FACTORIZATION MACHINE: MODEL, OPTIMIZATION AND APPLICATIONS

Yang LIU
Email: yliu@cse.cuhk.edu.hk
Supervisors: Prof. Andrew Yao
Prof. Shengyu Zhang

OUTLINE

- Factorization machine (FM)
  - A generic predictor
  - Auto feature interaction
- Learning algorithm
  - Stochastic gradient descent (SGD)
  - …
- Applications
  - Recommendation systems
  - Regression and classification
  - …

DouBan movie

Prediction Task

	Feature vector **x**																						Target y	
	User				Movie					Other Movies rated					Time	Last Movie rated							
	A	B	C	...	TI	NH	SW	ST	...	TI	NH	SW	ST	...	Time	TI	NH	SW	ST	...			
$\mathbf{x}^{(1)}$	1	0	0	...	1	0	0	0	...	0.3	0.3	0.3	0	...	13	0	0	0	0	...			5	$y^{(1)}$
$\mathbf{x}^{(2)}$	1	0	0	...	0	1	0	0	...	0.3	0.3	0.3	0	...	14	1	0	0	0	...			3	$y^{(2)}$
$\mathbf{x}^{(3)}$	1	0	0	...	0	0	1	0	...	0.3	0.3	0.3	0	...	16	0	1	0	0	...			1	$y^{(2)}$
$\mathbf{x}^{(4)}$	0	1	0	...	0	0	1	0	...	0	0	0.5	0.5	...	5	0	0	0	0	...			4	$y^{(3)}$
$\mathbf{x}^{(5)}$	0	1	0	...	0	0	0	1	...	0	0	0.5	0.5	...	8	0	0	1	0	...			?	
$\mathbf{x}^{(6)}$	0	0	1	...	1	0	0	0	...	0.5	0	0.5	0	...	9	0	0	0	0	...			?	
$\mathbf{x}^{(7)}$	0	0	1	...	0	0	1	0	...	0.5	0	0.5	0	...	12	1	0	0	0	...				

- e.g. **Alice** rates **Titanic** $\boxed{5}$ at time **13**

Prediction Task

- Format: $y(x)\colon \mathbb{R}^n \to T$
  - $T = \mathbb{R}$ for regression,
  - $T = \{+1, -1\}$ for classification

- Training set: $\mathrm{Tr} = \{(x^1, y^1), (x^2, y^2) \dots\}$

- Testing set: $\mathrm{Te} = \{x_1, x_2, \dots\}$,

- Objective: to predict $\{y(x_1), y(x_2), \dots\}$

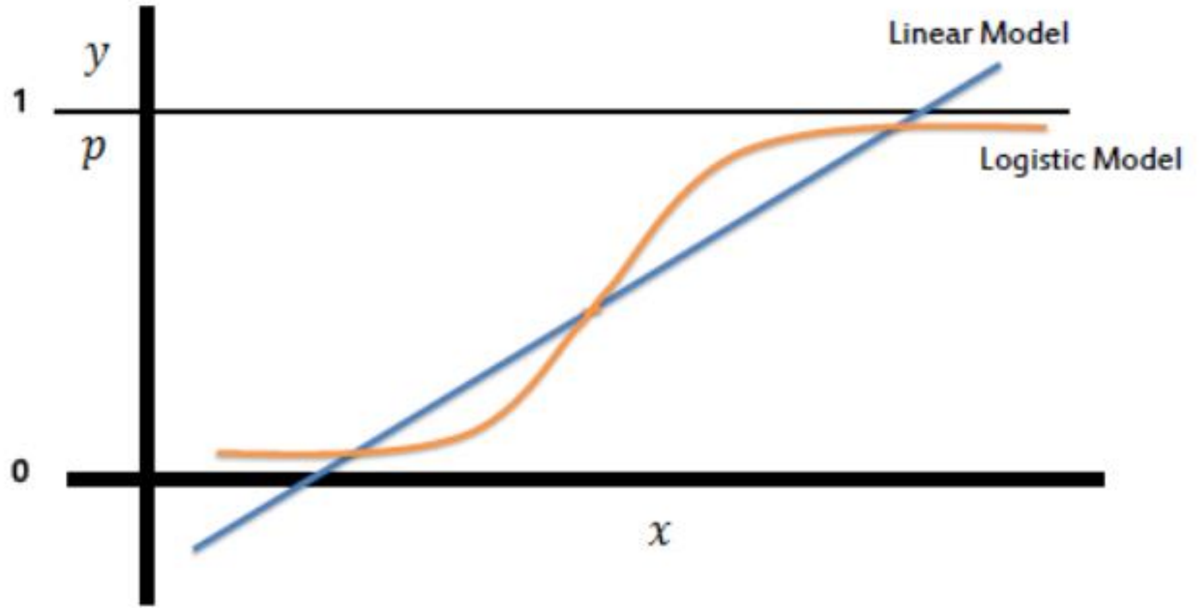
Linear Model – Feature Engineering

- Linear SVM

$$\hat{y}(x) = w_0 + w^T x$$

- Logistic Regression

$$\hat{y}(x) = \frac{1}{1 + w_0 \exp(-w^T x\ )}$$



Factorization model

Linear: $\hat{y}(x) := w_0 + \sum_{i=1}^{n} w_i x_i$

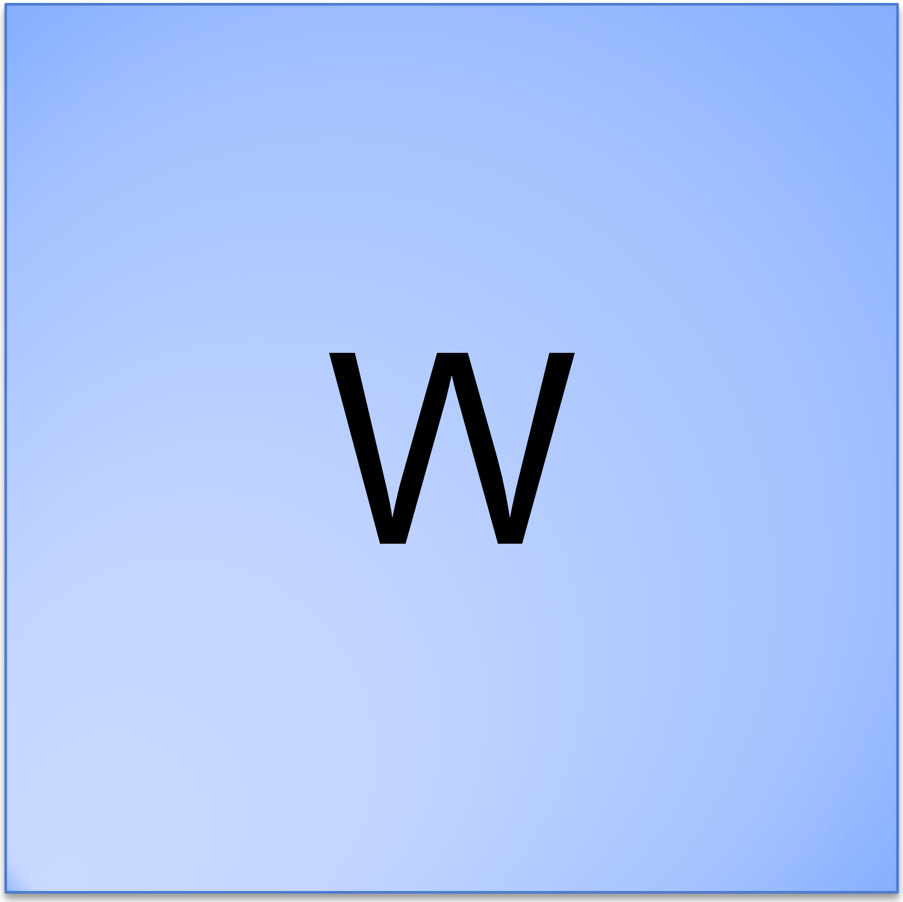
FM: $\hat{y}(x) := w_0 + \sum_{i=1}^{n} w_i x_i + \sum_{i=1}^{n} \sum_{j=i+1}^{n} \langle v_i, v_j \rangle x_i x_j$

Interaction between variables

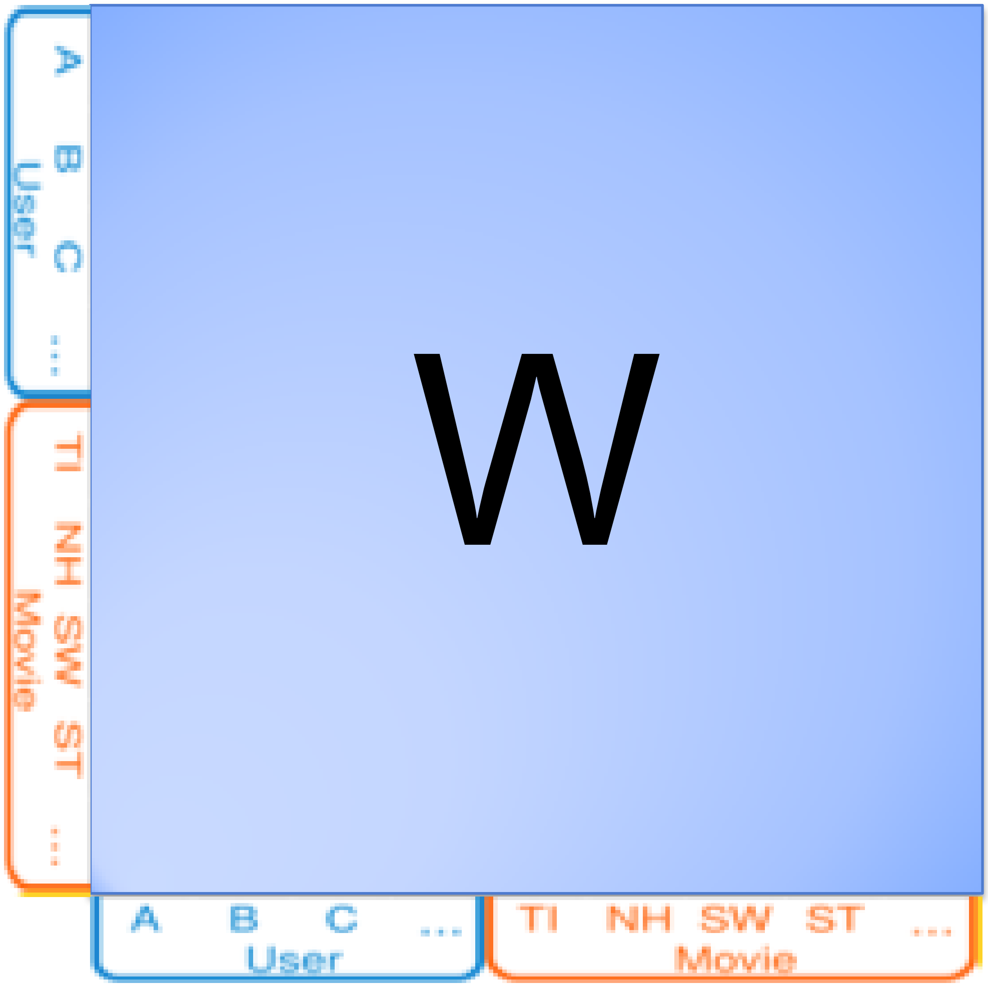
- Model parameters $\Theta = \{w_0, w_1, \ldots w_n, v_1, \ldots, v_n\}$
  - $v_i \in \mathbb{R}^k, i = 1, \ldots, n$, where
- k is the inner dimension

Interaction matrix

$$w_{i,j} = \langle v_i, v_j \rangle$$

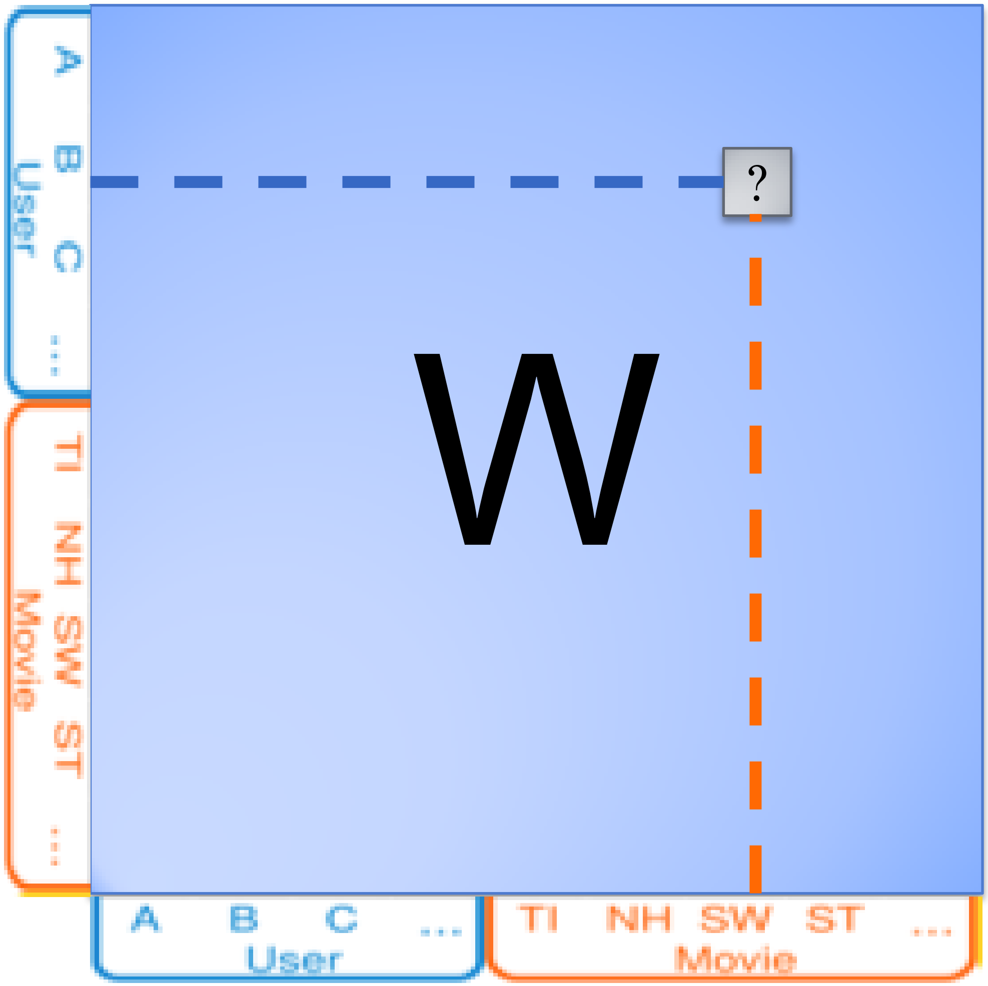


INTERACTION MATRIX $w_{i,j} = \langle v_i, v_j \rangle$



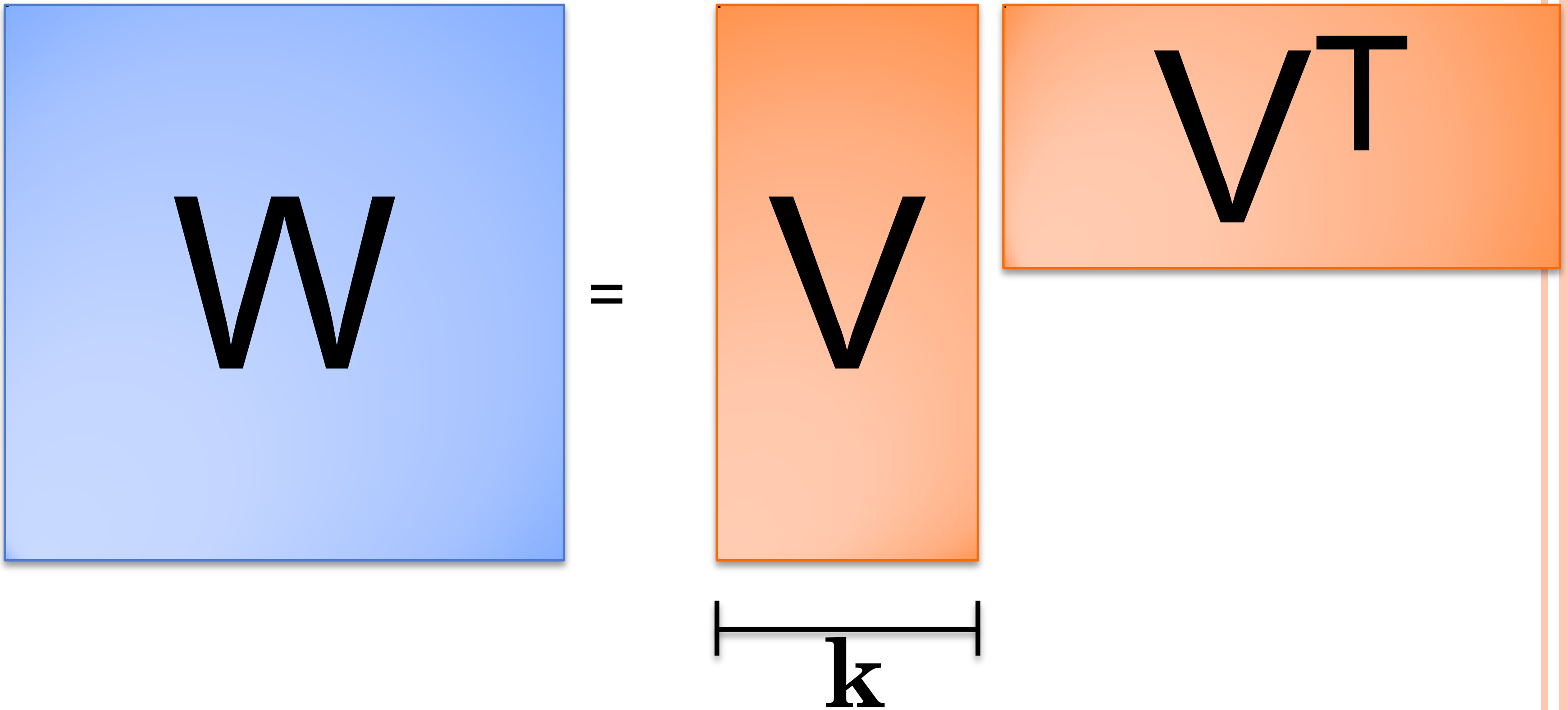
INTERACTION MATRIX

$$w_{i,j} = \langle v_i, v_j \rangle$$



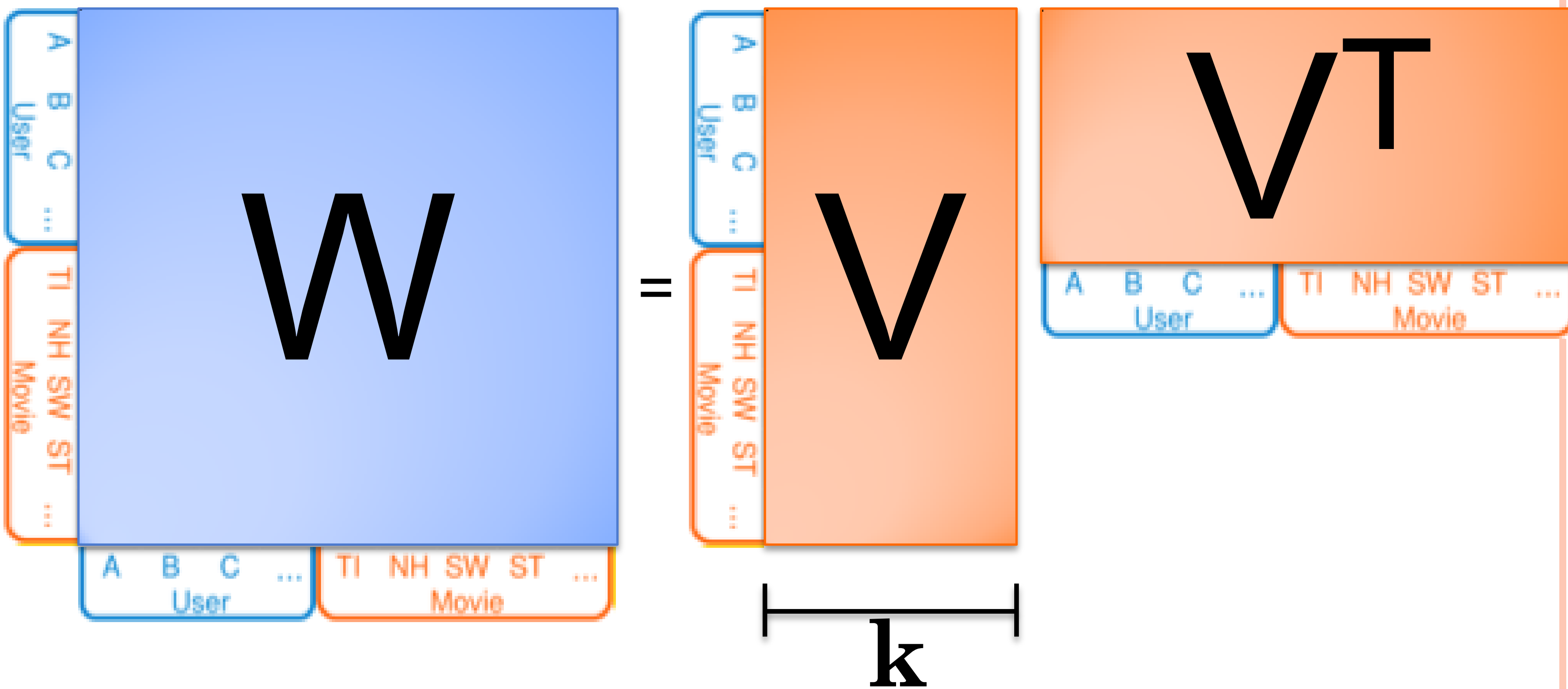
Interaction matrix

$$w_{i,j} = \langle v_i, v_j \rangle$$



Interaction matrix

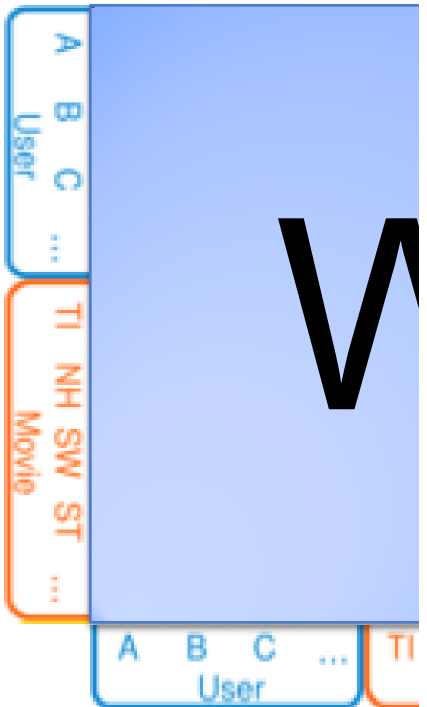
$$w_{i,j} = \langle v_i, v_j \rangle$$



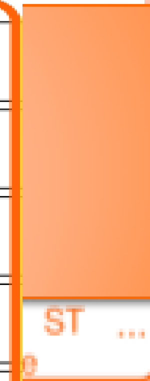
$$\hat{y}(x) := w_0 + \sum_{i=1}^{n} w_i x_i + \sum_{i=1}^{n} \sum_{j=i+1}^{n} \langle v_i, v_j \rangle x_i x_j$$

Interaction matrix

$$w_{i,j} = \langle v_i, v_j \rangle$$



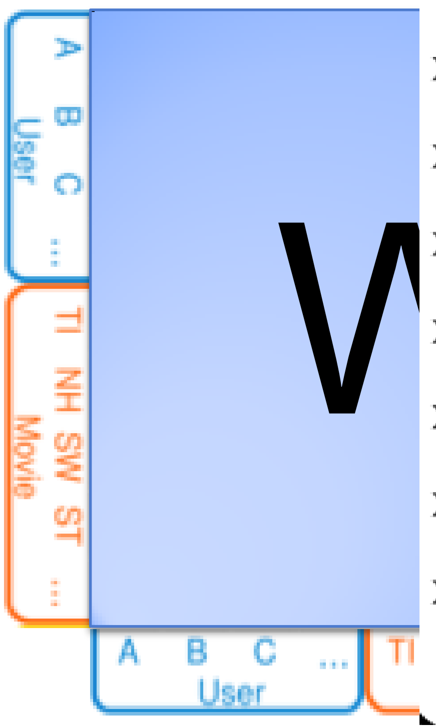
	A	B	C	...	TI	NH	SW	ST	...
$\mathbf{x}^{(1)}$	1	0	0	...	1	0	0	0	...
$\mathbf{x}^{(2)}$	1	0	0	...	0	1	0	0	...
$\mathbf{x}^{(3)}$	1	0	0	...	0	0	1	0	...
$\mathbf{x}^{(4)}$	0	1	0	...	0	0	1	0	...
$\mathbf{x}^{(5)}$	0	1	0	...	0	0	0	1	...
$\mathbf{x}^{(6)}$	0	0	1	...	1	0	0	0	...
$\mathbf{x}^{(7)}$	0	0	1	...	0	0	1	0	...
	User				Movie				



$$\hat{y}(x) := w_0 + \sum_{i=1}^{n} w_i x_i + \sum_{i=1}^{n} \sum_{j=i+1}^{n} \langle v_i, v_j \rangle x_i x_j$$

Interaction matrix

$$w_{i,j} = \langle v_i, v_j \rangle$$



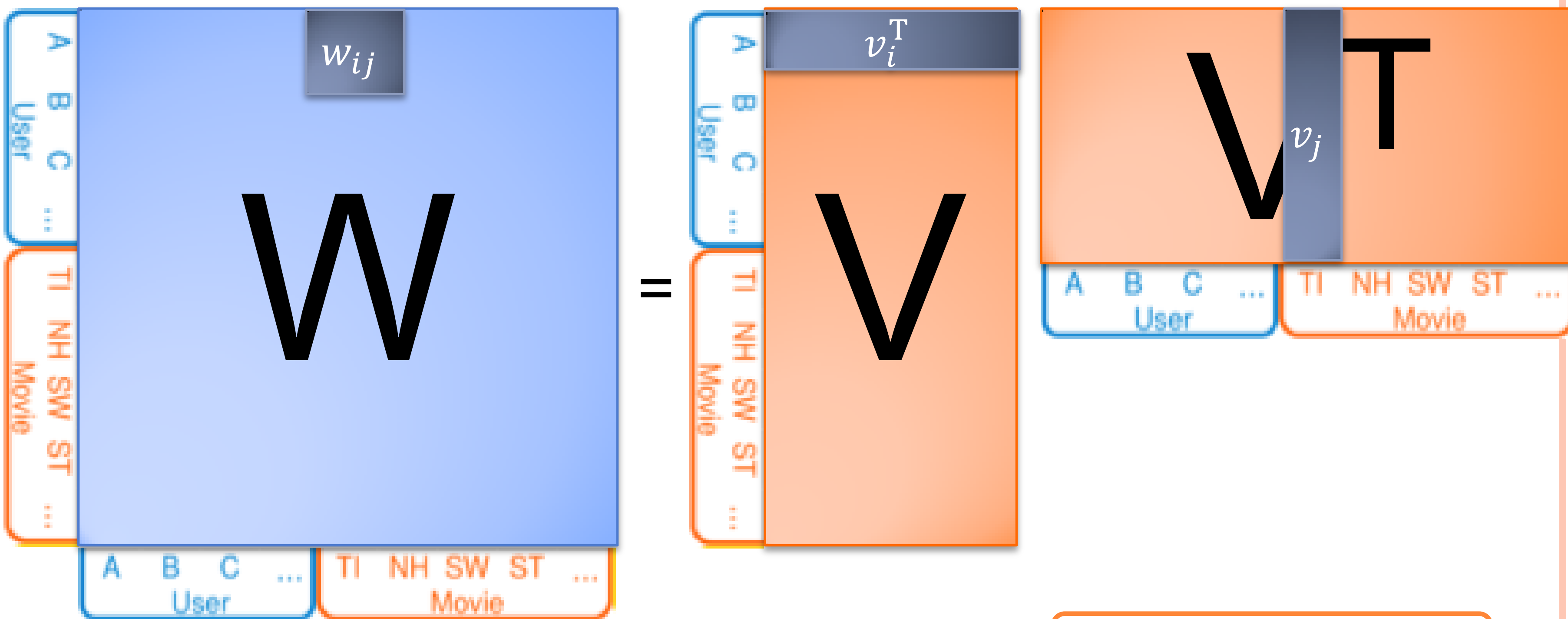
	A	B	C	...	TI	NH	SW	ST	...
$\mathbf{x}^{(1)}$	1	0	0	...	1	0	0	0	...
$\mathbf{x}^{(2)}$	1	0	0	...	0	1	0	0	...
$\mathbf{x}^{(3)}$	1	0	0	...	0	0	1	0	...
$\mathbf{x}^{(4)}$	0	1	0	...	0	0	1	0	...
$\mathbf{x}^{(5)}$	0	1	0	...	0	0	0	1	...
$\mathbf{x}^{(6)}$	0	0	1	...	1	0	0	0	...
$\mathbf{x}^{(7)}$	0	0	1	...	0	0	1	0	...
	User				Movie				

$$= v_A^T v_{TI}$$

$$\hat{y}(x) := w_0 + \sum_{i=1}^{n} w_i x_i + \sum_{i=1}^{n} \sum_{j=i+1}^{n} \boxed{\langle v_i, v_j \rangle x_i x_j}$$

INTERACTION MATRIX

$$w_{i,j} = \langle v_i, v_j \rangle$$

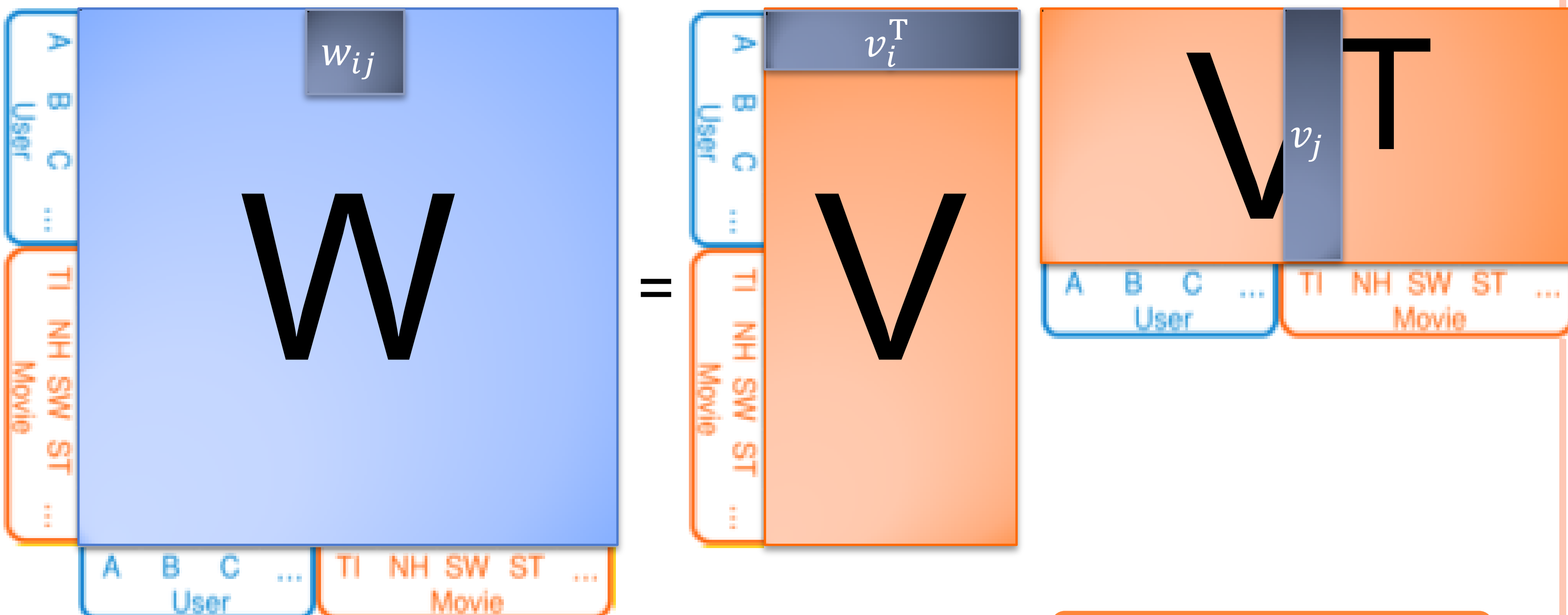


Factorization

$$\hat{y}(x) := w_0 + \sum_{i=1}^{n} w_i x_i + \boxed{\sum_{i=1}^{n} \sum_{j=i+1}^{n} \langle v_i, v_j \rangle x_i x_j}$$

Interaction matrix

$$w_{i,j} = \langle v_i, v_j \rangle$$



Machine

Factorization

$$\hat{y}(x) := w_0 + \sum_{i=1}^{n} w_i x_i + \sum_{i=1}^{n} \sum_{j=i+1}^{n} \langle v_i, v_j \rangle x_i x_j$$

FM: PROPERTIES

- $\hat{y}(x) \coloneqq w_0 + \sum_{i=1}^{n} w_i x_i + \sum_{i=1}^{n} \sum_{j=i+1}^{n} \langle v_i, v_j \rangle x_i x_j$
  $= w_0 + w^T x + \frac{1}{2} x^T \left(VV^T - diag(VV^T) \right) x$

- Expressiveness:
  - $\forall W \in \mathbb{R}^{n \times n} \succcurlyeq 0, \exists V \in \mathbb{R}^{n \times k}\ s.t. W = VV^T$

- Feature dependency:
  - $w_{i,j} = \langle v_i, v_j \rangle$ and $w_{j,k} = \langle v_j, v_k \rangle$ are dependent

- Linear computation complexity:
  - $O(kn)$

Optimization Target

- Min ERROR
- Min ERROR + Regularization

- $\text{OPT} = \underset{\Theta}{\operatorname{argmin}}\left(\sum_{(x,y)\in Tr} l(\hat{y}(x|\Theta), y) + \sum_{\theta\in\Theta} \lambda_\theta \theta^2\right)$

- Loss function
  - $l(y_1, y_2) = (y_1 - y_2)^2$
  - $l(y_1, y_2) = \ln(1 + \exp(-y_1 y_2))$

Stochastic Gradient Descent (SGD)

- For item (x, y), update θ by:
- $\theta \leftarrow \theta - \eta \left(\frac{\partial}{\partial \theta} l(\hat{y}(x), y) + 2\lambda_\theta \theta \right)$
  - θ_0: initial value of θ
  - η: learning rate
  - λ_θ: regularization
- Pros
  - Easy to implement
  - Fast convergence on big training data
- Cons
  - Parameter tuning
  - Sequential method

Applications

- EMI Music Hackathon 2012
  - Song recommendation
- Given:
  - Historical ratings
  - User demographics
- # features: 51K
- # items in training: 188K

	A	B	C	D	E
1	Artist	Track	User	Rating	Time
2	40	179	47994	9	17
3	9	23	8575	58	7
4	46	168	45475	13	16
5	11	153	39508	42	15
6	14	32	11565		19
7	31	79	27130		11
8	21	48	19623	?	21
9	2	174	47505		17
10	12	34	15290		8
11	28	73	24151	70	22
12	0	151	40578	32	15

RESULTS FOR EMI MUSIC

- FM: Root Mean Square Error (RMSE) 13.27626
  - Target value [0,100]
  - The best (SVD++) is 13.24598

- Details
  - Regression
  - Converges in 100 iterations
  - Time for each iteration: < 1 s
    - Win 7, Intel Core 2 Duo CPU 2.53GHz, 6G RAM

Other applications

- Ads CTR prediction (KDD Cup 2012)
  - Features
    - User_info, Ad_info, Query_info, Position, etc.
  - # features: 7.2M
  - # items in training: 160M
  - Classification
  - Performance:
    - AUC: 0.80178, the best (SVM) is 0.80893

OTHER APPLICATIONS

- HiCloud App Recommendation

  智汇云应用市场
  app.vmall.com

  - Features
    - App_info, Smartphone model, installed apps, etc.
  - # features: 9.5M
  - # items in training: 16M
  - Classification
  - Performance:
    - Top 5: 8%, Top 10: 18%, Top 20: 32%; AUC: 0.78

Summary

- FM: a general predictor
- Works under sparsity
- Linear computation complexity
- Estimates interactions automatically
- Works with any real valued feature vector

Thanks!