Zurich University of Applied Sciences

School of Engineering

Collaborative Filtering

Manuel Dömer
ZHAW School of Engineering

Recommending Movies

- The Netflix Prize

Supervised Learning Setting

Feature representation (encoding)

$$\Phi : \text{item}^{(m)} \rightarrow \mathbf{x}^{(m)} = \begin{pmatrix} x_1^{(m)} \\ \vdots \\ x_N^{(m)} \end{pmatrix}$$

Training Data

Design matrix $\mathbf{X}$

feature n → item m ↓	**1**	**2**	**…**	N
1	x_1^1	x_2^1		x_N^1
2	x_1^2	x_2^2		x_N^2
$\vdots$				
M	x_1^M	x_2^M		x_N^M

Ratings $\mathbf{y}$

y^1
y^2
$\vdots$
y^M

Learning Task

Learn a predictor, f, that maps an N-dimensional vector representation of an item (row in $\mathbf{X}$) to an output value (element in $\mathbf{y}$)

$$\mathrm{f}\left(\mathbf{x}^{(\mathrm{m})}\right) \rightarrow \mathrm{y}^{(\mathrm{m})}$$

- $\mathrm{y}^{(\mathrm{m})} \in \{1, 2, 3, 4, 5\}$ → classification
- $\mathrm{y}^{(\mathrm{m})} \in \mathbb{R}$ → regression

Regression Problem

- Hypothesis, e.g. linear: $f(\mathbf{x}^{(m)}) = \theta^T \mathbf{x}^{(m)}$
- Loss function: $\square = \sum_{m=1}^{M} \left(y^m - f(\mathbf{x}^{(m)})\right)^2$
- + regularisation
- Cost function:

$$J(\theta) = \frac{1}{2M} \sum_{m=1}^{M} \left(y^m - \theta^T \mathbf{x}^{(m)}\right)^2 + \frac{\lambda}{2} ||\theta||^2$$

- Minimise: Solve analytically or by gradient descent

Problems

- Difficult to design expressive features
- For personal recommendations data from other users is not leveraged

Rating Matrix

$\mathbf{Y}$: M users $\times$ N items

item n → user m ↓	**1**	**2**	**...**	N
1	$y_1^{(1)}$	$y_2^{(1)}$		$y_N^{(1)}$
2	$y_1^{(2)}$	$y_2^{(2)}$		$y_N^{(2)}$
$\vdots$				
M	$y_1^{(M)}$	$y_2^{(M)}$		$y_N^{(M)}$

Matrix Filling Task

Rating matrix is very sparse

item n → user m ↓	1	2	…	N
1	5			3
2		1		
⋮				2
M	1			

Collaborative Filtering - The Principle

Using the Nearest Neighbour approach

- User-based vs. item-based

The Nearest Neighbour Approach

Calculate the unknown rating as the average rating of the other users weighted by similarity

E.g. by cosine similarity

$$s_{m,m'} = \frac{\mathbf{x}^{(m)} \cdot \mathbf{x}^{(m')}}{|\mathbf{x}^{(m)}||\mathbf{x}^{(m')}|}$$

- by row → user-based
- By columns → item-based

User ↓ Item →	1	2	3	4	5	6
1	1	3	1		5	4
2	5	4	4		1	
3	2		5	4	5	
4		3				5
5		2		5	4	
6			4	4		5

Similarity of Users 1 and 2

User ↓ Item →	1	2	3	4	5	6
1	1	3	1		5	4
2	5	4	4		1	
…						

Only consider items rated by both users

$$
\begin{aligned}
s_{1,2} &= \frac{\mathbf{x}^{(1)} \cdot \mathbf{x}^{(2)}}{|\mathbf{x}^{(1)}||\mathbf{x}^{(2)}|} \\
&= \frac{1 \cdot 5 + 3 \cdot 4 + 1 \cdot 4 + 5 \cdot 1}{\sqrt{1^2 + 3^2 + 1^2 + 5^2} \cdot \sqrt{5^2 + 4^2 + 4^2 + 1^2}} = 0.57
\end{aligned}
$$

Similarity Matrix

User↓ User→	1	2	3	4	5	6
1	1.00	0.57	0.84	0.99	1.00	0.91
2	0.57	1.00	0.73	1.00	0.65	1.00
3	0.84	0.73	1.00	0.00	0.98	0.99
4	0.99	1.00	0.00	1.00	1.00	1.00
5	1.00	0.65	0.98	1.00	1.00	1.00
6	0.91	1.00	0.99	1.00	1.00	1.00

Calculation of an Unknown Rating

User↓ Item→	1	2	3	4	5	6
1	1	3	1		5	4
2	5	4	4	?	1	
3	2		5	4	5	
4		3				5
5		2		5	4	
6			4	4		5

User↓ User→	1	2	3	4	5	6
1	1.00	0.57	0.84	0.99	1.00	0.91
2	0.57	1.00	0.73	1.00	0.65	1.00
3	0.84	0.73	1.00	0.00	0.98	0.99
4	0.99	1.00	0.00	1.00	1.00	1.00
5	1.00	0.65	0.98	1.00	1.00	1.00
6	0.91	1.00	0.99	1.00	1.00	1.00

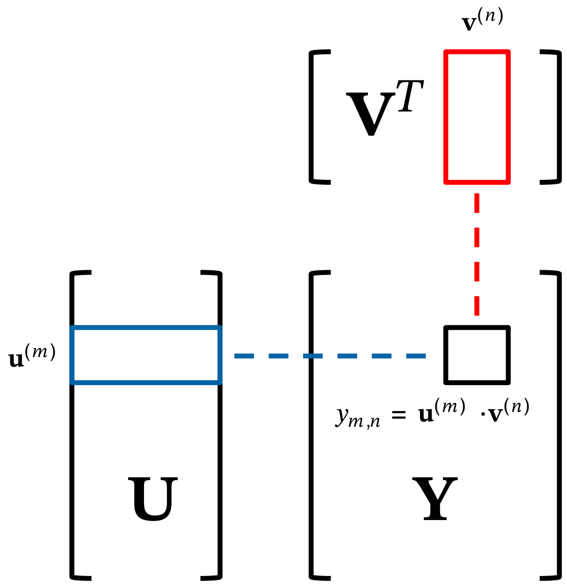
Predicted rating for user $m = 2$, item $n = 4$:

$$
\begin{aligned}
\hat{y}_{2,4} &= \frac{1}{s_{2,3} + s_{2,5} + s_{2,6}} \left(s_{2,3} \cdot y_{3,4} + s_{2,5} \cdot y_{5,4} + s_{2,6} \cdot y_{6,4} \right) \\
&= \frac{1}{0.73 + 0.65 + 1.0} (0.73 \cdot 4 + 0.65 \cdot 5 + 1.0 \cdot 4) \\
&= 4.27
\end{aligned}
$$

Matrix Factorisation

For a matrix $\mathbf{Y} : \mathrm{M} \times \mathrm{N}$ of rank K there exist $\mathbf{U} : \mathrm{N} \times \mathrm{K}$ and $\mathbf{V} : \mathrm{M} \times \mathrm{K}$, such that

$$\mathbf{Y} = \mathbf{U}\mathbf{V}^{\mathrm{T}}$$

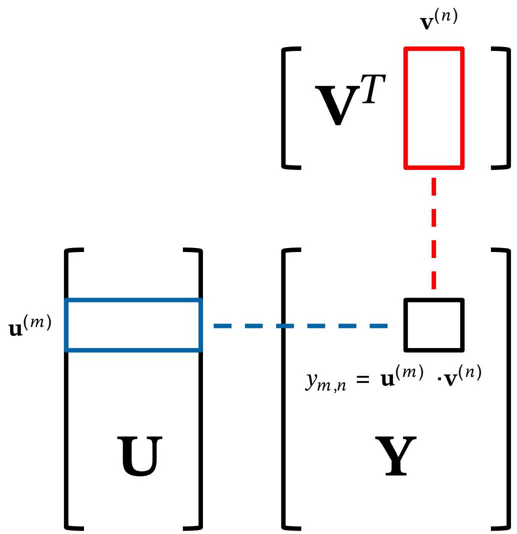


Matrix Factorisation

$$\mathbf{Y} = \mathbf{U}\mathbf{V}^{\mathrm{T}}$$

- Problem: $\mathbf{Y}$ is sparse
- Approach:
  - Calculate $\mathbf{U}$ and $\mathbf{V}$ based on available entries in $\mathbf{Y}$
  - Use $\mathbf{U}$ and $\mathbf{V}$ to predict unknown ratings $\hat{\mathbf{Y}}$

Factorisation Machines



$$\text{Cost: } J(U,V) = \frac{1}{2} \sum_{\substack{(m,n) \text{ where} \\ y_{m,n} \neq 0}} \left(y_{m,n} - \left(\mathbf{u}^{(m)} \cdot \mathbf{v}^{(n)} + b_u^{(m)} + b_v^{(n)} \right) \right)^2$$

$$+ \frac{\lambda}{2} \sum ||\mathbf{u}^{(m)}||^2 + \frac{\lambda}{2} \sum ||\mathbf{v}^{(n)}||^2$$

Minimise by alternating least squares or stochastic gradient descent

Summary

Collaborative filtering:

- No item features needed
- User ratings required
- Current interests infered from historic user behavior
- Sparsity
- Cold start problems
- Users’s range of interests can be expanded

Content-based:

- Item features required
- No ratings required
- No cold-start or sparsity problem
- new and less famous objects are also recommended
- Serendipity effect is not really supported