}^r \sigma_i u_i v_i^T$ 와 같이 가장 큰 특이값  $r$ 개에 대응되는 특이벡터로 분해되는 행렬일 때  $\|A - B\| \leq \|A - A_k\|$ 가 성립한다는 것이다. 정리의 의미는 행렬  $A$ 에 가장 근사한, 즉 프로비니어스 놈의 차이를 최소화하는  $r$  계수의 행렬  $A_r$ 이  $\sum_{i=1}^r \sigma_i u_i v_i^T$ 와 같이 나타난다는 것이고, 계수(Rank)가 고정되었을 때 가장 근사한 행렬을 구하는 방법을 제시한다고 할 수 있다.

존슨-린든스트라우스 보조정리[8]는  $0 < \epsilon < 1$ ,  $R^N$ 에 속하는  $m$ 개의 점들의 집합  $X$ ,  $n > 8 \ln(m)/\epsilon^2$ 과 같이 주어졌을 때, 모든  $(u, v) \in X$ 에 대해 식 (1)을 만족하는 선형 변환  $f: R^N \rightarrow R^n$ 가 존재한다는 것이다.

$$(1 - \epsilon) \|u - v\|^2 \leq \|f(u) - f(v)\|^2 \leq (1 + \epsilon) \|u - v\|^2 \quad (1)$$

존슨-린든스트라우스 보조정리는 에카르트-영 정

리와는 반대로 목표로 하는 오차의 범위가 주어졌을 때 해당 오차를 벗어나지 않는 범위에서 차원 축소를 적용하려면 필요한 최소 계수를 알 수 있게 해 준다. 또한 이러한 차원 축소는 주로 랜덤 투영(Random projection) 기법[9]에 의해서 이루어진다.

위와 같은 정리들은 주로 스펙트럴 놈(Spectral norm) 또는 프로비니어스 놈에 대한 것으로 코사인 유사도의 보존에 대한 것은 아니다. 또한 코사인 유사도의 평균 제곱 오차(Mean squared error)를 프로비니어스 놈으로부터 유도할 수 있다 해도 추천 분야에 있어서는 큰 의미가 없는데, 그 이유는 상대적인 유사도의 순위가 보존되는지가 중요하기 때문이다. 개별 유사도에 변동이 생기더라도 유사도의 상대적인 순위에만 변동이 없다면, 근접 이웃 방법을 통해서 뽑는 추천 후보군의 목록에는 변화가 없게 되기 때문에 상대적인 순위가 중요하다. 따라서 코사인 유사도의 상대 순위가 얼마나 보존되는지에 대한 측정 및 추정이 필요한데, 기존에 이와 같은 주제를 다룬 연구를 찾기 어려웠다. 본 논문에서는 먼저 3장에서 코사인 유사도의 상대 순위 보존을 측정하기 위한 지표를 정의한 뒤, 4장에서 특이값을 이용해 이와 같은 지표를 추정하는 방법을 제시한다.

### III. 상대 순위 정확도

차원 축소 과정에서 데이터 사이의 코사인 유사도가 어느 정도 보존되는지 측정하기 위해서 본 논문에서는 상대 순위 정확도(Relative ranking precision)라는 것을 정의하였다. 행렬  $A \in R^{m \times n}$ 가 특이값 분해에 의해  $A = UV^T$ 와 같이 분해된다고 가정하자.  $a_m^T$ 는 개별 데이터를 나타내고,  $v_n$ 은  $V$ 의 열벡터를 나타낸다.

$$p_r = \frac{\sum_{(a_i^T, a_j^T, a_k^T) \in S} I(\text{equality}_0 = \text{equality}_1)}{N} \quad (4)$$

where

$$\text{equality}_0 = \text{sgn} \left( \frac{a_i^T a_j}{\|a_i\| \|a_j\|} - \frac{a_i^T a_k}{\|a_i\| \|a_k\|} \right)$$

$$\text{equality}_1 = \text{sgn} \left( \frac{a_i^T V_r V_r^T a_j}{\|a_i^T V_r\| \|a_j^T V_r\|} - \frac{a_i^T V_r V_r^T a_k}{\|a_i^T V_r\| \|a_k^T V_r\|} \right)$$

$$A = \begin{bmatrix} -a_1^T \\ -a_2^T \\ \vdots \\ -a_m^T \end{bmatrix}, \quad V = \begin{bmatrix} | & | & & | \\ v_1 & v_2 & \vdots & v_n \\ | & | & & | \end{bmatrix} \quad (2)$$

개별 데이터의 차원 축소는  $V_r$ 이 가장 큰  $r$ 개의 특이값에 해당하는 특이벡터로 이루어진 행렬이라고 할 때,  $a_i^T V_r \in R^{1 \times r}$ 과 같이 나타낼 수 있다.  $r = n$ 일 때,  $V_r$ 은  $V$ 와 같아지고,  $V$ 는 직교행렬(Orthogonal matrix)이기 때문에,  $VV^T = I$ 이고  $\|a_i^T V\| = \|a_i\|$ 이 되어, 개별 데이터 사이의 코사인 유사도는 식 (3)과 같이 변화가 없게 된다.

$$\frac{a_i^T V V^T a_j}{\|a_i^T V\| \|a_j^T V\|} = \frac{a_i^T a_j}{\|a_i\| \|a_j\|} \quad (3)$$

하지만,  $r < n$ 인 경우 차원 축소가 이루어지게 되면, 개별 데이터 사이의 코사인 유사도에는 변화가 생긴다. 추출된  $N$ 개의 데이터 표본의 집합을  $(a_i^T, a_j^T, a_k^T) \in S$ 라고 할 때, 상대 순위 정확도는 식 (4)과 같이 정의할 수 있다.  $I$ 는 지시 함수(Indicator function)를,  $\text{sgn}$ 은 값의 부호를 나타낸다.

만약 차원 축소 전에  $a_i^T$ 와  $a_j^T$  사이의 코사인 유사도가  $a_i^T$ 와  $a_k^T$  사이의 유사도보다 크다면, 차원 축소 후에도 이와 같은 대소 관계가 유지되는 것이 바람직하다. 식 (4)의 값은 표본 삼중항(Triplet) 중 차원 축소 전후에 동일한 대소 관계를 가지는 삼중항의 비율을 측정한다고 볼 수 있다. 개인화 추천 머신러닝 문제에서, 코사인 유사도 오차의 합보다 이 지표가 더 중요한 이유는 추천 문제는 기본적으로 순위 문제이기 때문이다.

예를 들어, 상대적인 코사인 유사도 순위가 그대로 유지될 경우, 근접 이웃 방법에 의해 선택되는 목록은 전혀 변하지 않는다.

차원 축소 후의 코사인 유사도 표현식에서,  $V_r V_r^T$ 는  $V_r$  행렬의 열 공간(Column space)으로의 투영 행렬(Projection matrix)을 나타낸다. 만약  $a_i$ 가 이미  $V_r$ 의 열 공간 안에 있다면, 이와 같은 투영은  $a_i$ 를 변화시키지 않는다.  $a_i$ 가  $V_r$ 의 열 공간 안에 있지는 않지만 충분히 가깝다면 투영 행렬에 의한 변화는 작게 나타난다. 하지만 특이값 분해에 의한 차원 축소 시, 이와 같은 변화가 모든 개별 데이터에 대해서 이루어지기 때문에, 코사인 유사도의 상대적인 순위에 어떤 식으로 변화를 일으키는지는 분석적으로 파악하기 어렵다. 그래서 본 논문에서는 특이값과 상대 순위 정확도 사이에 어떤 관계가 있는지를 실험적으로 파악하려고 시도하였다.

#### IV. 특이값을 이용한 상대 순위 정확도의 추정

본 논문에서는 Movielens[10], BGG[11], Jester[12], Datafinti[13]와 같이 사용자들이 품목에 대해 평점을 매긴 공개 데이터셋을 계획 행렬로 사용하여, 특이

값 분해를 이용한 차원 축소를 사용자 벡터에 적용 후, 상대 순위 정확도가 얼마나 떨어지는지에 대한 실험을 수행하였다. 그리고 특이값을 이용해 상대 순위 정확도를 추정하기 위한 다양한 시도를 해 본 결과 (5)와 같은 식이 실제 정확도를 매우 가깝게 근사한다는 것을 발견하였다. 식 (5)에서  $\hat{p}_r$ 은 상대 순위 정확도의 추정치를,  $r$ 은 데이터를 축소할 계수의 크기를,  $N$ 은 특이값의 총 수를,  $\sum_{k=1}^r \sigma_k$ 은 가장 큰  $r$ 개의 특이값의 합을 나타낸다.

$$\hat{p}_r = \frac{\log \sum_{k=1}^r \sigma_k}{\log \sum_{k=1}^N \sigma_k} \quad (5)$$

그림 1은 공개 데이터셋에 등장하는 사용자 사이의 코사인 유사도에 대한 상대 순위 정확도를 나타낸 것으로, x축은 차원 축소시의 계수를, y 축은 상대 순위 정확도를 나타낸다. 동그라미는 상대 순위 정확도를, ×표는 식 (5)를 사용한 상대 순위 정확도의 추정치  $\hat{p}_r$ 을 나타낸다.

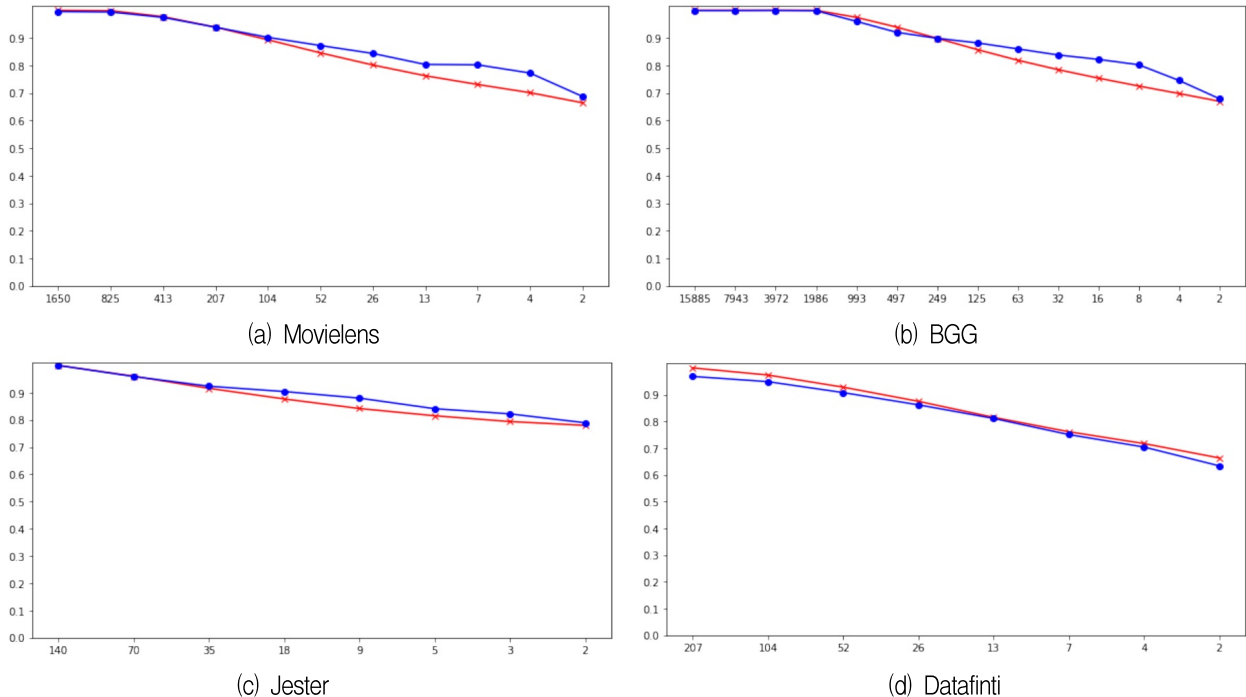


그림 1. 실제 상대 순위 정확도와 특이값을 이용한 추정값의 비교

Fig. 1. Comparisons between actual relative ranking precision and estimation based on singular values

모든 가능한 조합( $(a_i^T, a_j^T, a_k^T) \in S$ )에 대해서 정확도를 측정하는 것은 행렬  $A \in R^{m \times n}$ 에 대해  $O(m^3)$  시간이 걸리기 때문에,  $3m$ 개만큼의 표본을 랜덤 추출하여 실험을 수행하였다.

실험 결과, 많은 계수에서 추정치가 실제값과 상당히 유사한 값을 가지는 것을 볼 수 있다. 각 데이터셋의 그림 1에 나타난 점들의 퍼센트 오차 절대값 평균(MAPE, Mean Absolute Percent Error)[14]은 표 1과 같다. 대체로 2%에서 4% 사이의 MAPE 값을 보여주고 있어서, 차원 축소를 위해 특이값 분해를 적용하는 경우에 모든 계수에 대해 상대 순위 정확도를 실험을 통해 계산해 보지 않아도, 특이값을 이용한 방법으로 비교적 정확하게 추정이 가능함을 알 수 있다.

표 1. 추정의 MAPE

Table 1. MAPE of estimation

Dataset	MAPE(%)
Movielens	3.58
BGG	2.57
Jester	1.97
Datafinti	1.98

표 2. 데이터셋의 크기

Table 2. Size of datasets

Dataset	User	Item	Rating	Rating range	Sparsity(%)
Movielens	943	1650	80000	[1, 5]	94.86
BGG	2000	16034	495845	[0, 10]	98.45
Jester	2000	140	246573	[-10, 10]	11.94
Datafinti	408	207	2605	[1, 5]	96.92

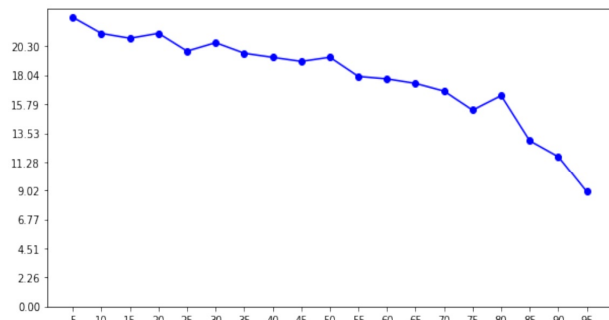
실험에 사용된 4개 데이터셋의 사용자, 품목, 평점 수 및 계획 행렬의 희소성(Sparsity)은 표 2와 같다. 일부 데이터는 표 2에 나타난 것보다 데이터 크기가 컸는데, 매우 큰 행렬에 대한 특이값 분해는 실험이 쉽지 않기 때문에, 사용자를 표본 추출하여 행렬의 크기를 표 2와 같이 줄인 다음 실험에 사용하였다. 행렬의 희소성은 밀도(Density)와 반대되는 개념으로, 행렬의 원소 중 0인 원소의 비율을 나타낸다. 예를 들어 총 원소의 수가 20개인  $5 \times 4$  행렬이 있을 때, 값이 0인 원소가 18개 있다면, 이 행렬의 희소성은 90%가 되고 밀도는 10%가 된다.

각 계획 행렬을 특이값 분해했을 때 특이값이 가지는 통계값은 표 3과 같다. 표 2를 보면 Jester 데이터셋을 제외한 다른 데이터셋은 희소성이 90% 이상으로 매우 큰 것을 볼 수 있다. 또한 표 3에서 왜도(Skewness)를 보면, 특이값의 분포가 고르지 않고 한 쪽으로 쏠려 있어, 수치적으로 낮은 계수의 (Numerically low rank) 행렬이라는 것을 볼 수 있다.

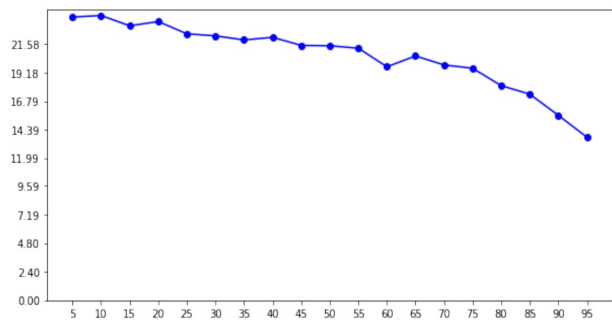
표 3. 데이터셋 별 특이값 통계

Table 3. Statistics of singular values of datasets

Dataset \ Statistic	Movielens	BGG	Jester	Datafinti
Max	525.77	2200.50	1604.96	95.31
Min	0.45	16.25	11.93	6.16
Mean	21.64	79.57	157.27	7.97
Variance	693.96	5442.66	20598.05	83.88
Median	14.88	64.12	139.88	5.65
Skewness	8.60	13.59	7.72	4.86
Kurtosis	145.17	351.59	72.62	39.01



(a) Random matrix of movielens shape



(b) Random matrix of BGG shape

그림 2. 희소성에 따른 상대 순위 정확도의 MAPE  
Fig. 2. MAPE of relative ranking precision by sparsity

식 (5)에 의한 상대 순위 정확도의 추정이 모든 행렬이 아니라 특정한 성질을 가지는 행렬에서만 성립하는 것일 수 있기 때문에, 실험에 사용된 계획 행렬과 같은 형태(행과 열 수) 및 같은 평점값 범위를 가지되, 평점값이 범위 내에서 랜덤으로 정해지고, 희소성이 서로 다른 랜덤 행렬을 생성하여 식 (5)에 의한 추정의 MAPE 값을 그림 2와 같이 계산해 보았다. 그림 2에서 x축은 희소성을, y축은 실제 상대 순위 정확도에 대한 추정의 MAPE 값을 나타낸다.

그림 2를 보면, 대체로 랜덤 행렬의 희소성이 증가할수록 식 (5)에 의한 추정의 MAPE가 감소하는 것을 볼 수 있다. 이것은 식 (5)가 희소한 계획 행렬에 대해 보다 정확한 추정을 할 수 있다는 것을 의미한다. 하지만 희소성이 95%일 때의 랜덤 행렬에 대한 MAPE 값과 실험에 사용된 공개 데이터셋들의 계획 행렬에 대한 MAPE 값에는 차이가 있는데, 이는 두 행렬의 계수 차이 때문이라고 생각된다.

표 3과 표 4의 왜도를 비교해 보면 랜덤 행렬의 왜도가 더 작은 값을 가지기 때문에 실험에 사용된 공개 데이터셋의 계획 행렬이 비교적 더 수치적으로 낮은 계수를 가진다고 볼 수 있으며, 희소성이 같은 상황에서는 낮은 계수의 데이터에서 보다 작은 MAPE 값을 보여준다고 할 수 있다. 결론적으로 식 (5)에 의한 추정은 희소한 행렬일수록, 그리고 낮은 계수의 행렬에서 높은 정확도를 나타낸다고 보인다. 실제 개인화 추천 분야의 데이터셋들이 대부분 이와 같은 경향을 나타내기 때문에, 식 (5)가 실용적으로 추정에 사용될 것을 기대할 수 있다.

표 4. 랜덤 행렬의 특이값 통계

Table 4. Statistics of singular values of random matrices

Statistic \ Random matrix	Movielens shape	BGG shape
Max	196.06	448.36
Min	7.26	59.50
Mean	27.66	91.21
Variance	165.99	347.19
Median	26.77	90.42
Skewness	2.49	3.58
Kurtosis	29.29	65.83

## V. 결 론

본 논문에서는 계획 행렬에 특이값 분해를 적용하고, 큰 특이값에 해당하는 특이벡터만을 남겨 데이터에 대한 차원 축소를 시도할 경우, 데이터 사이의 코사인 유사도의 순위가 얼마나 보존되는지를 측정하기 위한 지표인 상대 순위 정확도를 정의하였다. 또한 다양한 데이터에 대한 실험을 통해, 특이값과 상대 순위 정확도 사이에 관계가 있다는 것과, 특이값을 이용해 상대 순위 정확도를 식 (5)와 같이 추정할 수 있음을 보였다. 그리고 이와 같은 추정의 대상이 된 계획 행렬의 희소성 및 특이값의 통계를 살펴봄으로써, 희소성이 크고 수치적으로 계수가 작은 행렬에서 위와 같은 추정의 정확도가 더 높아진다는 것을 분석하였다. 향후에는 실험적으로 뿐 아니라 보다 분석적인 방법을 통해 식 (5)를 유도할 수 있는 가능성에 대해서 검토해 보고자 한다.

## References

- [1] Netflix prize, <https://web.archive.org/web/20090919150646/http://www.netflixprize.com/index>. [accessed: Mar. 10, 2020]
- [2] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization", *Advances in Neural Information Processing Systems 13 (NIPS 2000)*, Colorado, USA. pp. 556-562, Dec. 2000.
- [3] M. Grbovic, V. Radosavljevic, N. Djuric, N. Bhamidipati, J. Savla, V. Bhagwan, and D. Sharp, "E-commerce in your inbox: Product recommendations at scale", in *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, Sydney NSW Australia, pp. 1809-1818, Aug. 2015.
- [4] P. Covington, J. Adams, and E. Sargin, "Deep neural networks for youtube recommendations", in *Proceedings of the 10th ACM conference on recommender systems*, Boston, MA, USA. pp. 191-198, Sep. 2016.
- [5] D. C. Liu, S. Rogers, R. Shiao, D. Kislyuk, K. C. Ma, Z. Zhong, J. Liu, and Y. Jing, "Related pins

at pinterest: The evolution of a real-world recommender system", in Proceedings of the 26th International Conference on World Wide Web Companion, Perth, Austria, pp. 583-592, Apr. 2017.

- [6] Cosine similarity, [https://en.wikipedia.org/wiki/Cosine\\_similarity](https://en.wikipedia.org/wiki/Cosine_similarity). [accessed: Mar. 10, 2020]
- [7] C. Eckart and G. Young, "The approximation of one matrix by another of lower rank", *Psychometrika*, Vol. 1, No. 3, pp. 211-218, Sep. 1936.
- [8] W. B. Johnson and J. Lindenstrauss, "Extensions of lipschitz mappings into a hilbert space", *Contemporary mathematics*, Vol. 26, pp. 189-206, Jan. 1984.
- [9] E. Bingham and H. Mannila, "Random projection in dimensionality reduction: Applications to image and text data", in Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining, San Francisco California, USA, pp. 245-250, Aug. 2001.
- [10] Movielens 100k dataset, <https://www.kaggle.com/prajitdatta/movielens-100k-dataset>. [accessed: Mar. 10, 2020]
- [11] Boardgamegeek reviews, <https://www.kaggle.com/jvanelteren/boardgamegeek-reviews>. [accessed: Mar. 10, 2020]
- [12] Jester 1.7m jokes ratings dataset, <https://www.kaggle.com/vikashrajuhaniwal/jester-17m-jokes-ratings-dataset>. [accessed: Mar. 10, 2020]
- [13] Grammar and online product reviews, <https://www.kaggle.com/datafiniti/grammar-and-online-product-reviews>. [accessed: Mar. 10, 2020]
- [14] Mean absolute percentage error, [https://en.wikipedia.org/wiki/Mean\\_absolute\\_percentage\\_error](https://en.wikipedia.org/wiki/Mean_absolute_percentage_error). [accessed: Mar. 10, 2020]

## 저자소개

유 찬 우 (Chanwoo Yoo)



2003년 8월 : 서울대학교  
컴퓨터공학부(공학사), 경영학과  
(경영학사)

2012년 2월 : 서울대학교  
전기컴퓨터공학부(공학박사)

2012년 7월 ~ 2014년 9월 : LG  
전자

2016년 3월 ~ 2018년 4월 : 네무스텍

2018년 5월 ~ 현재 : 라인플러스 Data Science Dev Lead

관심분야 : 머신러닝, 온라인 러닝, 추천 알고리즘