UFMG UNIVERSIDADE FEDERAL DE MINAS GERAIS

Recommender Systems

Restricted Boltzmann Machines

Rodrygo L. T. Santos
rodrygo@dcc.ufmg.br

The Deep Learning wave

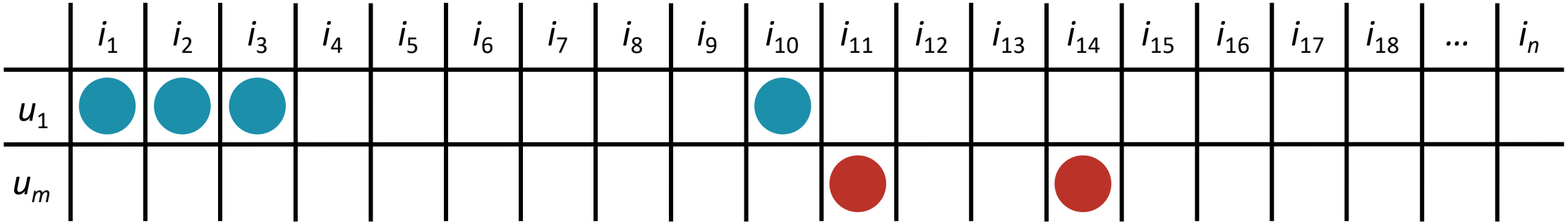
Deep Learning

- Machine learning algorithms based on learning multiple levels of representation / abstraction

Amazing improvements in error rate

- Object recognition and detection
- Speech recognition, natural language processing

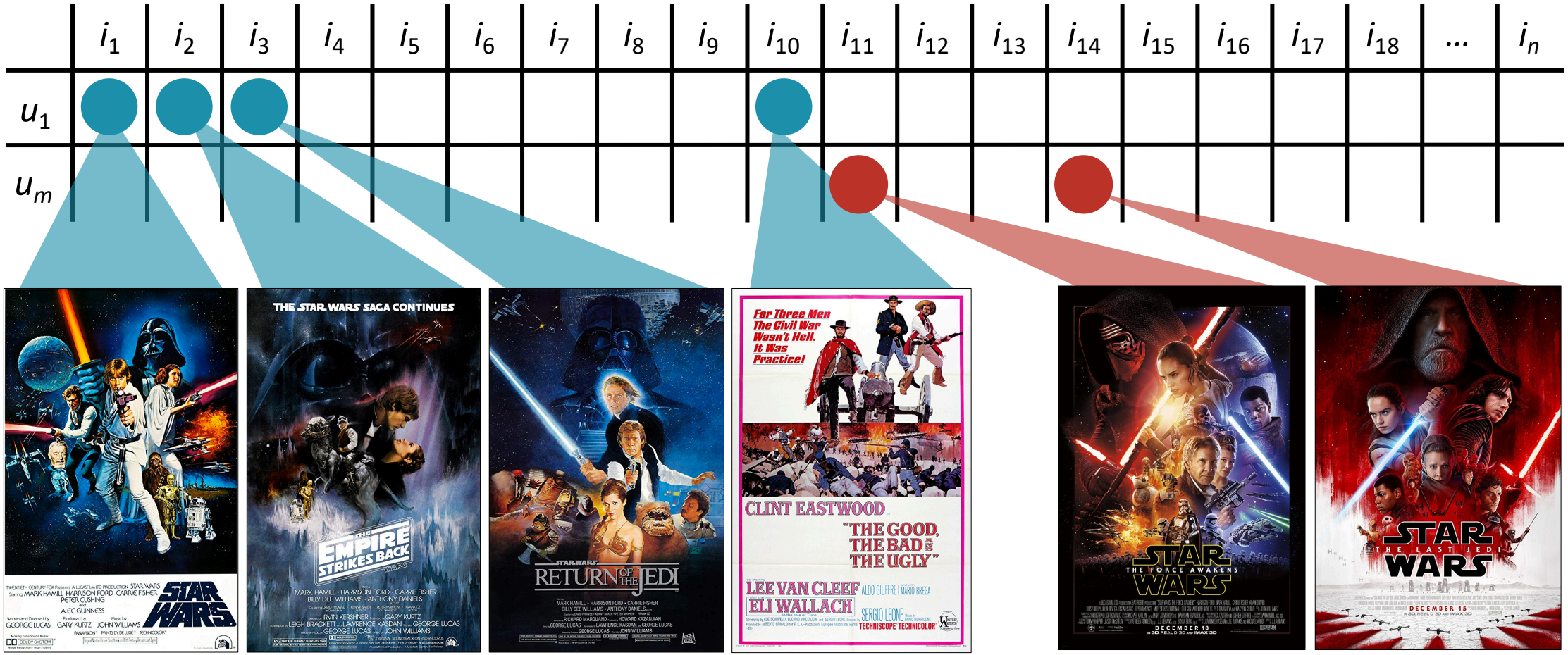
User-based collaborative filtering



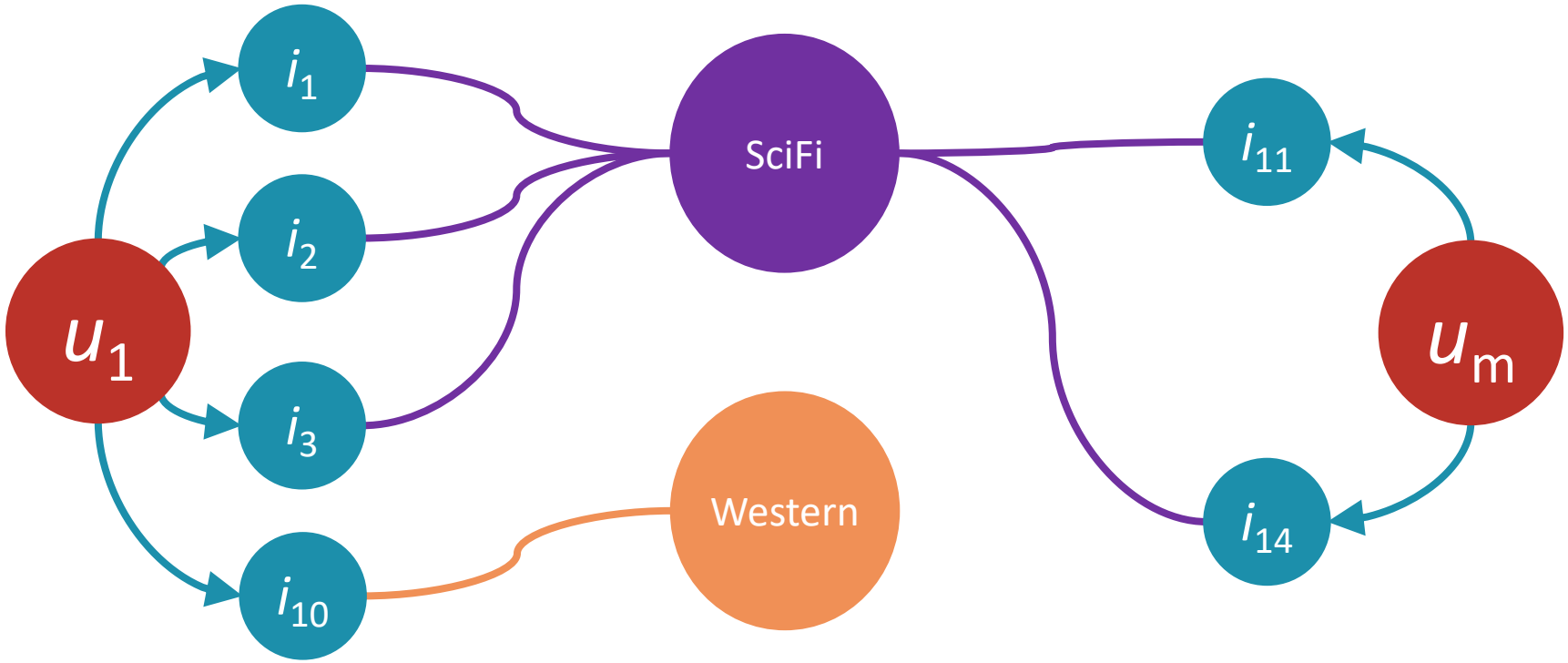
are they neighbors?

User-based collaborative filtering

Latent factor models



Latent factor models



Latent factor models

Distinct spaces in neighborhood models

- Users as n-dimensional vectors over items
- Items as m-dimensional vectors over users

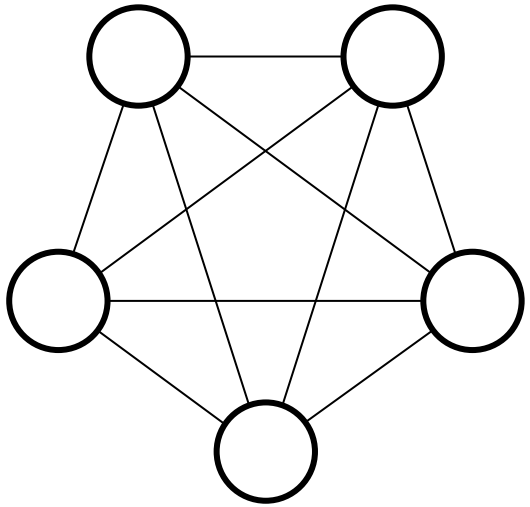
Unified space in latent factor models

- Users and items as k-dimensional vectors
- Straightforward predictions via dot products

Boltzmann machines

A network of units of “energy”

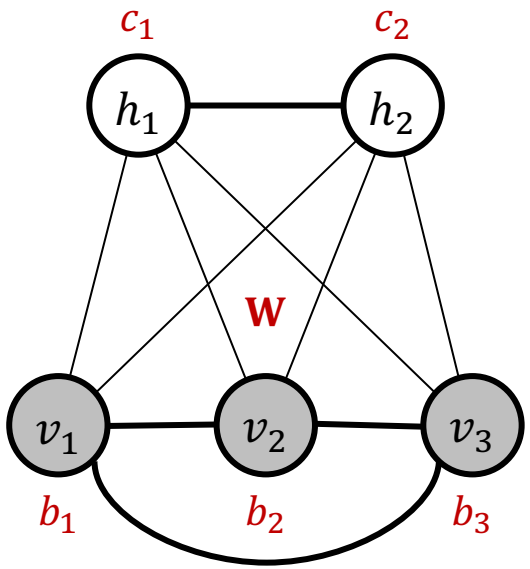
- Also an undirected graphical model



Boltzmann machines

Also a two-layer unsupervised neural network

- Goal is to reconstruct the input



$$\mathbf{c} \in \mathbb{R}^k$$

$$\mathbf{h} \in \{0,1\}^k$$

$$\mathbf{W} \in \mathbb{R}^{(n+k)^2}$$

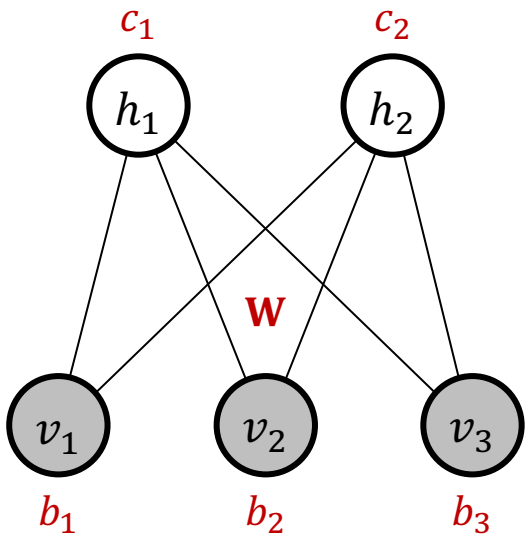
$$\mathbf{v} \in \{0,1\}^n$$

$$\mathbf{b} \in \mathbb{R}^n$$

Restricted Boltzmann machines

Restrict the connectivity to make learning easier

- No connections between visible or hidden units



$$\mathbf{c} \in \mathbb{R}^k$$

$$\mathbf{h} \in \{0,1\}^k$$

$$\mathbf{W} \in \mathbb{R}^{nk}$$

$$\mathbf{v} \in \{0,1\}^n$$

$$\mathbf{b} \in \mathbb{R}^n$$

RBMs as energy-based models

Associate a **scalar energy** to each configuration of the variables of interest ($\mathbf{v}$, $\mathbf{h}$, $\mathbf{W}$, $\mathbf{b}$, $\mathbf{c}$)

- $\mathrm{E}(\mathbf{v}, \mathbf{h}|\mathbf{W},\mathbf{b},\mathbf{c}) = -\mathbf{b}^T\mathbf{v} - \mathbf{c}^T\mathbf{h} - \mathbf{h}^T\mathbf{W}\mathbf{v}$

$$= -\sum_i b_i v_i - \sum_j c_j h_j - \sum_i \sum_j w_{ij} h_j v_i$$

Goal is to seek low-energy configurations

- Model parameters "force" states on $\mathbf{v}$ and $\mathbf{h}$

Learning RBMs

We want to minimize

- $\mathrm{E}(\mathbf{v}, \mathbf{h}|\mathbf{W},\mathbf{b},\mathbf{c}) = -\sum_i b_i v_i - \sum_j c_j h_j - \sum_i \sum_j w_{ij} h_j v_i$

Model parameters "force" states on $\mathbf{v}$ and $\mathbf{h}$

- $b_i < 0 \therefore v_i \to 0$; otherwise $v_i \to 1$
- $c_j < 0 \therefore h_j \to 0$; otherwise $h_j \to 1$
- $w_{ij} < 0 \therefore v_i$ or $h_j \to 0$; otherwise v_i and $h_j \to 1$

Learning RBMs

Energy modeled probabilistically (softmax)

- $p(\mathbf{v}, \mathbf{h}) = \frac{e^{-\mathbf{E}(\mathbf{v},\mathbf{h})}}{\sum_{\mathbf{v}',\mathbf{h}'} e^{-\mathbf{E}(\mathbf{v}',\mathbf{h}')}}$

Goal is to maximize the probability of the data

- $\text{argmax}_{\theta} \sum_{t=1}^{T} \log p(\mathbf{v}^{(t)})$ $\qquad \theta = \{\mathbf{W}, \mathbf{b}, \mathbf{c}\}$
- $\text{argmax}_{\theta} \sum_{t=1}^{T} \sum_{\mathbf{h}} \log p(\mathbf{v}^{(t)}, \mathbf{h})$

Learning RBMs

Solution: take the gradient wrt $\theta = \{\mathbf{W}, \mathbf{b}, \mathbf{c}\}$

- $\nabla_\theta \sum_{\mathbf{h}} \log p(\mathbf{v}^{(t)}, \mathbf{h}) = \nabla_\theta \sum_{\mathbf{h}} \log \frac{e^{-\mathbf{E}(\mathbf{v},\mathbf{h})}}{\sum_{\mathbf{v}',\mathbf{h}'} e^{-\mathbf{E}(\mathbf{v}',\mathbf{h}')}}$

Problem: intractable partition $Z = \sum_{\mathbf{v}',\mathbf{h}'} e^{-\mathbf{E}(\mathbf{v}',\mathbf{h}')}$

- Sum over exponentially many configurations (precisely, 2^{n+k} possible configurations)

From joints to conditionals

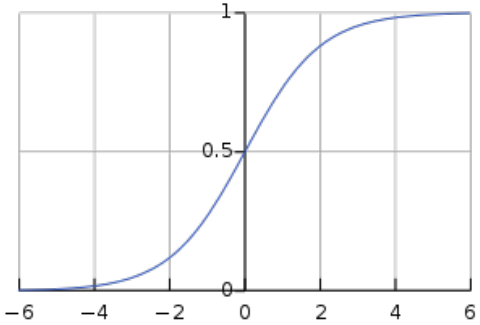
Can't efficiently estimate joint $p(\mathbf{v}, \mathbf{h})$

- Solution: approximate the gradient

Use conditionals $p(\mathbf{v}|\mathbf{h})$ and $p(\mathbf{h}|\mathbf{v})$ instead

- $p(\mathbf{h}|\mathbf{v}) = \prod_j p(h_j|\mathbf{v}) = \prod_j \sigma(c_j + \mathbf{v}^T \mathbf{W}_{:j})$
- $p(\mathbf{v}|\mathbf{h}) = \prod_i p(v_i|\mathbf{h}) = \prod_i \sigma(b_i + \mathbf{W}_{i:}\mathbf{h})$

Highly parallelizable



Contrastive divergence

Start with a training example $\mathbf{v}^{(0)}$

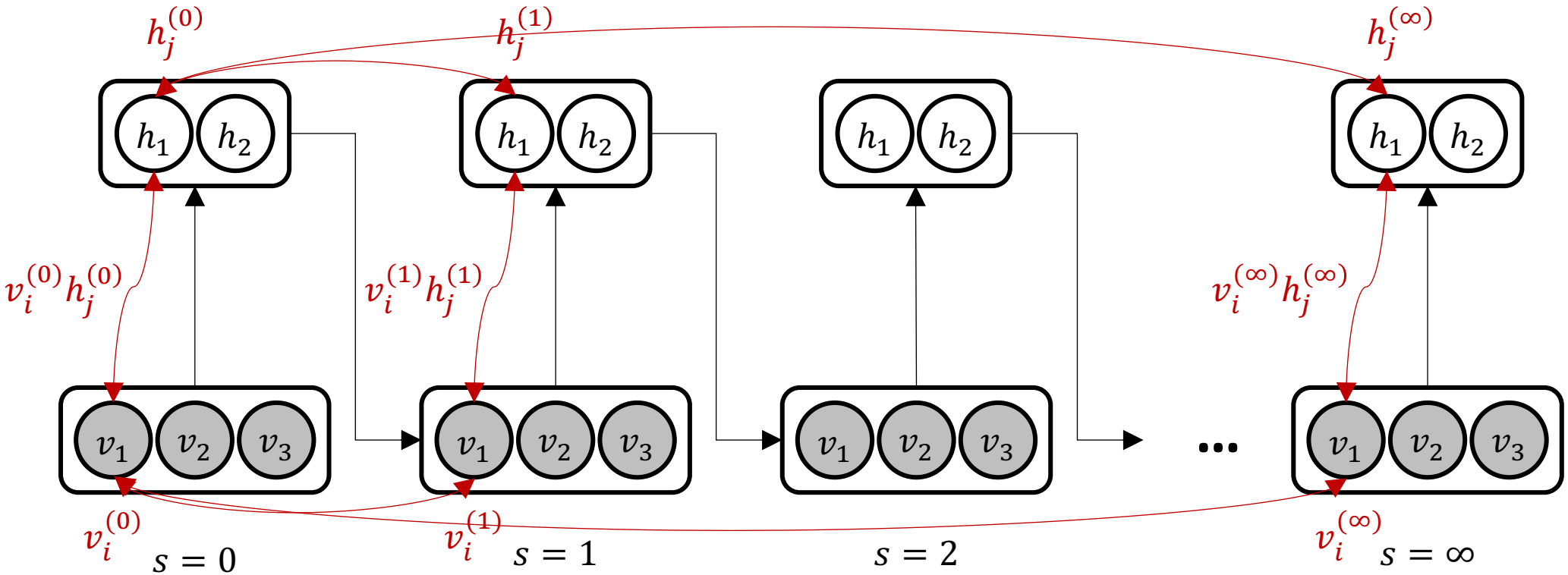
Sample hidden units: $h_j^{(0)} \sim \sigma\left(c_j + \mathbf{v}^{(0)T}\mathbf{W}_{:j}\right)$

Reconstruct visible units: $v_i^{(1)} \sim \sigma\left(b_i + \mathbf{W}_{i:}\mathbf{h}^{(0)}\right)$

Sample hidden units: $h_j^{(1)} \sim \sigma\left(c_j + \mathbf{v}^{(1)T}\mathbf{W}_{:j}\right)$

Update parameters: $\mathbf{W}$, $\mathbf{b}$, $\mathbf{c}$

Contrastive divergence



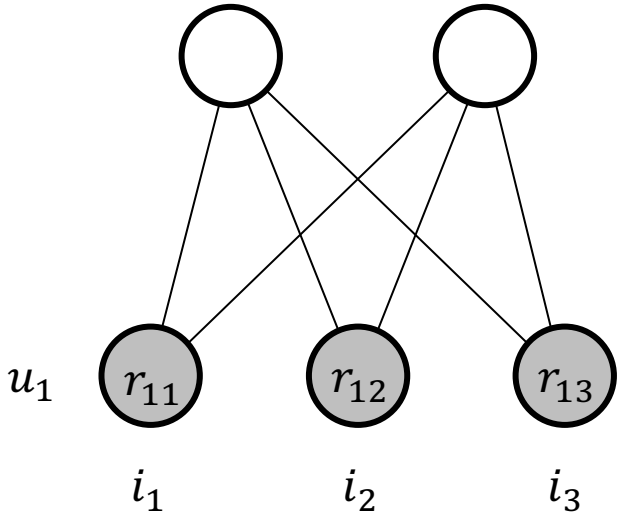
Contrastive divergence

Update parameters: $\mathbf{W}$, $\mathbf{b}$, $\mathbf{c}$

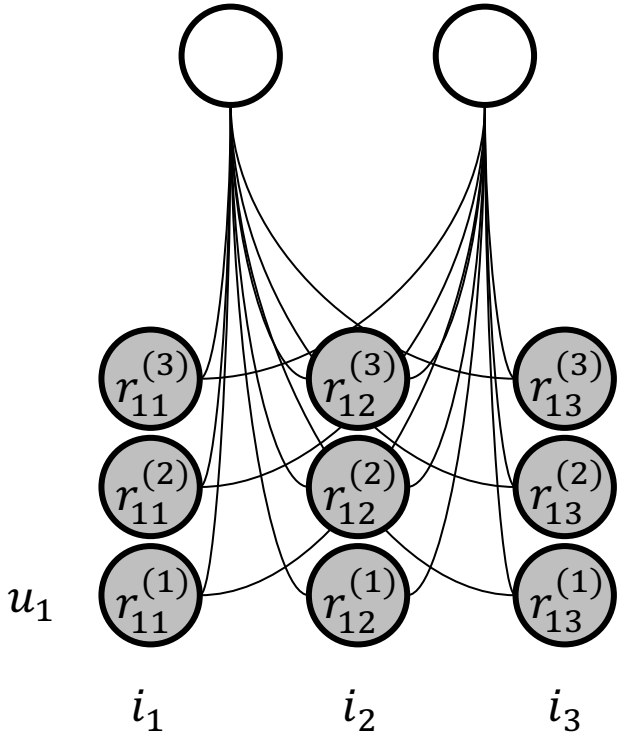
- $w_{ij} = w_{ij} + \alpha\left(v_i^{(0)} h_j^{(0)} - v_i^{(s)} h_j^{(s)}\right)$
- $b_i = b_i + \alpha(v_i^{(0)} - v_i^{(s)})$
- $c_j = c_j + \alpha(h_j^{(0)} - h_j^{(s)})$
- $s \to \infty$ approximates the true gradient
- $s = 1$ gives decent results in practice

Handling explicit feedack

Implicit feedback (e.g. clicks)



Explicit feedback (e.g. 3 stars)



Handling missing feedback

So far, we assumed complete feedback

- In reality, user preferences are extremely sparse

Solution: one user, one (shared) RBM

- Visible units over known feedback only
- Fixed number of hidden units (hyperparameter)
- Weights shared across RBMs

How well does it work?

RBMs work about as well as matrix factorization methods, but they give very different errors

- Opportunity for ensembling!

Winners at the Netflix Prize used multiple RBM models in their ensemble of over a hundred models

- Main models were matrix factorization and RBMs

Summary

RBMs automatically model a data distribution

- Effective results for collaborative filtering
- Can be stacked into deep belief networks

Trivial implementation may be expensive

- Speedup via parallelization
- Speedup via shortcutting contrastive divergence

References

RBMs for Collaborative Filtering (ICML 2007)

Ruslan Salakhutdinov, Andrity Mnih, Geoffrey Hinton

Restricted Boltzmann Machines

Ali Ghodsi

Neural Networks Course

Hugo Larochelle