Regularization Methods for the Restricted Bayesian Network BESOM

ICONIP 2016
2016-10-17

Yuuji Ichisugi and Takashi Sano
Artificial Intelligence Research Center (AIRC),
National Institute of Advanced Industrial Science and Technology(AIST)

Outline

- Our research goal:
  - Implement a cerebral cortex model
- Our developing model: BESOM model
- Local minimum problem of BESOM
- Regularization methods
- Network translation technique in order to use EM algorithm

Our research goals

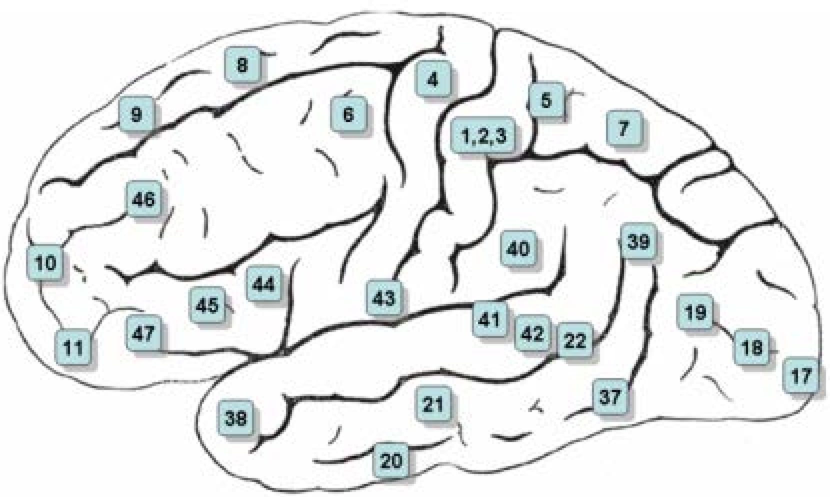
- Long term goal : **Human-like intelligence** by WBA approach

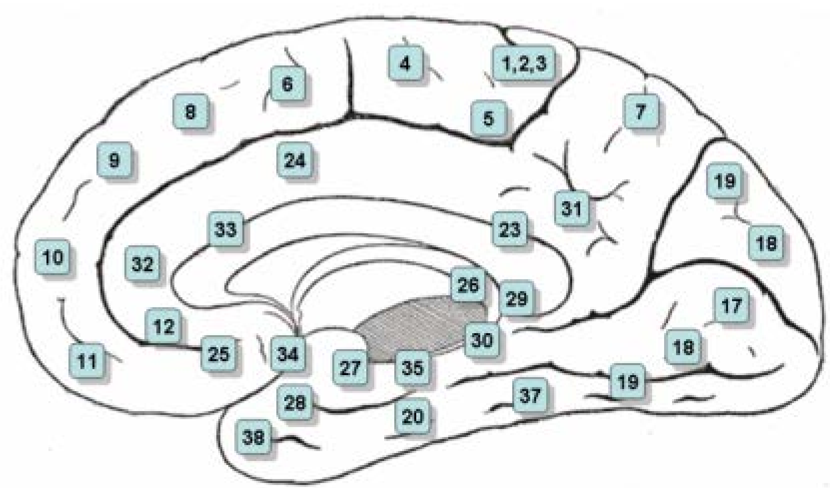
http://www.irasutoya.com/2015/05/ai.html

- Short term goals:
  - Implement a cerebral cortex model
    - Our working hypothesis :
      The cerebral cortex is a kind of Bayesian network
  - Implement **visual area, language area, motor area etc.** using the cerebral cortex model

Cerebral cortex

- Realizes human's intelligence.
  - Sensory, Motor, Language, ...
- It is important to reveal the information-processing principle of the cortex.





from http://en.wikipedia.org/wiki/Brodmann_area

Bayesian network models of cerebral cortex

- **Pattern recognition**
  [George and Hawkins 2005][Hasegawa and Hagiwara 2010]
- **Electrophysiological phenomena**
  [Lee and Mumford 2003] [Rao 2005] [Chikkerur, Serre, Tan and Poggio 2010][Hosoya 2010][Hosoya 2012]
- **Psychophysical phenomena**
  [Chikkerur, Serre, Tan and Poggio 2010]
- **Anatomical structures**
  [George and Hawkins 2005] [Ichisugi 2007] [Rohrbein, Eggert and Korner 2008] [Ichisugi 2011]
- **Motor areas**
  [Hosoya 2009]
- The others [Litvak and Ullman 2009][Ichisugi 2011]

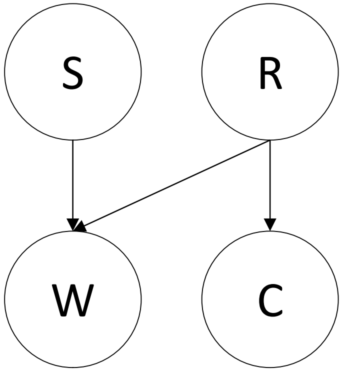
A cerebral cortex seems to be a huge Bayesian network with layered structure like Deep Learning.

What is Bayesian network?

– **Very efficient and expressive data structure for probabilistic knowledge.**

- If a joint probability table can be factored into small **conditional probability tables (CPTs)**, time and space complexity will decrease.

ex.: $P(S,W,R,C) = P(W \mid S,R)P(C \mid R)P(S)P(R)$



CPTs

P(S=yes)
0.2

P(R=yes)
0.02

S	R	P(W=yes|S,R)
no	no	0.12
no	yes	0.8
yes	no	0.9
yes	yes	0.98

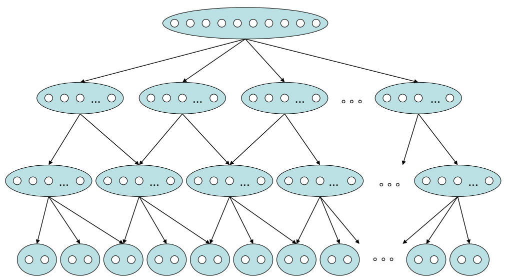
R	P(C=yes|R)
no	0.3
yes	0.995

Similarities between Cerebral Cortex and Bayesian network

- Asymmetric and bidirectional connections between lower and higher areas.
- Local and asynchronous communications.
- Non negative values.
- Normalization of values.
- Hebb's learning rule.
- Context dependent recognition.
- Behavior based on Bayesian Statistics.

Deep Learning using a Bayesian network is thought to be promising

- Because of its similarity to the human brain
- Inference in Bayesian networks can sometime be executed with low computational complexity



- Top-down information flow
- It is easy to build in prior knowledge about learning targets

BESOM (BidirEctional SOM) [Ichisug 2007]

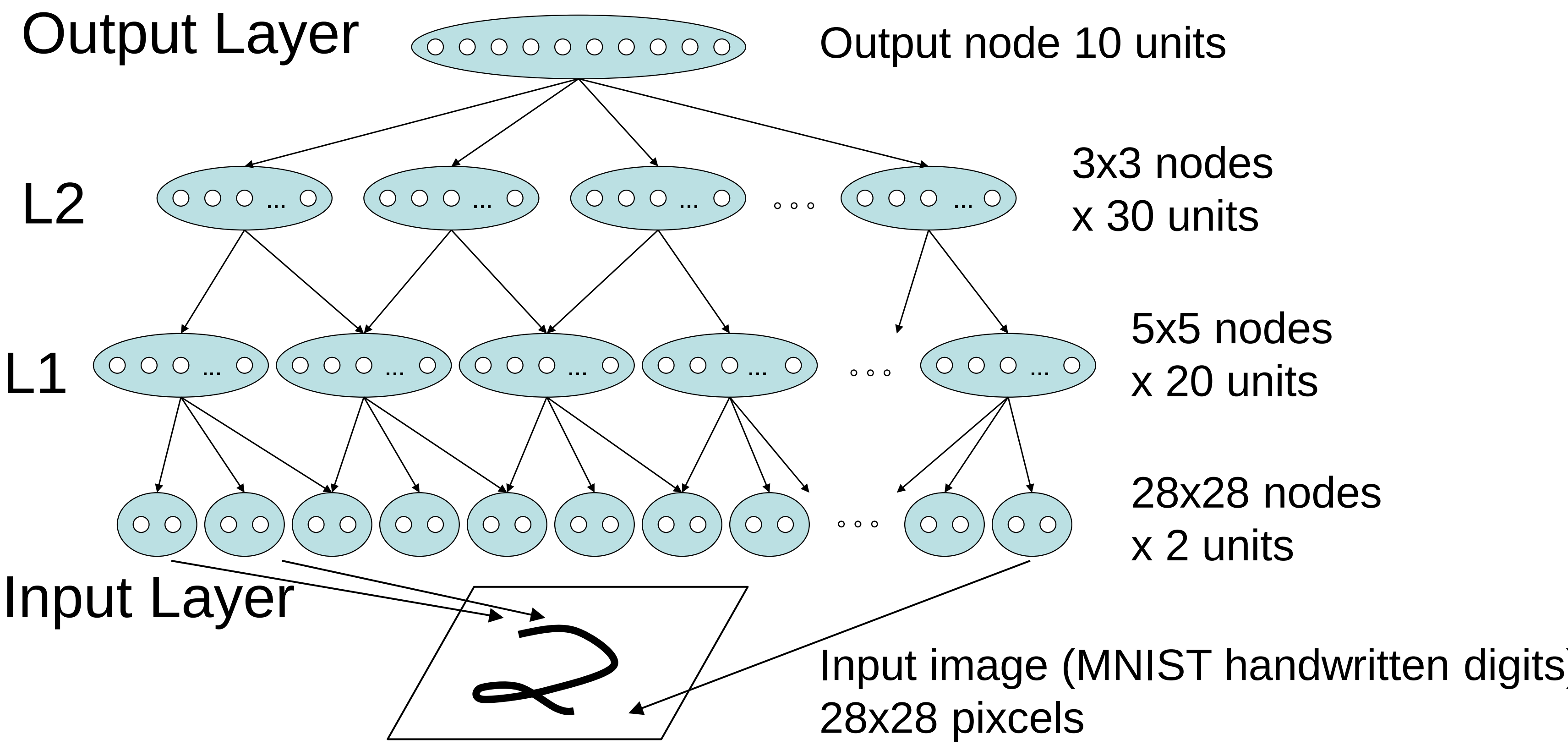
- A Bayesian network model of cerebral cortex
- Combination of Bayesian Networks, Deep Learning, Self-Organizing Maps and Independent Component Analysis
  - **Incomplete technology, however**
- Our goal:
  - **Scalability** of computation amount
  - **Scalability** of accuracy
  - **Usefulness** as a machine learning algorithm
  - **Plausibility** as a neuroscientific model

BESOM Ver.3.0 features

- Restricted Conditional Probability Tables:
$$P(x|u_1, \ldots, u_m) = \frac{1}{m} \Sigma_{k=1}^{m} P(x|u_k)$$
- Scalable recognition algorithm [Ichisugi, Takahashi 2015]
- Regularization methods:
  - **Win-rate penalty**
  - **Lateral-inhibition penalty**
  - Neighborhood learning
  - Edge selection

Today's topics

4 layer BESOM for supervised learning



Ovals are nodes (random variables)
White circles inside are units (possible values for the random variables)

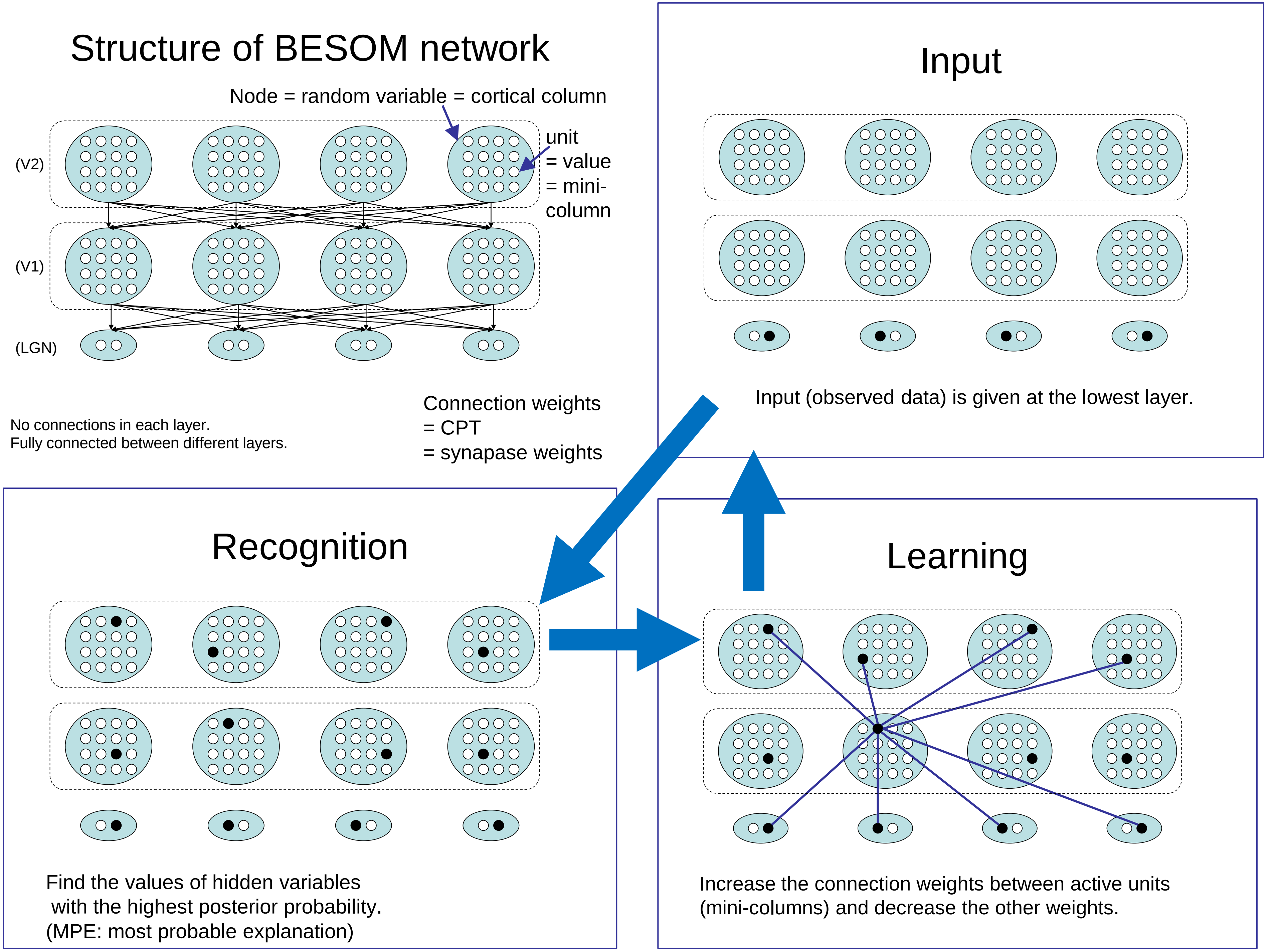
Objective of learning

- Calculate MAP estimator of the parameter θ

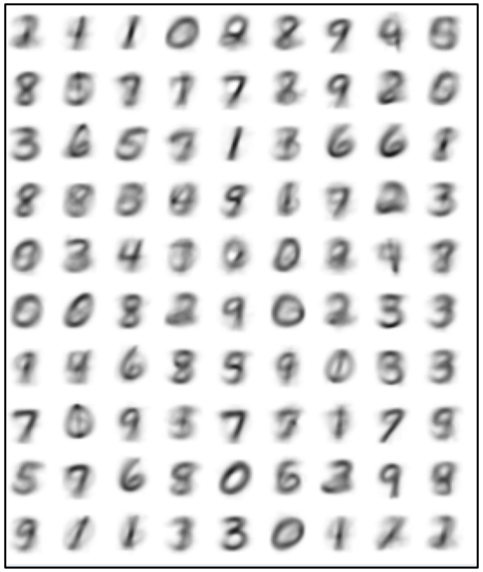
$$\theta^* = \arg\max_{\theta} \left[\prod_{i=1}^{t} P(\mathbf{i}(i) \mid \theta) \right] P(\theta)$$

$$= \arg\max_{\theta} \left[\prod_{i=1}^{t} \sum_{\mathbf{h}} P(\mathbf{h}, \mathbf{i}(i) \mid \theta) \right] P(\theta)$$

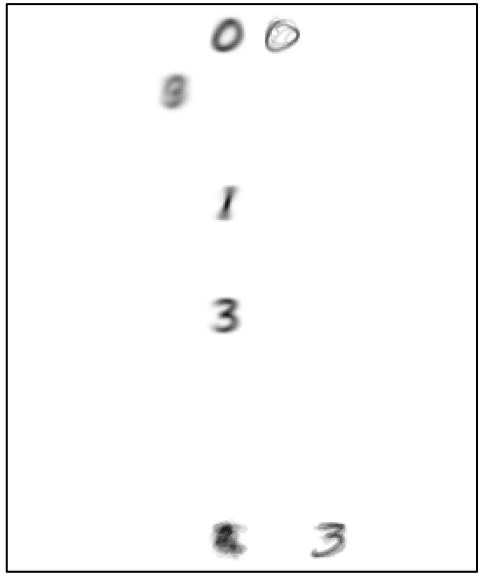
To estimate parameter , the online EM (Expectation-Maximization) algorithm or its approximation is used.


Structure of BESOM network
Node = random variable = cortical column
unit
= value
= mini-
column
(V2)
(V1)
(LGN)
No connections in each layer.
Fully connected between different layers.
Connection weights
= CPT
= synapase weights
Input
Input (observed data) is given at the lowest layer.
Recognition
Find the values of hidden variables
with the highest posterior probability.
(MPE: most probable explanation)
Learning
Increase the connection weights between active units
(mini-columns) and decrease the other weights.

Problem of utilization ratio of units



Learned **with proposed priors**.



Learned with **no priors**.
Most units never become active.

Seems to be very bad local minimum.
- Wastes units.
- Low recognition rates.
  63.6% MNIST

Each image is the mean image of inputs which activate the unit.
(Selected 10 units of L2 nodes are shown.)

White image indicate the unit **never become active**.

Win-Rate penalty

- All units should be **used evenly**.
- Penalties are imposed when the histograms of win-rates are difference from the uniform distributions.

$$P^{WinRate}(\theta) = \prod_{X \in \mathbf{X}} e^{-C^{WinRate} D_{KL}(Q(X)||P(X;\theta))} \quad (8)$$

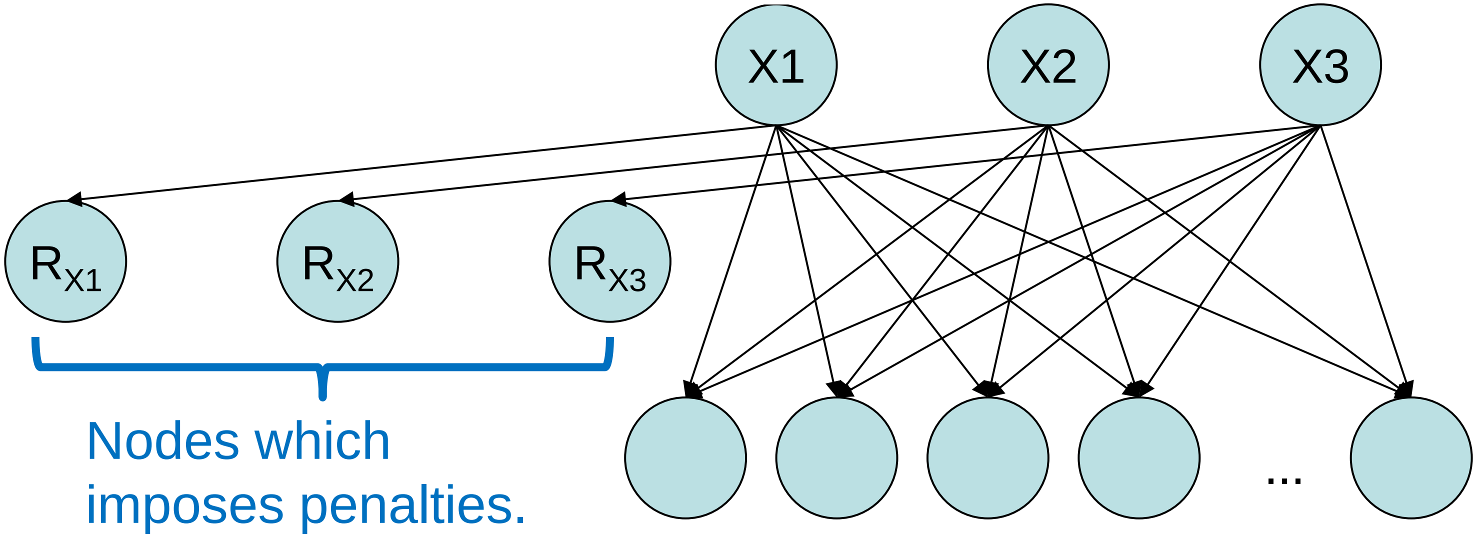
$$Q(X = x_i) = 1/s \quad (9)$$

Problem: When the parameter has a complex prior distribution, it is not obvious how to perform the EM algorithm efficiently.

Equivalent network

- **Fortunately**, the network with win-rate penalty can be expressed as **an approx. equivalent network without prior**.
  - Then, EM is straightforwardly applicable.

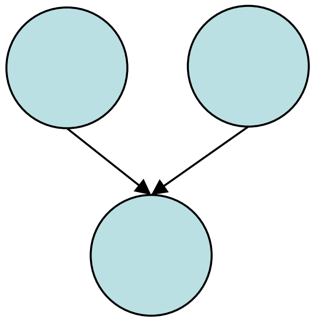
$$P(R_X = 1 | X = x; \theta) \;\; = \;\; e^{-(1/t) C^{WinRate} R(x;\theta)}, \quad R(x;\theta) = \frac{Q(x)}{P(x;\theta)} \log \frac{Q(x)}{P(x;\theta)}$$



Nodes which imposes penalties.

Lateral-Inhibition penalty

- Nodes which shares the same child nodes should be **independent**.
  - Otherwise, redundant representation is acquired by learning.



- Penalties are imposed when designated pairs of nodes are **not independent**.

$$P^{Lateral}(\theta) = \prod_{(U,V)\in L} e^{-C^{Lateral} I(U,V;\theta)} \quad (17)$$

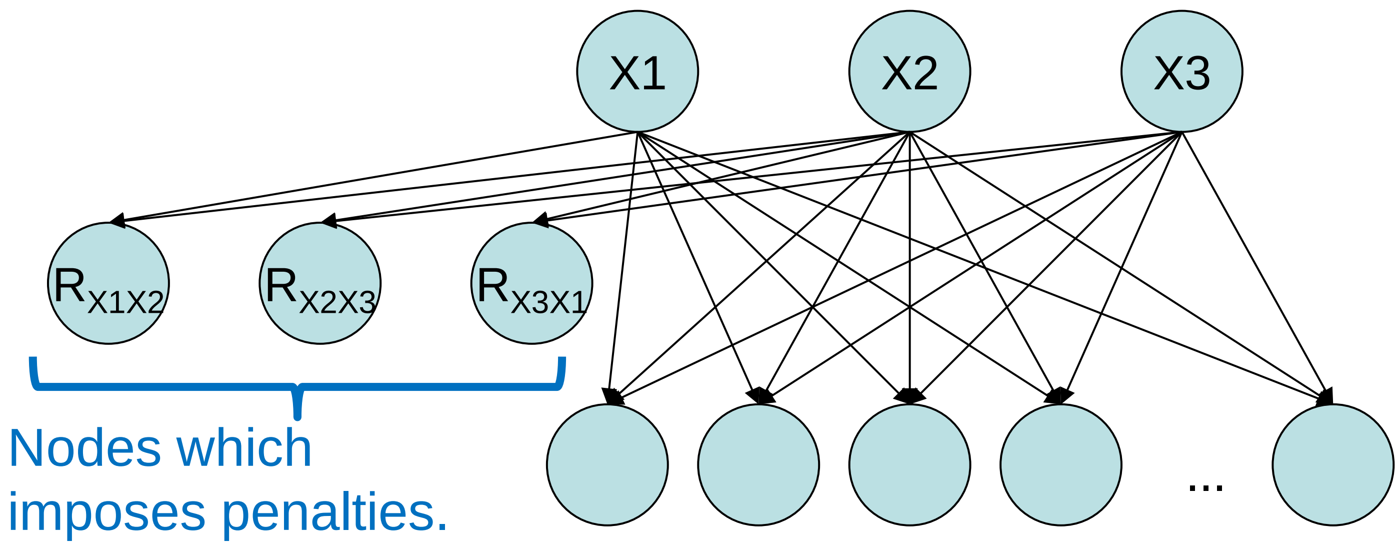
$$I(U,V;\theta) = \sum_{u} \sum_{v} P(u,v;\theta) \log \frac{P(u,v;\theta)}{P(u;\theta)P(v;\theta)} \quad (18)$$

Equivalent network

- This penalty can also be represented by **an approximately equivalent network without prior**.

$$P(R_{UV} = 1|u, v; \theta) \;=\; e^{-(1/t)C^{Lateral}R(u,v;\theta)},$$

$$
\begin{aligned}
& R(u, v) \\
= \;& s \, \frac{P(u, v)}{P(u)P(v)} \log \frac{P(u, v)}{P(u)P(v)} \\
= \;& (P(u|v)/P(u)) \log P(u|v)/P(u)
\end{aligned}
$$



Evaluation Result (MNIST)

	With Win-Rate Penalty	Without Win-Rate Penalty
With Lateral-Inhibition	**80.6%**	**81.8%**
Without Lateral-Inhibition	**82.2%**	63.6%

For both penalties, the recognition rate was higher than when no penalties were applied.

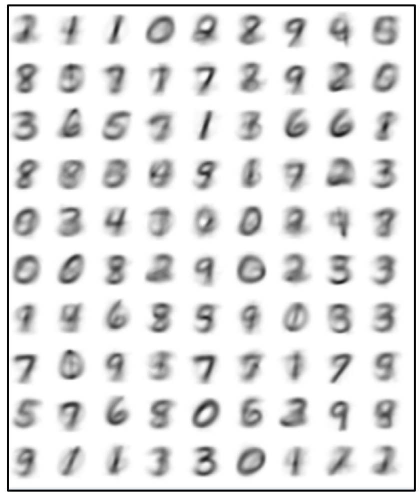
This result also shows that two prior distribution can be applied simultaneously; however, it does not show the best accuracy in this case.

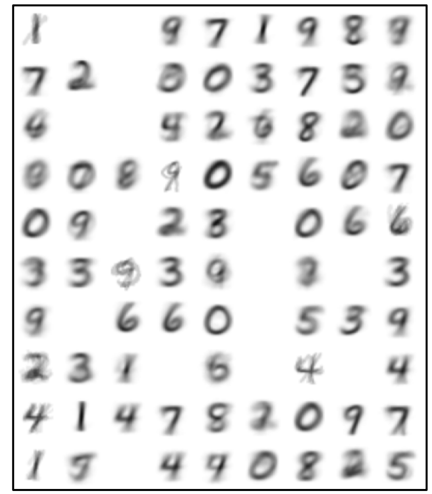
Status of utilization of units

With Win-Rate Penalty

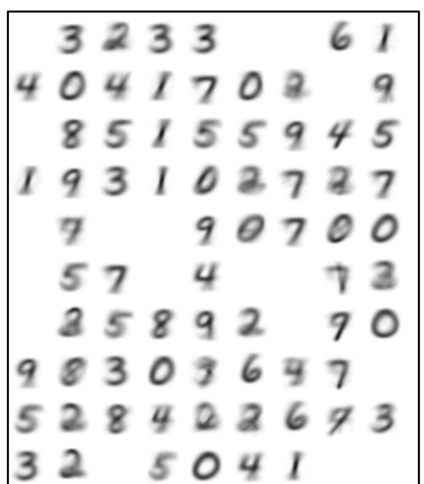
Without Win-Rate Penalty

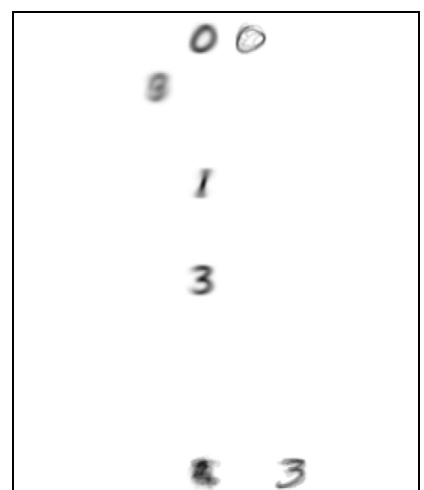
With Lateral-Inhibition penalty





Without Lateral-Inhibition penalty





Conclusion

- Two regularization methods for parameter learning of layered Bayesian networks like deep learning are proposed.
  - Win-Rate penalty and Lateral Inhibition penalty
  - Standard EM can be used for learning by network translation technique
- They may alleviate both local minima and overfitting problems.