Rating Prediction using Matrix Factorization and SVD

Analytics Vidhya

Matrix Factorization

- SVD is the factorization of a matrix M into 3 constituent matrices

$$M = U \times \Sigma \times V^T$$

where U and V are called *left* and *right singular vectors* and the values of the diagonal of Σ are called the *singular values*

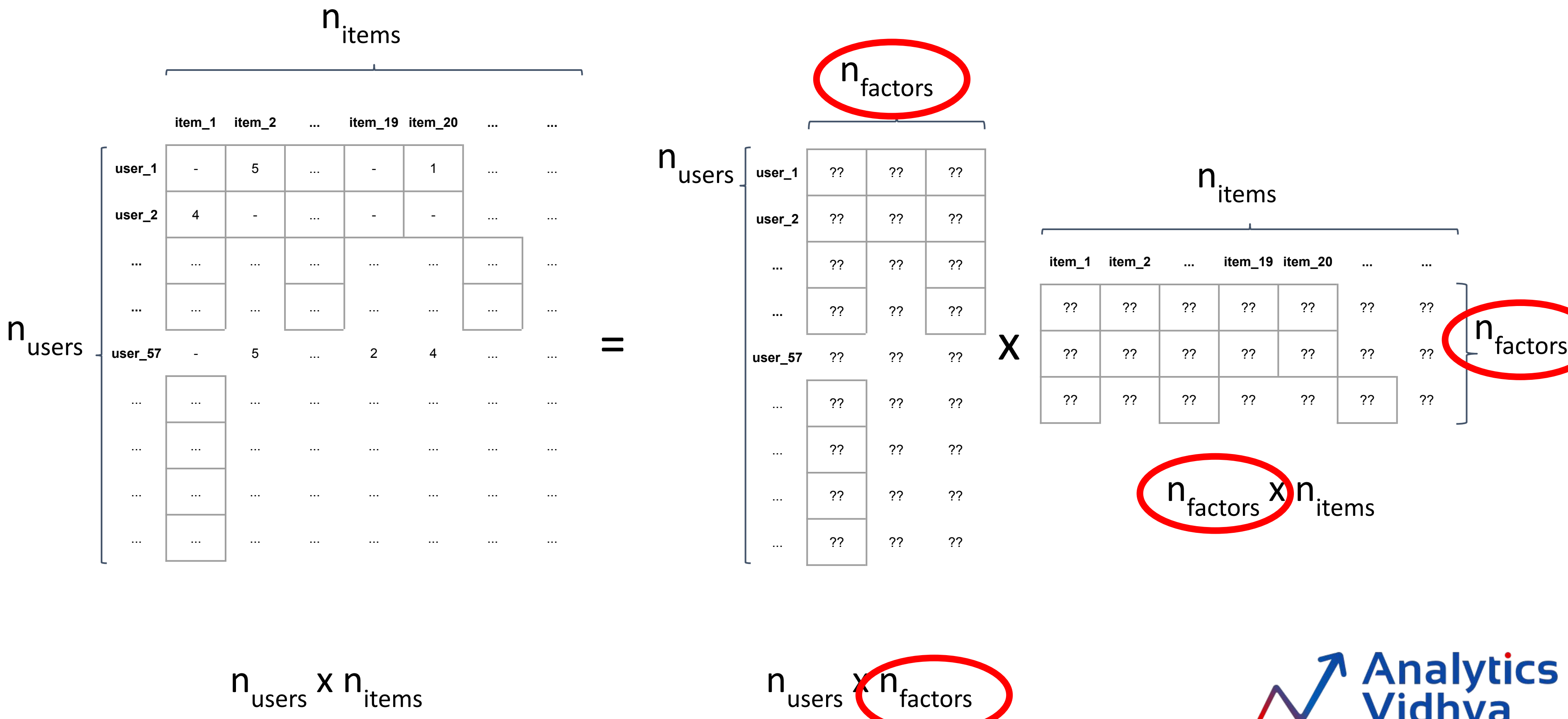
- We can approximate the full matrix by observing only the most important features – those with the largest singular values
- This can be done by decomposing rating matrix into a user and item matrix using a dimensionality reduction technique

Analytics Vidhya

Singular Value Decomposition

- $$R = P\Sigma Q^T$$
- R is *m x n* ratings matrix
- P is *m x k* user-feature affinity matrix
- Q is *n x k* item-feature relevance matrix
- Σ is *k x k* diagonal feature weight matrix
- SVD defines a shared vector space for item and users (feature space)

Matrix Factorization



Analytics Vidhya

Interpretation of the User and Item matrices

attribute k an item can possess and a user can be susceptible to

how much user u is susceptible to this attribute k / how much it is important to them.

	user hidden factor k-1	user hidden factor k	user hidden factor k+1
user_1	??	??	??
user_2	??	??	??
...	??	??	??
...	??	??	??
user_u	??	??	??
...	??	??	??
...	??	??	??
...	??	??	??
...	??	??	??

User

	item hidden factor k-1	item hidden factor k	item hidden factor k+1
item_1	??	??	??
item_2	??	??	??
...	??	??	??
item_i	??	??	??
...	??	??	??
...	??	??	??
...	??	??	??

Item

how much item i possesses this attribute k

Example for SVD-based recommendation

- **SVD:** $M_k = U_k \times \Sigma_k \times V_k^T$

U_k	Dim1	Dim2
Alice	0.47	-0.30
Bob	-0.44	0.23
Mary	0.70	-0.06
Sue	0.31	0.93

V_k^T	Terminator	Die Hard	Twins	Eat Pray Love	Pretty Woman
Dim1	-0.44	-0.57	0.06	0.38	0.57
Dim2	0.58	-0.66	0.26	0.18	-0.36

Σ_k	Dim1	Dim2
Dim1	5.63	0
Dim2	0	3.23

- **Prediction:** $\hat{r}_{ui} = \bar{r}_u + U_k(Alice) \times \Sigma_k \times V_k^T(EPL)$

$$= 3 + 0.84 = 3.84$$

Analytics Vidhya

What does SVD Achieve?

- SVD captures hidden relationships between users and items

Solving Problem of
Synonymy & Sparsity

- SVD provides lower dimension representation of the original user-item space

Solving Problem of
Scalability