# Perspectives on Probabilistic Graphical Models

Dong Liu

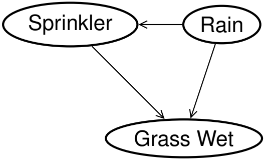
*Information Science and Engineering*
*KTH - Royal Institute of Technology*

Profile page: https://firsthandscientist.github.io/

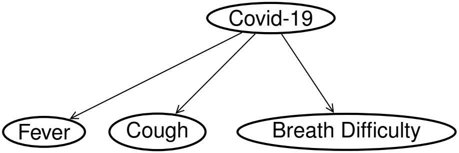
Slide is available at: https://github.com/FirstHandScientist/phdthesis

Why are Probabilistic Graphical Models interested?

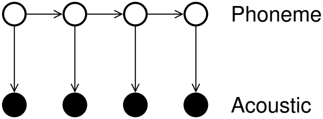
# DIRECTED GRAPH REPRESENTATION



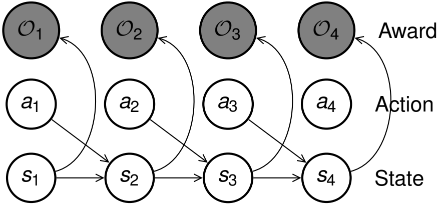
Is the sprinkler working?



Is the person get contiguous by COVID?

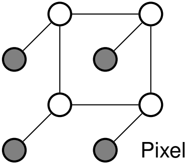


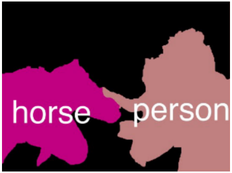
Speech Recognition



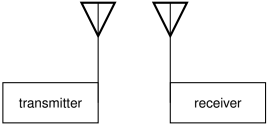
Control, reinforcement learning

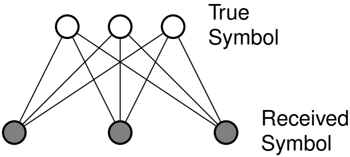
# Undirected Graph Representations





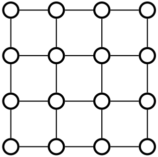
Vision Perception





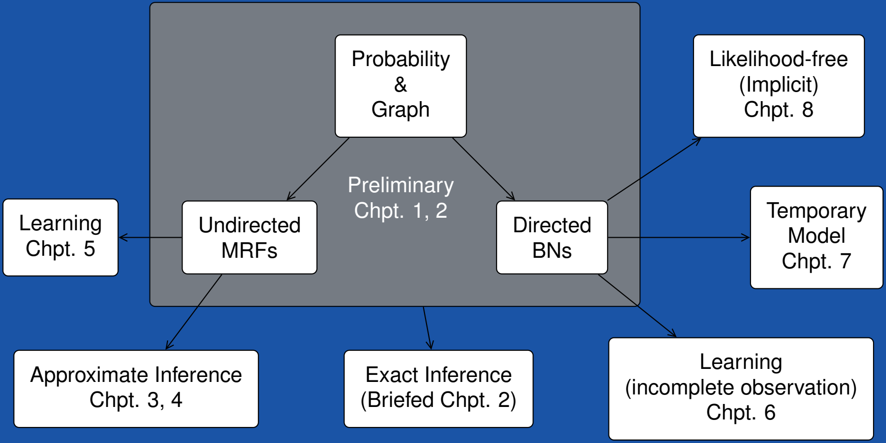
Digital

communication



- Error-control codes
- Computational biology
- Natural language processing
- etc.

# A Guide to This Dissertation



# WHAT ARE PROBABILISTIC GRAPHICAL MODELS

Informally...

- attributes of our interests in a system → variable nodes
- relationship of these factors → structures of a graph

Intrinsic property: **reasoning with uncertainty**

A directed/undirected graph encoding dependencies/indepedencies of distribution $p(\boldsymbol{x};\boldsymbol{\theta})$:

- A BN/Generative model is a directed graph
  - $p(\boldsymbol{x};\boldsymbol{\theta}) = \prod_{n=1}^{N} p(x_n|\mathcal{P}(x_n))$
  - $\mathcal{P}(\cdot)$ are parent nodes
  - the local functions are proper distributions
- An MRF denoted by an undirected graph $\mathcal{G}(\mathcal{V},\mathcal{E})$
  - The probability distribution (Gibbs distribution) is $p(\boldsymbol{x};\boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})} \prod_{a\in\mathcal{I}} \psi_a(\boldsymbol{x}_a;\boldsymbol{\theta}_a)$
  - $a$ indexes potential functions $\mathcal{I} = \{\psi_A, \psi_B, \cdots, \psi_M\}$
  - $Z(\boldsymbol{\theta}) = \sum_{\boldsymbol{x}} \prod_a \psi_a(\boldsymbol{x}_a;\boldsymbol{\theta}_a)$.

# What are Probabilistic Graphical Models

Informally...

- attributes of our interests in a system → variable nodes
- relationship of these factors → structures of a graph

Intrinsic property: **reasoning with uncertainty**

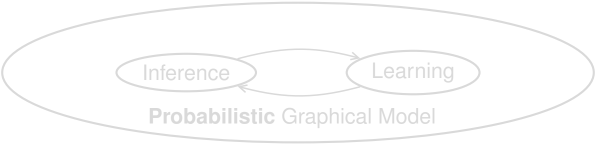
A directed/undirected graph encoding dependencies/indepedencies of distribution $p(\boldsymbol{x}; \boldsymbol{\theta})$:

- A BN/Generative model is a directed graph
  - $p(\boldsymbol{x}; \boldsymbol{\theta}) = \prod_{n=1}^{N} p(x_n | \mathcal{P}(x_n))$
  - $\mathcal{P}(\cdot)$ are parent nodes
  - the local functions are proper distributions
- An MRF denoted by an undirected graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$
  - The probability distribution (Gibbs distribution) is $p(\boldsymbol{x}; \boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})} \prod_{a \in \mathcal{I}} \psi_a(\boldsymbol{x}_a; \boldsymbol{\theta}_a)$
  - $a$ indexes potential functions $\mathcal{I} = \{\psi_A, \psi_B, \cdots, \psi_M\}$
  - $Z(\boldsymbol{\theta}) = \sum_{\boldsymbol{x}} \prod_a \psi_a(\boldsymbol{x}_a; \boldsymbol{\theta}_a)$.

# USAGE OF GRAPHICAL MODELS

- The common inference problems:
  - Computing the likelihood of observed data.
  - Computing the marginals distribution $p(\boldsymbol{x}_A)$ over particular subset $A \subset \mathcal{V}$ of nodes
  - Computing the conditional distribution $p(\boldsymbol{x}_A|\boldsymbol{x}_B)$,
  - Computing the partition function or the Helmholtz free energy (for MRFs)
- Learning:
  - To model or determine $p(\boldsymbol{x}; \boldsymbol{\theta})$.

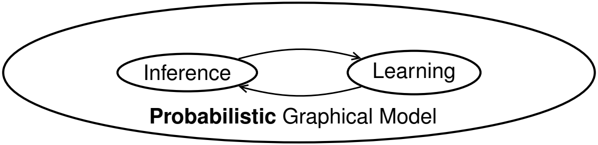
Two key components interacting with each other:



# USAGE OF GRAPHICAL MODELS

- The common inference problems:
  - Computing the likelihood of observed data.
  - Computing the marginals distribution $p(\boldsymbol{x}_A)$ over particular subset $A \subset \mathcal{V}$ of nodes
  - Computing the conditional distribution $p(\boldsymbol{x}_A|\boldsymbol{x}_B)$,
  - Computing the partition function or the Helmholtz free energy (for MRFs)
- Learning:
  - To model or determine $p(\boldsymbol{x}; \boldsymbol{\theta})$.

Two key components interacting with each other:



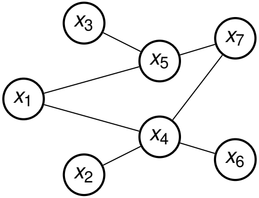
# WHAT IS THE STATE OF $x$?

## A TOY EXAMPLE

Assume that we are interested into the state of node $i$ in an MRF, it can be answered by

- the probability $p(x_i)$, or
- an empirical version, a collection of samples $\{x_i^n\}_{n=1}^N$

It is similar for the case when $\boldsymbol{x}$ is of interests, instead of $x_i$.

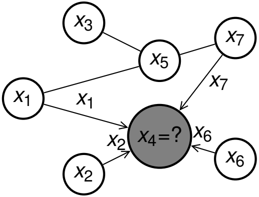


what is the state of $x_4$

# WHAT IS THE STATE OF $x$?

Gibbs sampling: let us guess by sampling

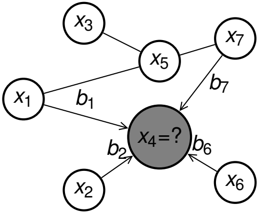
Sample iteratively: $x_i \sim p(x_i|\boldsymbol{x}_{-i}) \sim p(x_i, \boldsymbol{x}_{-i})$



Queries by collected samples $\{\boldsymbol{x}^n\}_1^N$.

Mean Field and BP: *message in form of sample values → message in form of belief*

Propagating beliefs iteratively



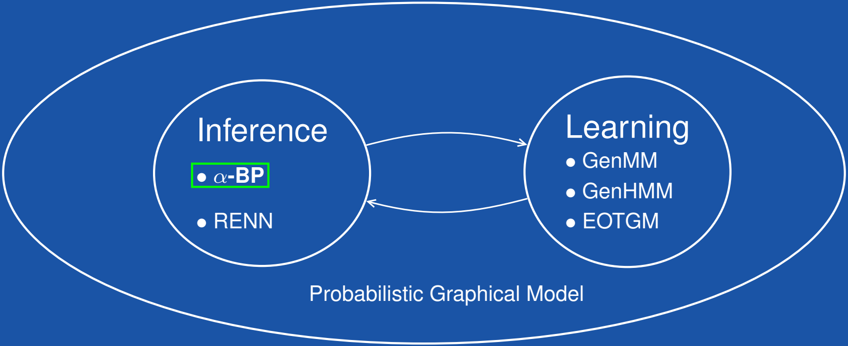
Queries by collected samples $\{b_i\}$.

---

Intuition from *Gibbs (variational) free energy*

$$F_V(b) = \mathrm{KL}(b(\boldsymbol{x})||p(\boldsymbol{x};\boldsymbol{\theta})) - \log Z(\boldsymbol{\theta})$$

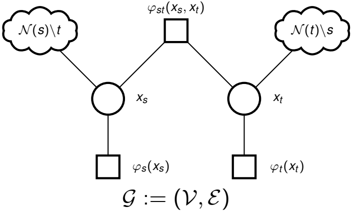
with trial $b(\boldsymbol{x})$. Instance: Bethe free energy.


Inference
• $\alpha$-BP
• RENN
Learning
• GenMM
• GenHMM
• EOTGM
Probabilistic Graphical Model

# ALTERNATIVE VEIW OF BP: $\alpha$-BP

Input:

- A pairwise Markov random field: $p(\boldsymbol{x}) \propto \prod_{s \in \mathcal{V}} \varphi_s(x_s) \prod_{(s,t) \in \mathcal{E}} \varphi_{st}(x_s, x_t)$
- A trial distribution: $q(\boldsymbol{x}) \propto \prod_{s \in \mathcal{V}} \tilde{\varphi}_s(x_s) \prod_{(s,t) \in \mathcal{E}} \tilde{\varphi}_{st}(x_s, x_t)$ with factorization $\tilde{\varphi}_{s,t}(x_s, x_t) := m_{st}(x_t) m_{ts}(x_s)$
- A metric: $\alpha$-Divergence

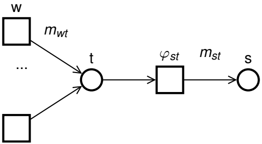

$\mathcal{N}(s)\backslash t$
$\varphi_{st}(x_s, x_t)$
$\mathcal{N}(t)\backslash s$
$x_s$
$x_t$
$\varphi_s(x_s)$
$\varphi_t(x_t)$
$\mathcal{G} := (\mathcal{V}, \mathcal{E})$

Approximate local minimization:

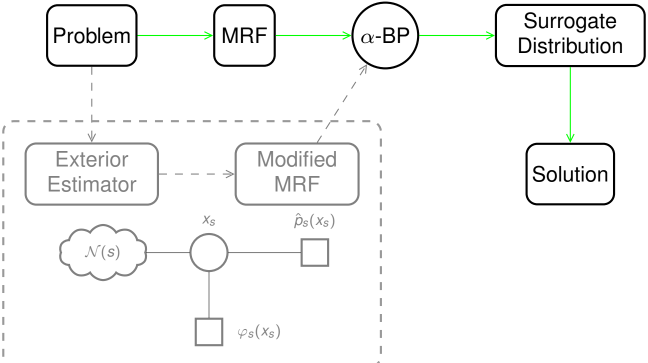
- Direct minimization: $\underset{\tilde{\varphi}_{ts}^{\text{new}}(x_t, x_s)}{\operatorname{argmin}} \mathcal{D}_{\alpha_{ts}}(p^{\backslash(t,s)}(\boldsymbol{x})\varphi_{ts}(x_t, x_s) \| q^{\backslash(t,s)}(\boldsymbol{x})\tilde{\varphi}_{ts}^{\text{new}}(x_t, x_s))$
- Local minimization: $\underset{\tilde{\varphi}_{ts}^{\text{new}}(x_t, x_s)}{\operatorname{argmin}} \mathcal{D}_{\alpha_{ts}}(q^{\backslash(t,s)}(\boldsymbol{x})\varphi_{ts}(x_t, x_s) \| q^{\backslash(t,s)}(\boldsymbol{x})\tilde{\varphi}_{ts}^{\text{new}}(x_t, x_s))$, say you are updating $\tilde{\varphi}_{ts}^{\text{new}}(x_t, x_s) = m_{ts}^{\text{new}}(x_s) m_{st}(x_t)$

Definition of $\alpha$-divergence $\mathcal{D}_\alpha(p\|q) = \frac{\sum_{\boldsymbol{x}} \alpha p(\boldsymbol{x}) + (1-\alpha) q(\boldsymbol{x}) - p(\boldsymbol{x})^\alpha q(\boldsymbol{x})^{1-\alpha}}{\alpha(1-\alpha)}$

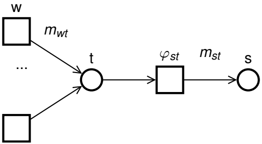
Updating message via $\alpha$-BP:



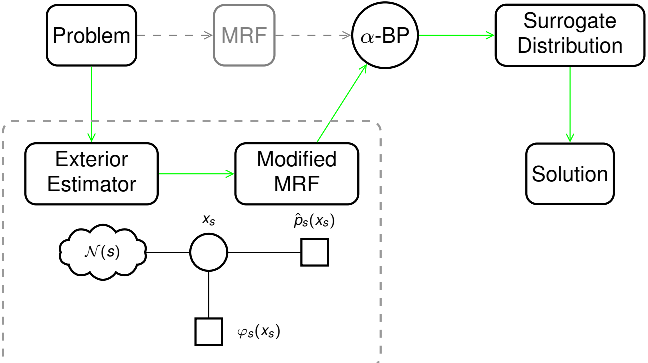
$$\underbrace{m_{ts}^{\text{new}}(x_s)}_{\text{new msg. to s}} \propto \underbrace{m_{ts}(x_s)^{1-\alpha_{ts}}}_{\text{old msg. to s}} \left[\sum_{x_t} \varphi_{ts}(x_t, x_s)^{\alpha_{ts}} \underbrace{m_{st}(x_t)^{1-\alpha_{ts}}}_{\text{old msg. to t}} \underbrace{\varphi_t(x_t) \prod_{w \in \mathcal{N}(t) \setminus s} m_{wt}(x_t)}_{\text{msg. from variable node } t \text{ to factor } \varphi_{st}}\right].$$



Updating message via $\alpha$-BP:



$$\underbrace{m_{ts}^{\text{new}}(x_s)}_{\text{new msg. to s}} \propto \underbrace{m_{ts}(x_s)^{1-\alpha_{ts}}}_{\text{old msg. to s}} \left[\sum_{x_t} \varphi_{ts}(x_t, x_s)^{\alpha_{ts}} \underbrace{m_{st}(x_t)^{1-\alpha_{ts}}}_{\text{old msg. to t}} \underbrace{\varphi_t(x_t) \prod_{w \in \mathcal{N}(t) \setminus s} m_{wt}(x_t)}_{\text{msg. from variable node } t \text{ to factor } \varphi_{st}}\right].$$



# Insights of $\alpha$-BP

## Connection to standard BP

- $\alpha \to 0$
- $\alpha$-divergence reduce KL-divergence
- Update rule of $\alpha$-BP reduces to $m_{ts}^{\text{new}}(x_s) \propto \sum_{x_t} \varphi_{st}(x_s, x_t)\varphi_t(x_t) \prod_{w \in \mathcal{N}(t)\backslash s} m_{wt}(x_t)$, which is standard BP update rule

## Convergence

For an arbitrary pairwise Markov random field over binary variables, if the largest singular value of matrix $\boldsymbol{M}(\boldsymbol{\alpha}, \boldsymbol{\theta})$ is less than one, $\alpha$-BP converges to a fixed point. The associated fixed point is unique.

See Corollary 3.1 for relaxed condition where singular value computation is avoided.

## What does that mean

- You can safely use $\alpha$-BP as an alternative to (loopy) BP
- You can use matrix $\boldsymbol{M}$ to check if you are guaranteed to get stable solution from $\alpha$-BP

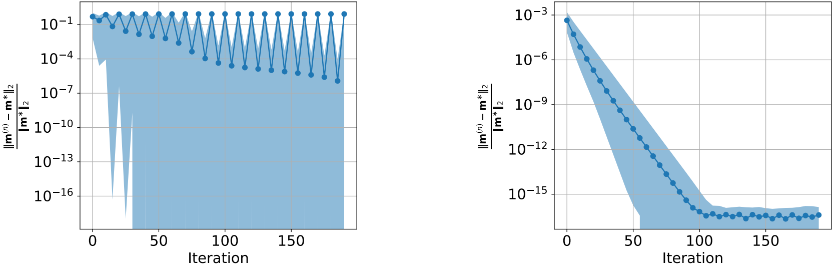
For binary, symmetric log-potentials

$$\begin{aligned} \varphi_{st}(x_s, x_t) &= \exp\left\{\theta_{st}(x_s, x_t)\right\}, \\ \varphi_s(x_s) &= \exp\left\{\theta_s(x_s)\right\}. \end{aligned}$$

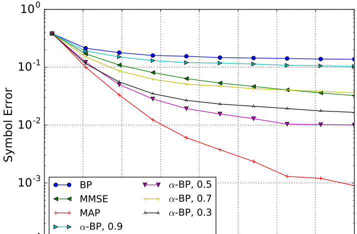
Matrix $\boldsymbol{M}(\boldsymbol{\alpha}, \boldsymbol{\theta})$, size $|\vec{\mathcal{E}}| \times |\vec{\mathcal{E}}|$, indexed by directed edges $(t \to s)$, as

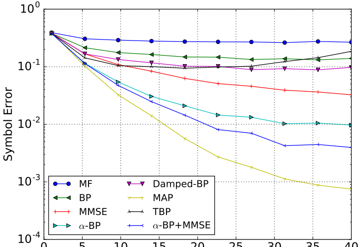
$$M_{(t\to s),(u\to v)} = \begin{cases} |1-\alpha_{ts}|, & u=t, v=s, \\ |1-\alpha_{ts}| \tanh|\alpha_{ts}\theta_{ts}|, & u=s, v=t, \\ \tanh|\alpha_{ts}\theta_{ts}|, & u \in \mathcal{N}(t)\backslash s, v=t, \\ 0, & \text{otherwise}. \end{cases}$$

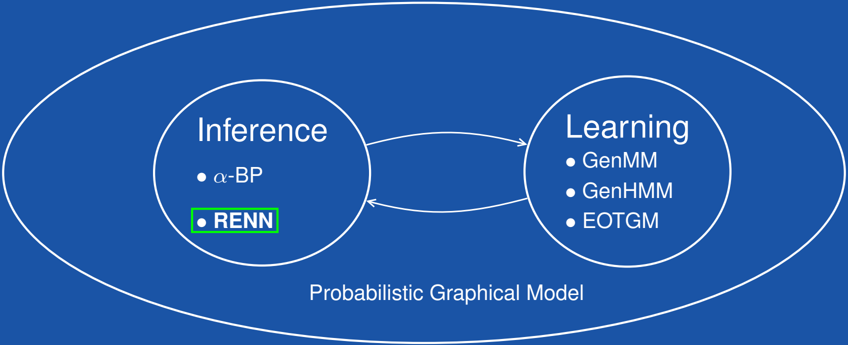
# Some Numerical Results



Numerical illustration of convergence, with normalized error $\|\boldsymbol{m}^{(n)} - \boldsymbol{m}^*\|_2 / \|\boldsymbol{m}^*\|_2$ versus the number of iterations. Number of nodes $N = 16$. Blue region denotes the range from minimum to maximum of the normalized error of 100 graph realizations, whereas the curve stands for mean error of the 100 realized graphs.



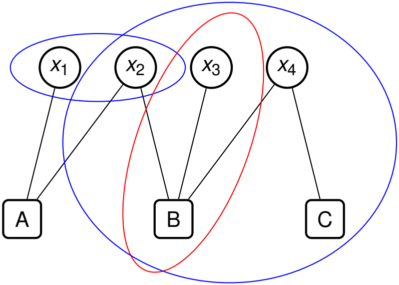



Inference
• $\alpha$-BP
• **RENN**
Learning
• GenMM
• GenHMM
• EOTGM
Probabilistic Graphical Model

# CONTINUING: WHAT IS THE STATE OF $x$?

## YEDIDIA, FREEMAN, WEISS: A STEP TO GENERALIZATION

**Message among variables & factors** → **message among regions** Maybe I should combine this slide to the next one



- Blue region: valid region
- Red region: invalid region

Generalized belief propagation (GBP) generalizes loopy BP

- usual better approximation than LBP
- higher complexity
- sensitive to scheduling of region messages

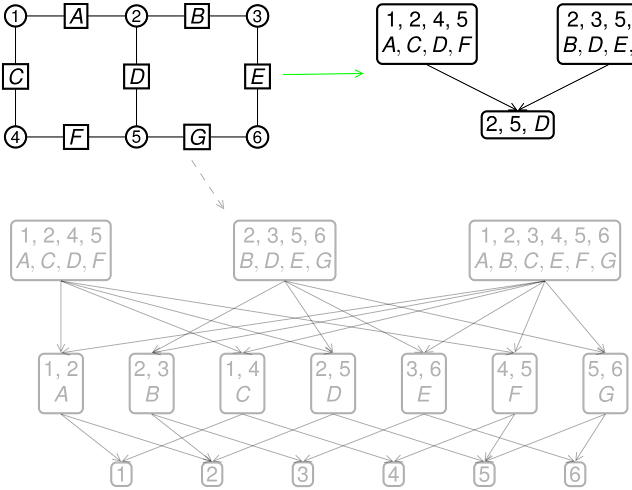
Corresponding to minimization of approximated **variational free energy** $F_v(b)$ **with trial** $b$ **including** $\{b_R\}$.

A *region* $R$ is a set $V_R$ of variables nodes and a set $A_R$ of factor nodes, such that if a factor node '$a$' belongs to $A_R$, all the variables nodes neighboring $a$ are in $V_R$.

# An Toy Example of Region Graphs

Factor graph representation of MRF (2-by-3 grid) with factor nodes.

MRF → region graph:



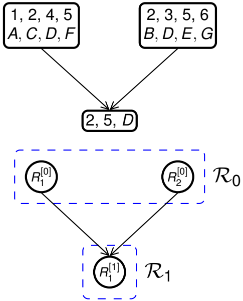
- Clustering nodes
- Valid regions
- Energy Calculated per region
- Msg. Scheduling
- ...
- See Section 4.1

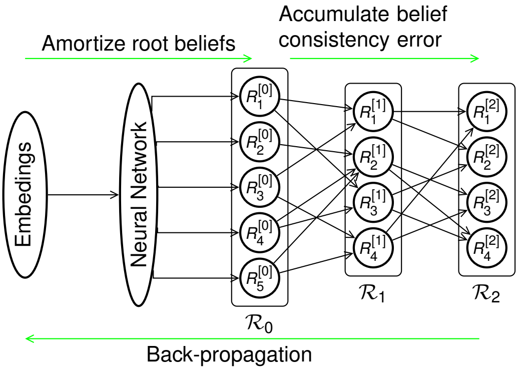
# RENN: Region-based Energy Neural Network

The **region-based free energy** of a region graph is

$$F_R(\mathcal{B};\boldsymbol{\theta}) = \sum_{R\in\mathcal{R}} \underbrace{c_R}_{\text{Counting number}} \sum_{\boldsymbol{x}_R} \underbrace{b_R(\boldsymbol{x}_R)}_{\text{Belief on region R}} \Big( \underbrace{E_R(\boldsymbol{x}_R;\boldsymbol{\theta}_R)}_{\text{Region average energy}} + \ln b_R(\boldsymbol{x}_R)\Big),$$

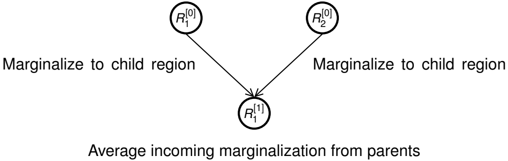
- counting number: balance the contribution of each region
- region average energy: $-\sum_{a\in A_R} \ln \varphi_a(\boldsymbol{x}_a;\boldsymbol{\theta}_a)$





# RENN

Non-root belief:



Marginalize to child region

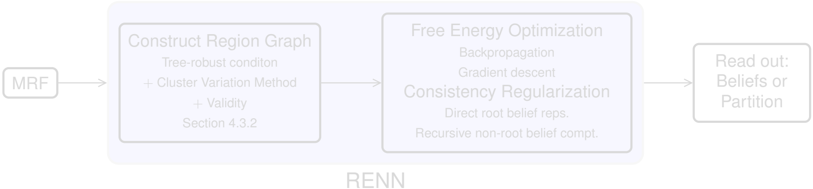
Marginalize to child region

Average incoming marginalization from parents

Objective of RENN:

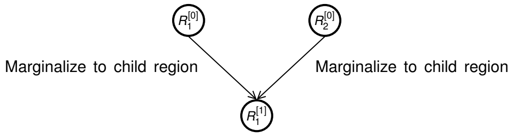
$$\min_{\substack{\text{parameter}\\ \text{of NN}}} \underbrace{\textbf{region-based free energy}(F_R)}_{\text{Assumulated average of all regions}} + \underbrace{\textbf{panelty on belief consistency}}_{\text{Recursively computed via levels of region graph}}$$

RENN Inference:



# RENN

Non-root belief:



$R_1^{[0]}$

$R_2^{[0]}$

Marginalize to child region

Marginalize to child region

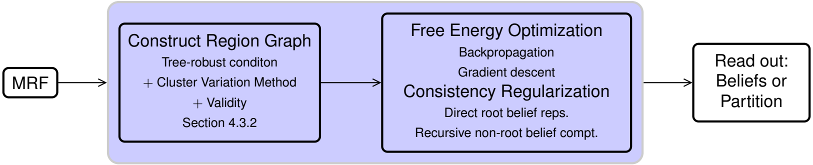
$R_1^{[1]}$

Average incoming marginalization from parents

Objective of RENN:

$$\min_{\substack{\text{parameter}\\ \text{of NN}}} \underbrace{\textbf{region-based free energy}(F_R)}_{\text{Assumulated average of all regions}} + \underbrace{\textbf{panelty on belief consistency}}_{\text{Recursively computed via levels of region graph}}$$

RENN Inference:



MRF

Construct Region Graph

Tree-robust conditon

+ Cluster Variation Method

+ Validity

Section 4.3.2

Free Energy Optimization

Backpropagation

Gradient descent

Consistency Regularization

Direct root belief reps.

Recursive non-root belief compt.

Read out: Beliefs or Partition

RENN

# INSIGHTS OF RENN

## Generalization

Bethe free energy can be recovered from region-based free energy:

- two-level region graph representation
- constraint that each region can contain at most one factor node

Section 4.2.1

## Attributes of RENN

- RENN requires neither sampling technique nor training data (ground-truth marginal probabilities) in performing inference tasks;
- RENN does gradient descent w.r.t. its neural network parameter instead of iterative message-passing, and returns approximation of marginal probabilities and partition estimation in one-shot
- No message propagation, thus no need of message scheduling
- Competitive performance and efficiency

# Some Numerical Comparisons

Ising model: $p(\boldsymbol{x};\boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})} \exp\left(\sum_{(i,j)\in\mathcal{E}_F} J_{ij}x_ix_j + \sum_{i\in\mathcal{V}} h_ix_i\right)$, $\boldsymbol{x} \in \{-1,1\}^N$,

- $J_{ij}$ is the pairwise log-potential between node $i$ and $j$, $J_{ij} \sim \mathcal{N}(0,1)$
- $h_i$ is the node log-potential for node $i$, $h_i \sim \mathcal{N}(0,\gamma^2)$

Inference on grid graph ($\gamma = 0.1$).

| Metric | $n$ | Mean Field | Loopy BP | Damped BP | GBP | Inference Net | RENN |
|---|---|---|---|---|---|---|---|
| $\ell_1$ error | 25 | 0.271 ± 0.051 | 0.086 ± 0.078 | 0.084 ± 0.076 | 0.057 ± 0.024 | 0.111 ± 0.072 | **0.049** ± 0.078 |
| | 100 | 0.283 ± 0.024 | 0.085 ± 0.041 | 0.062 ± 0.024 | 0.064 ± 0.019 | 0.074 ± 0.034 | **0.025** ± 0.011 |
| | 225 | 0.284 ± 0.019 | 0.100 ± 0.025 | 0.076 ± 0.025 | 0.073 ± 0.013 | 0.073 ± 0.012 | **0.046** ± 0.011 |
| | 400 | 0.279 ± 0.014 | 0.110 ± 0.016 | 0.090 ± 0.016 | 0.079 ± 0.009 | 0.083 ± 0.009 | **0.061** ± 0.009 |
| Correlation $\rho$ | 25 | 0.633 ± 0.197 | 0.903 ± 0.114 | 0.905 ± 0.113 | 0.923 ± 0.045 | 0.866± 0.117 | **0.951** ± 0.112 |
| | 100 | 0.582 ± 0.112 | 0.827 ± 0.134 | 0.902 ± 0.059 | 0.899 ± 0.043 | 0.903± 0.049 | **0.983** ± 0.012 |
| | 225 | 0.580 ± 0.080 | 0.801 ± 0.078 | 0.863 ± 0.088 | 0.869 ± 0.037 | 0.873 ± 0.037 | **0.949** ± 0.022 |
| | 400 | 0.596 ± 0.054 | 0.779 ± 0.059 | 0.822 ± 0.047 | 0.852 ± 0.024 | 0.841 ± 0.028 | **0.912** ± 0.025 |
| $\log Z$ error | 25 | 2.512 ± 1.060 | 0.549 ± 0.373 | 0.557 ± 0.369 | **0.169** ± 0.142 | 0.762 ± 0.439 | 0.240 ± 0.140 |
| | 100 | 13.09 ± 2.156 | 1.650 ± 1.414 | 1.457 ± 1.365 | **0.524** ± 0.313 | 2.836 ± 2.158 | 1.899 ± 0.495 |
| | 225 | 29.93 ± 4.679 | 3.348 ± 1.954 | 3.423 ± 2.157 | **1.008** ± 0.653 | 3.249 ± 2.058 | 4.344 ± 0.813 |
| | 400 | 51.81 ± 4.706 | 5.738 ± 2.107 | 5.873± 2.211 | **1.750** ± 0.869 | 3.953 ± 2.558 | 7.598 ± 1.146 |

- $\ell_1$ error of beliefs v.s. true
- correlation $\rho$ between true and approximate marginals,
- $\log Z$ error, true v.s. free energy approximation.

# Some Numerical Comparisons

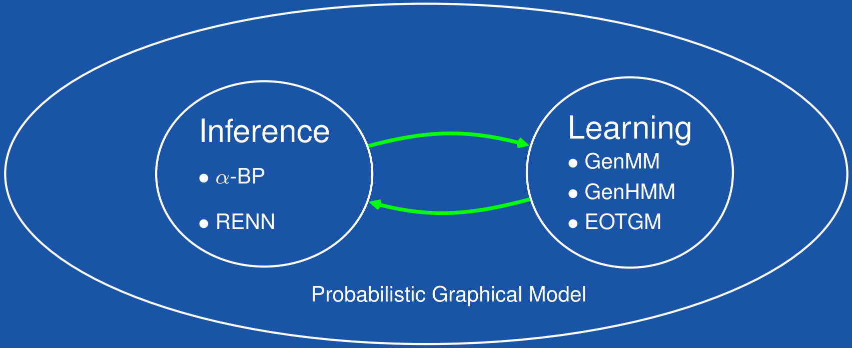
## Richer Comparisons

Inference on grid and complete graphs.

| | | Metric | | Mean Field | Loopy BP | Damped BP | GBP | Inference Net | RENN |
|---|---|---|---|---|---|---|---|---|---|
| High Temp -erature | Complete graph | $\ell_1$- | $\gamma = 1$ | 0.273 $\pm$ 0.086 | 0.239 $\pm$ 0.059 | 0.239 $\pm$ 0.059 | 0.260 $\pm$ 0.086 | 0.249 $\pm$ 0.067 | **0.181** $\pm$ 0.092 |
| | N=16 | error | $\gamma = 4$ | 0.197 $\pm$0.049 | 0.181 $\pm$ 0.035 | 0.180 $\pm$ 0.034 | 0.210 $\pm$ 0.070 | 0.174 $\pm$ 0.030 | **0.125** $\pm$ 0.050 |
| | $J_{ij} \sim \mathcal{N}(0, 1)$ | $\log Z$ | $\gamma = 1$ | 20.66 $\pm$ 5.451 | 178.7 $\pm$ 22.18 | 178.9 $\pm$ 21.88 | 153.3 $\pm$ 25.29 | 213.6 $\pm$ 12.75 | **14.41** $\pm$ 4.135 |
| | $h_i \sim \mathcal{N}(0, \gamma^2)$ | error | $\gamma = 4$ | **10.74** $\pm$ 7.385 | 565.7 $\pm$ 73.33 | 566.1 $\pm$ 73.13 | 106.0 $\pm$ 54.43 | 588.3 $\pm$ 62.58 | 14.72 $\pm$ 4.155 |
| Low Temp -erature | Grid graph | $\ell_1$ | 5 | 0.257 $\pm$ 0.065 | 0.115 $\pm$ 0.071 | 0.120 $\pm$ 0.073 | 0.250 $\pm$ 0.024 | 0.164 $\pm$ 0.036 | **0.100** $\pm$ 0.046 |
| | N=100 | error | 15 | 0.328 $\pm$ 0.068 | 0.228 $\pm$ 0.088 | 0.267 $\pm$ 0.147 | 0.303 $\pm$ 0.026 | 0.279 $\pm$ 0.024 | **0.207** $\pm$ 0.054 |
| | $J_{ij} \sim \mathcal{U}(-u, u)$ | $\log Z$ | 5 | 42.65 $\pm$ 17.86 | 7.346 $\pm$ 7.744 | **5.444** $\pm$ 4.811 | 8.369 $\pm$ 7.401 | 65.60 $\pm$ 8.786 | 11.34 $\pm$ 4.724 |
| | $h_i \sim \mathcal{U}(-1, 1)$ | error | 15 | 164.9 $\pm$ 56.07 | 58.40 $\pm$ 41.36 | 101.9 $\pm$ 54.31 | **23.10** $\pm$ 15.06 | 224.3 $\pm$ 25.52 | 78.85 $\pm$ 15.08 |

Average consumed time per inference instance (unit: second)

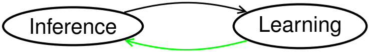
| | Mean Field | Loopy BP | Damped BP | GBP | Inference Net | RENN |
|---|---|---|---|---|---|---|
| Grid $\mathcal{G}$, $N = 400$, $J_{ij} \sim \mathcal{N}(0, 1)$, $h_i \sim \mathcal{N}(0, 0.1^2)$ | 9.897 | 425.0 | 328.3 | 286.3 | 74.41 | 101.0 |
| Complete $\mathcal{G}$, $N = 16$, $J_{ij} \sim \mathcal{N}(0, 1)$, $h_i \sim \mathcal{N}(0, 1)$ | 0.457 | 0.777 | 1.285 | 14.29 | 12.45 | 16.16 |
| Grid $\mathcal{G}$, $N = 100$, $J_{ij} \sim \mathcal{U}(-15, 15)$, $h_i \sim \mathcal{U}(-1, 1)$ | 2.314 | 253.3 | 229.3 | 53.72 | 103.4 | 79.38 |
| Complete $\mathcal{G}$, $N = 9$, $J_{ij} \sim \mathcal{U}(-15, 15)$, $h_i \sim \mathcal{U}(-1, 1)$ | 0.502 | 15.86 | 18.23 | 3.213 | 17.21 | 7.857 |


Inference
• $\alpha$-BP
• RENN
Learning
• GenMM
• GenHMM
• EOTGM
Probabilistic Graphical Model

# Two Direction Impact

Why impact in two direction?

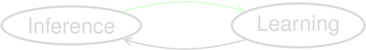
- Learning to Inference:



Stuff we have been taken as default, a given $p(\boldsymbol{x};\boldsymbol{\theta})$

- built by expert knowledge, or
- built by extracting information from evidence (empirical data).

- Inference to Learning:



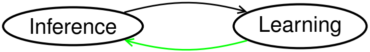
Model learning: an error trial process that compares inferred 'fact' and actual fact (evidence).

Model learning usually needs inference as a subroutine, which sometimes are replaced by sampling in particle based methods.

# TWO DIRECTION IMPACT

Why impact in two direction?

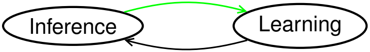
- Learning to Inference:



Stuff we have been taken as default, a given $p(\boldsymbol{x}; \boldsymbol{\theta})$

- built by expert knowledge, or
- built by extracting information from evidence (empirical data).

- Inference to Learning:



Model learning: an error trial process that compares inferred 'fact' and actual fact (evidence).
Model learning usually needs inference as a subroutine, which sometimes are replaced by sampling in particle based methods.

# Inference Routine in Learning

What is $\boldsymbol{\theta}$ in $p(\boldsymbol{x};\boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})}\prod_a \psi_a(\boldsymbol{x}_a;\boldsymbol{\theta}_a)$?
A direct view:

$$\max_{\boldsymbol{\theta}} \log p(\boldsymbol{x};\boldsymbol{\theta}) = \max_{\boldsymbol{\theta}} \sum_a \log \psi_a(\boldsymbol{x}_a;\boldsymbol{\theta}_a) \qquad \underbrace{-\log Z(\boldsymbol{\theta})}_{\text{Helmholtz free energy, can be est. by } F} \qquad ,$$

An alternative view:

$$\frac{\partial \log p(\boldsymbol{x};\boldsymbol{\theta})}{\partial \boldsymbol{\theta}_a} = \frac{\partial \log \varphi_a(\boldsymbol{x}_a;\boldsymbol{\theta}_a)}{\partial \boldsymbol{\theta}_a} - \underbrace{\mathbb{E}_{p(\boldsymbol{x}_a;\boldsymbol{\theta})}\left[\frac{\partial \log \varphi_a(\boldsymbol{x}_a;\boldsymbol{\theta}_a)}{\partial \boldsymbol{\theta}_a}\right]}_{\text{can be est. by beliefs}}.$$

Remark:

- This essentially requires estimation of Helmholtz free energy or marginal probabilities.
- Stationary points translate into moment matching.

# Inference Routine in Learning

What is $\boldsymbol{\theta}$ in $p(\boldsymbol{x};\boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})}\prod_a \psi_a(\boldsymbol{x}_a;\boldsymbol{\theta}_a)$?

A direct view:

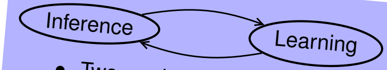
$$\max_{\boldsymbol{\theta}} \log p(\boldsymbol{x};\boldsymbol{\theta}) = \max_{\boldsymbol{\theta}} \sum_a \log \psi_a(\boldsymbol{x}_a;\boldsymbol{\theta}_a) \underbrace{-\log Z(\boldsymbol{\theta})}_{\text{Helmholtz free energy, can be est. by } F},$$

An alternative view:

$$\frac{\partial \log p(\boldsymbol{x};\boldsymbol{\theta})}{\partial \boldsymbol{\theta}_a} = \frac{\partial \log \varphi_a(\boldsymbol{x}_a;\boldsymbol{\theta}_a)}{\partial \boldsymbol{\theta}_a} - \underbrace{\mathbb{E}_{p(\boldsymbol{x}_a;\boldsymbol{\theta})}\left[\frac{\partial \log \varphi_a(\boldsymbol{x}_a;\boldsymbol{\theta}_a)}{\partial \boldsymbol{\theta}_a}\right]}_{\text{be est. by beliefs}}.$$

Remark:

- This essen al probabilities.
- Stationary



- Two modules are not necessarily coupled
- Each module may be replaced by another algorithm while the other one remains.

# Learning MRFs

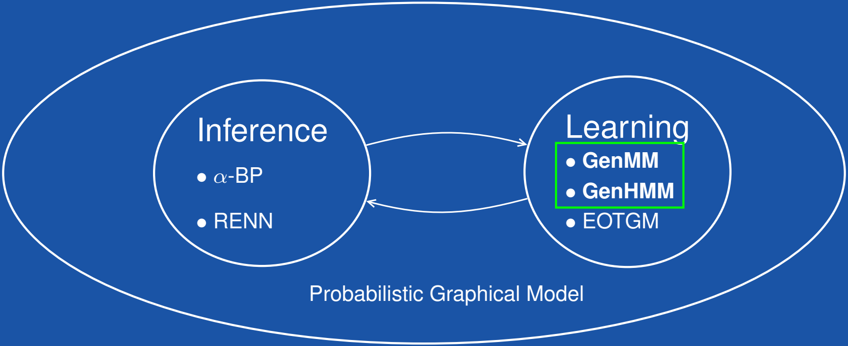
## What is $\boldsymbol{\theta}$ in $p(\boldsymbol{x};\boldsymbol{\theta})$?

Table of negative log-likelihood of learned MRFs

| N | True | Exact | Mean Field | Loopy BP | Damped BP | GBP | Inference Net | RENN |
|---|---|---|---|---|---|---|---|---|
| | | | | Grid Graph | | | | |
| 25 | 9.000 | 9.004 | 9.811 | 9.139 | 9.196 | 10.56 | 9.252 | **9.048** |
| 100 | 19.34 | 19.38 | 23.48 | 19.92 | 20.02 | 28.61 | 20. 29 | **19.76** |
| 225 | 63.90 | 63.97 | 69.01 | 66.44 | 66.25 | 92.62 | 68.15 | **64.79** |
| | | | | Complete Graph | | | | |
| 9 | 3.276 | 3.286 | 9.558 | 5.201 | 5.880 | 10.06 | 5.262 | **3.414** |
| 16 | 4.883 | 4.934 | 28.74 | 13.64 | 18.95 | 24.45 | 13.77 | **5.178** |

Average consumed time per epoch (unit: second) for two MRF learning cases.

| | Mean Field | Loopy BP | Damped BP | GBP | Inference Net | RENN |
|---|---|---|---|---|---|---|
| Grid $\mathcal{G}, N = 225$ | 40.09 | 335.1 | 525.1 | 12.37 | 19.49 | 16.03 |
| Complete $\mathcal{G}, N = 16$ | 2.499 | 12.40 | 5.431 | 1.387 | 0.882 | 2.262 |



Inference

- $\alpha$-BP
- RENN

Learning

- **GenMM**
- **GenHMM**
- EOTGM

Probabilistic Graphical Model

# Incomplete Observation

Partial observation: $\boldsymbol{x} = [\underbrace{\boldsymbol{x}_U}_{\textit{Unobserved}}, \underbrace{\boldsymbol{x}_O}_{\textit{Observed}}]$

$$l(\boldsymbol{x}_O;\boldsymbol{\theta}) = \log \sum_{\boldsymbol{x}_U} p(\boldsymbol{x}_U, \boldsymbol{x}_O;\boldsymbol{\theta}) = \log \underbrace{Z(\boldsymbol{x}_O;\boldsymbol{\theta})}_{\sum_{\boldsymbol{x}_U} \tilde{p}(\boldsymbol{x};\boldsymbol{\theta})} - \log Z(\boldsymbol{\theta}),$$

Connect Free Energy to Evidence Lower Bounder:

$$l(\boldsymbol{x}_O;\boldsymbol{\theta}) \geqslant -\underbrace{F_v(q(\boldsymbol{x}_U|\boldsymbol{x}_O))}_{\textit{VariationalFreeEnergy}} - \log Z(\boldsymbol{\theta})$$

$$= \mathbb{E}_{q(\boldsymbol{x}_U|\boldsymbol{x}_O)}\left[\log \frac{p(\boldsymbol{x}_U, \boldsymbol{x}_O;\boldsymbol{\theta})}{q(\boldsymbol{x}_U|\boldsymbol{x}_O)}\right]$$

$$= \underbrace{\mathbb{E}_{q(\boldsymbol{x}_U|\boldsymbol{x}_O)}[\log p(\boldsymbol{x}_U, \boldsymbol{x}_O;\boldsymbol{\theta})] + H(q(\boldsymbol{x}_U|\boldsymbol{x}_O))}_{\text{Evidence Lower Bound } F(q,\boldsymbol{\theta})}$$

Intuition of maximizing $F(q,\boldsymbol{\theta})$

- Maximizing (incomplete) likelihood
- Minimizing free energy

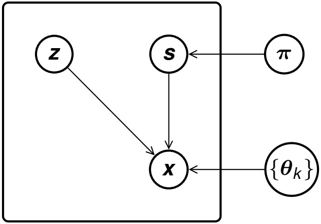
This gives the EM as a coordinate ascent method:

$$\text{E step}: \quad q^{(t+1)} = \operatorname*{argmax}_q F(q, \boldsymbol{\theta}^{(t)}),$$

$$\text{M step}: \quad \boldsymbol{\theta}^{(t+1)} = \operatorname*{argmax}_{\boldsymbol{\theta}} F(q^{(t+1)}, \boldsymbol{\theta}).$$

# GENERATOR MIXED MODEL

## EQUIPPING EM WITH NORMALIZING FLOWS



- Ideal case: The underline true $p^*(\boldsymbol{x})$ is in hypothesis space $\mathcal{H}$, i.e. $p^*(\boldsymbol{x}) \in \mathcal{H}$.
- Out of reach: Test $p^*(\boldsymbol{x}) \overset{?}{\in} \mathcal{H}$
- A general desire:

$$\mathcal{H} \text{ is large} \rightarrow \text{condidate } p(\boldsymbol{x};\boldsymbol{\theta}) \text{ is flexible}$$

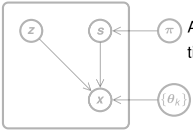
This brings up the finite **mixture** models.

$$p(\boldsymbol{x};\boldsymbol{\Theta}) = \sum_{k=1}^{K} \pi_k p_k(\boldsymbol{x}) = \sum_{k=1}^{K} \pi_k p(\underbrace{\boldsymbol{g}(\boldsymbol{z};\boldsymbol{\theta}_k)}_{\substack{\text{Variable change} \\ \text{via generator } \boldsymbol{g}}})$$

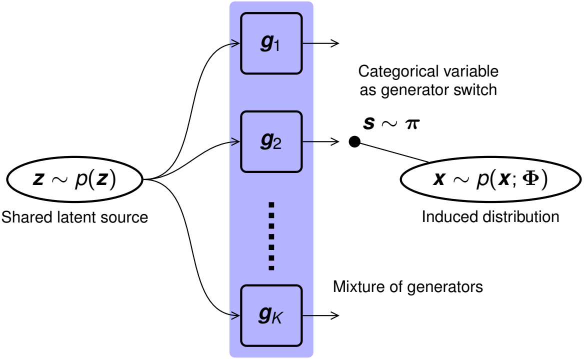
What to expect from GenMM:

- Flexible and expressive model, enlarging hyperspace $\mathcal{H}$
- Tractable likelihood
- Compatible with typical statistical models
- Compatible with NN tools/frameworks
- Scale to high-dimensional structured data
- Efficient in sampling (data generation)
- ...

# A HIGH-LEVEL VIEW OF GENMM: FINITE MIXTURE



Alternative illustration of GenMM

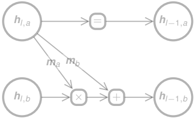


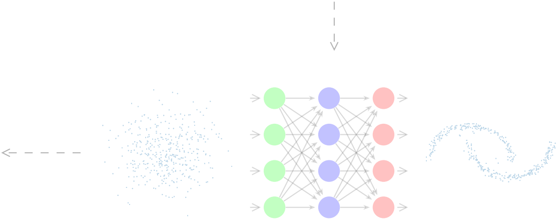
Categorical variable as generator switch

Shared latent source

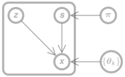
Induced distribution

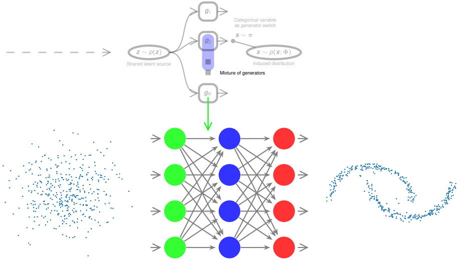
Mixture of generators



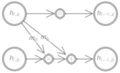


# A High-level View of GenMM: flow gears





When the $k$-th generator is selected, i.e., $s_k = 1$ and $s_{k'} = 0$ for $k' \neq k$, say $\tilde{\boldsymbol{x}} = \boldsymbol{x}|_{s_k=1}$. By following the change of variable rule

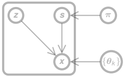


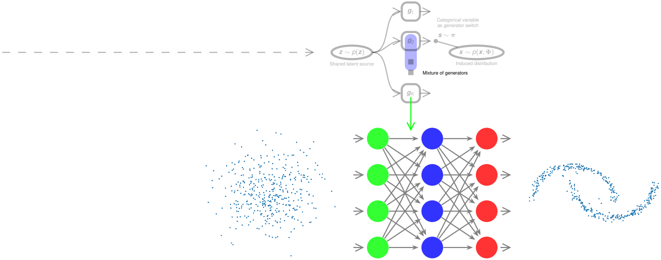
$$\underbrace{p(\tilde{\boldsymbol{x}})|_{\tilde{\boldsymbol{x}}=\tilde{\boldsymbol{g}}(\boldsymbol{z})}}_{\text{Induced distribution}} = \underbrace{p(\boldsymbol{z})}_{\substack{\text{Assumed known distribution}\\ \text{easy to sample}}} \cdot \underbrace{\left|\det\left(\frac{\partial \boldsymbol{z}}{\partial \tilde{\boldsymbol{x}}}\right)\right|}_{\substack{\text{Computational load}\\ \text{depends on the mapping}}} .$$

A toy example:

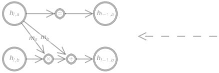
Gaussian linear transform: $Z \sim \mathrm{N}(0, 1) \xrightarrow{X=\sigma\cdot Z+\mu} X \sim \mathrm{N}(\mu, \sigma)$

# A High-level View of GenMM: flow gears





When the $k$-th generator is selected, i.e., $s_k = 1$ and $s_{k'} = 0$ for $k' \neq k$, say $\tilde{\boldsymbol{x}} = \boldsymbol{x}|_{s_k=1}$. By following the change of variable rule

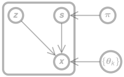


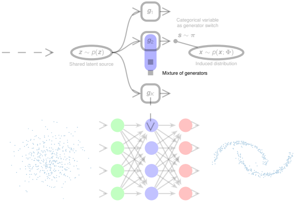
$$\underbrace{p(\tilde{\boldsymbol{x}})|_{\tilde{\boldsymbol{x}}=\tilde{\boldsymbol{g}}(\boldsymbol{z})}}_{\text{Induced distribution}} = \underbrace{p(\boldsymbol{z})}_{\substack{\text{Assumed known distribution}\\ \text{easy to sample}}} \cdot \underbrace{\left|\det\left(\frac{\partial \boldsymbol{z}}{\partial \tilde{\boldsymbol{x}}}\right)\right|}_{\substack{\text{Computational load}\\ \text{depends on the mapping}}} .$$

Powering it with a $L$-layer neural network implementation:

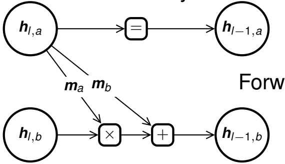
$$\begin{tikzcd}
\boldsymbol{z} = \boldsymbol{h}_0 \arrow[r, shift left, "\tilde{\boldsymbol{g}}_1"] & \boldsymbol{h}_1 \arrow[l, shift left, "\tilde{\boldsymbol{f}}_1"] \arrow[r, shift left, "\tilde{\boldsymbol{g}}_2"] & \cdots\cdots\cdots \arrow[l, shift left, "\tilde{\boldsymbol{f}}_2"] & \phantom{x} \arrow[r, shift left, "\tilde{\boldsymbol{g}}_L"] & \boldsymbol{x} = \boldsymbol{h}_L \arrow[l, shift left, "\tilde{\boldsymbol{f}}_L"]
\end{tikzcd}$$

# A High-level View of GenMM: Zoom into Layer





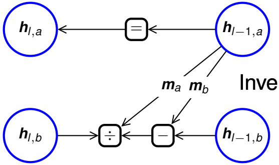
RealNVP structure by Dinh et al



Forward

Pick up one layer to zoom in

$$\boldsymbol{h}_{l-1} = \begin{bmatrix} \boldsymbol{h}_{l-1,a} \\ \boldsymbol{h}_{l-1,b} \end{bmatrix} = \begin{bmatrix} \boldsymbol{h}_{l,a} \\ \boldsymbol{m}_a(\boldsymbol{h}_{l,a}) \odot \boldsymbol{h}_{l,b} + \boldsymbol{m}_b(\boldsymbol{h}_{l,a}) \end{bmatrix}$$



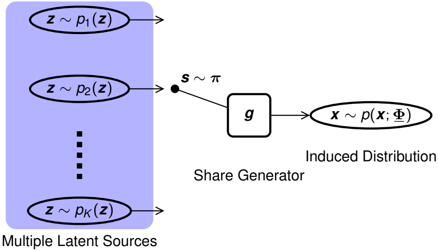
Inverse

$$\boldsymbol{h}_{l} = \begin{bmatrix} \boldsymbol{h}_{l,a} \\ \boldsymbol{h}_{l,b} \end{bmatrix} = \begin{bmatrix} \boldsymbol{h}_{l-1,a} \\ \left(\boldsymbol{h}_{l-1,b} - \boldsymbol{m}_b(\boldsymbol{h}_{l-1,a})\right) \oslash \boldsymbol{m}_a(\boldsymbol{h}_{l-1,a}) \end{bmatrix}$$

- $\odot$ denotes element-wise product, $\oslash$ denotes element-wise division
- Mapping $\boldsymbol{m}_a$, $\boldsymbol{m}_b$ can be as complex as possible and not necessary invertible
- Same computation complexity of forward and inverse mapping
- Triangular matix of Jacobian

Alternative arch. on market: Auto-regressive flow, Glow, ODE, etc.

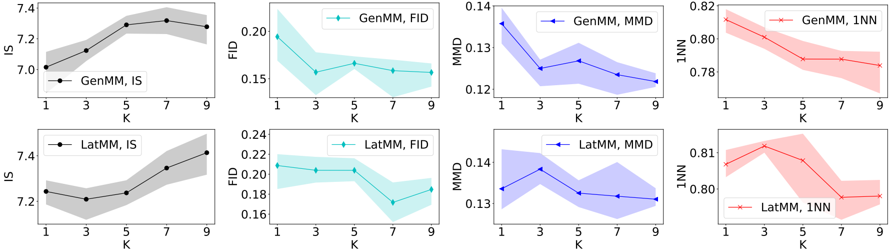
# Alternative Mixture and Remark



## Remark On GenMM/GenHMM

- Free dimension for flexibility: number of mixture + complexity of functional form of neural networks
- Compatible with statistic methods and neural network techniques (error back-propagation, optimizer)
- Embed batch-gradient descent into M-step
- Lack of closed-form update rule and generator changes at each gradient step. We tackle by maintaining old and new generators in EM steps

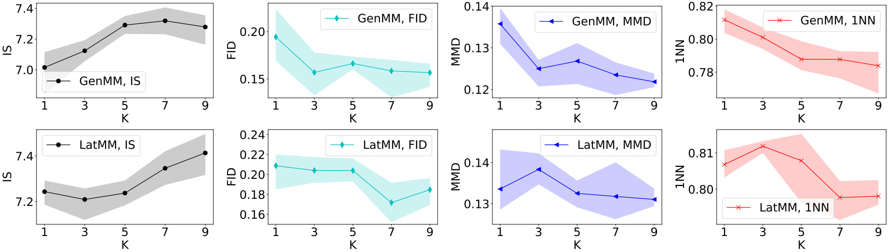
# Semantic Scores and Examples

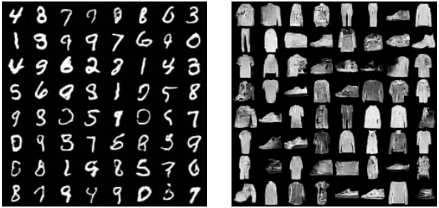


IS, FID, MMD and 1NN of GenMM and LatMM for MNIST dataset.

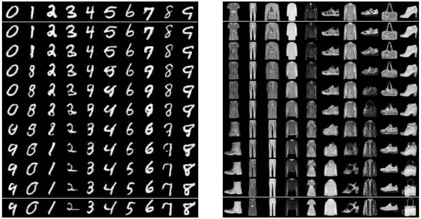
- IS: Inseption Score
- FID: Frechet Inception Distance
- MMD: Maximum Mean Discrepancy
- 1NN: 1-Nearest Neighbor

# SEMANTIC SCORES AND EXAMPLES





Generated samples by GenMM and LatMM.



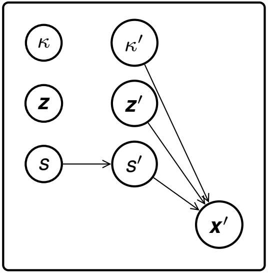
Interpolation in latent space

# APPLICATION TO CLASSIFICATION TASKS

Application to classification with maximum likelihood. Test Accuracy Table of GenMM for Classification Task

| Dataset | K=1 | K=2 | K=3 | K=4 | K=10 | K=20 | State Of Art |
|---|---|---|---|---|---|---|---|
| Letter | 0.9459 | 0.9513 | 0.9578 | 0.9581 | 0.9657 | **0.9674** | 0.9582 |
| Satimage | 0.8900 | 0.8975 | 0.9045 | 0.9085 | 0.9105 | **0.9160** | 0.9090 |
| Norb | 0.9184 | 0.9257 | 0.9406 | 0.9459 | 0.9538 | **0.9542** | 0.8920 |

# GenHMM: Bring the concept into HMM

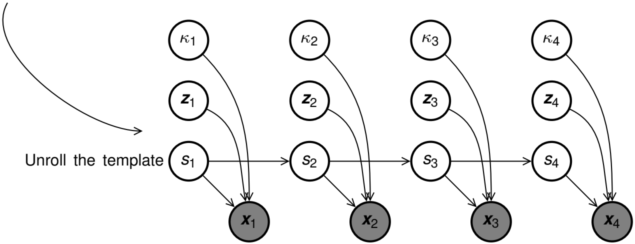


At time $t$, the probabilistic model of a state $s \in \mathcal{S}$ is then given by

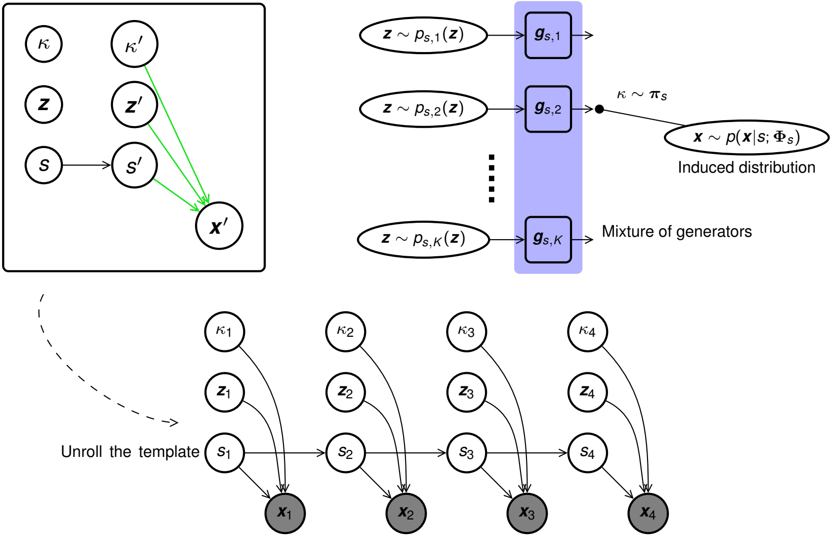
$$p(\boldsymbol{x}|s;\boldsymbol{\Phi}_s) = \sum_{\kappa=1}^{K} \pi_{s,\kappa} p(\boldsymbol{x}|s,\kappa;\boldsymbol{\theta}_{s,\kappa}),$$

where

- $\pi_{s,\kappa} = p(\kappa|s;\boldsymbol{H})$, naturally $\sum_{\kappa=1}^{K} \pi_{s,\kappa} = 1$
- $p_k(\boldsymbol{x})$ is induced by the $k$th generator $\boldsymbol{g}_k(\boldsymbol{z}) = \boldsymbol{g}(\boldsymbol{z};\boldsymbol{\theta}_k)$



# GenHMM: Bring the concept into HMM

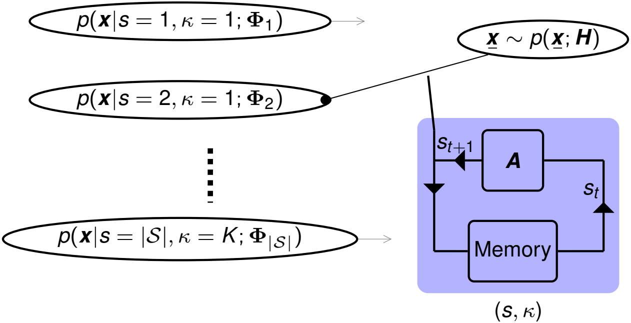


Induced distribution

Mixture of generators

Unroll the template

# Alternative View of GenHMM


$p(\boldsymbol{x}|s=1,\kappa=1;\boldsymbol{\Phi}_1)$
$\underline{\boldsymbol{x}} \sim p(\underline{\boldsymbol{x}};\boldsymbol{H})$
$p(\boldsymbol{x}|s=2,\kappa=1;\boldsymbol{\Phi}_2)$
$s_{t+1}$
$\boldsymbol{A}$
$s_t$
$p(\boldsymbol{x}|s=|\mathcal{S}|,\kappa=K;\boldsymbol{\Phi}_{|\mathcal{S}|})$
Memory
$(s,\kappa)$

We define the hypothesis set of HMM as $\mathcal{H} := \{\boldsymbol{H}|\boldsymbol{H} = \{\mathcal{S}, \boldsymbol{q}, \boldsymbol{A}, p(\boldsymbol{x}|s;\boldsymbol{\Phi}_s)\}\}$, where

- $\mathcal{S}$ is the set of hidden states of $\boldsymbol{H}$.
- $\boldsymbol{q} = \left[q_1, q_2, \cdots, q_{|\mathcal{S}|}\right]^\intercal$ is the initial state distribution of $\boldsymbol{H}$ with $|\mathcal{S}|$ as cardinality of $\mathcal{S}$. For $i \in \mathcal{S}$, $q_i = p(s_1 = i; \boldsymbol{H})$. We use $s_t$ to denote the state $s$ at time $t$.
- $\boldsymbol{A}$ matrix of size $|\mathcal{S}| \times |\mathcal{S}|$ is the transition matrix of states in $\boldsymbol{H}$. That is, $\forall i, j \in \mathcal{S}$, $\boldsymbol{A}_{i,j} = p(s_{t+1} = j|s_t = i; \boldsymbol{H})$.
- For a given hidden state $s$, the density function of the observable signal is $p(\boldsymbol{x}|s;\boldsymbol{\Phi}_s)$, where $\boldsymbol{\Phi}_s$ is the parameter set that defines this probabilistic model. Denote $\boldsymbol{\Phi} = \{\boldsymbol{\Phi}_s|s \in \mathcal{S}\}$.

# Learning Intuition

With empirical distribution $\hat{p}(\underline{\boldsymbol{x}}) = \frac{1}{N}\sum_n \delta_{\underline{\boldsymbol{x}}^{(n)}}(\underline{\boldsymbol{x}})$, learning of GenHMM boils down to

$$\min_{\boldsymbol{H}\in\mathcal{H}} \mathrm{KL}(\hat{p}(\underline{\boldsymbol{x}})\|p(\underline{\boldsymbol{x}};\boldsymbol{H}))$$

Then problem becomes maximum likelihood estimation

$$\hat{\boldsymbol{H}} = \arg\max_{\boldsymbol{H}\in\mathcal{H}} \log \prod_i p(\underline{\boldsymbol{x}}^{(i)};\boldsymbol{H}),$$

Use EM to solve:

E-step: the expected likelihood function

$$\mathcal{Q}(\boldsymbol{H};\boldsymbol{H}^{\mathrm{old}}) = \mathbb{E}_{\hat{p}(\underline{\boldsymbol{x}}),p(\underline{\boldsymbol{s}},\underline{\boldsymbol{\kappa}}|\underline{\boldsymbol{x}};\boldsymbol{H}^{\mathrm{old}})}\left[\log\, p(\underline{\boldsymbol{x}},\underline{\boldsymbol{s}},\underline{\boldsymbol{\kappa}};\boldsymbol{H})\right],$$

where $\mathbb{E}_{\hat{p}(\underline{\boldsymbol{x}}),p(\underline{\boldsymbol{s}},\underline{\boldsymbol{\kappa}}|\underline{\boldsymbol{x}};\boldsymbol{H}^{\mathrm{old}})}[\cdot]$ denotes the expectation operator by distribution $\hat{p}(\underline{\boldsymbol{x}})$ and $p(\underline{\boldsymbol{s}},\underline{\boldsymbol{\kappa}}|\underline{\boldsymbol{x}};\boldsymbol{H}^{\mathrm{old}})$.

M-step: the maximization step

$$\max_{\boldsymbol{H}} \mathcal{Q}(\boldsymbol{H};\boldsymbol{H}^{\mathrm{old}}).$$

The problem (40) can be reformulated as

$$\max_{\boldsymbol{H}} \mathcal{Q}(\boldsymbol{H};\boldsymbol{H}^{\mathrm{old}}) = \underbrace{\max_{\boldsymbol{q}} \mathcal{Q}(\boldsymbol{q};\boldsymbol{H}^{\mathrm{old}})}_{\text{Initial State}} + \underbrace{\max_{\boldsymbol{A}} \mathcal{Q}(\boldsymbol{A};\boldsymbol{H}^{\mathrm{old}})}_{\text{Transition}} + \underbrace{\max_{\boldsymbol{\Phi}} \mathcal{Q}(\boldsymbol{\Phi};\boldsymbol{H}^{\mathrm{old}})}_{\text{Generators}}, \tag{1}$$

Maximizing a lower bound of $\log\prod_i p(\underline{\boldsymbol{x}}^{(i)};\boldsymbol{H})$.

# LEARNING INTUITION

With empirical distribution $\hat{p}(\underline{\boldsymbol{x}}) = \frac{1}{N}\sum_n \delta_{\underline{\boldsymbol{x}}^{(n)}}(\underline{\boldsymbol{x}})$, learning of GenHMM boils down to

$$\min_{\boldsymbol{H}\in\mathcal{H}} \mathrm{KL}(\hat{p}(\underline{\boldsymbol{x}})\|p(\underline{\boldsymbol{x}};\boldsymbol{H}))$$

Then problem becomes maximum likelihood estimation

$$\hat{\boldsymbol{H}} = \arg\max \log \prod_i p(\underline{\boldsymbol{x}}^{(i)};\boldsymbol{H}),$$

- $F = \mathcal{Q}$ + Entropy
- E-step require inference (message-passing)
- No optimality in M-step (NN generators).
- Still, guaranteed non-decreasing lklh. (c.f. Proposition 7.1)

Use EM to solve:

E-step: the

where $\mathbb{E}_{\hat{p}(\underline{\boldsymbol{x}}),p(\underline{\boldsymbol{s}},\underline{\boldsymbol{\kappa}}|\underline{\boldsymbol{x}};\ldots)}$ ution $\hat{p}(\underline{\boldsymbol{x}})$ and $p(\underline{\boldsymbol{s}},\underline{\boldsymbol{\kappa}}|\underline{\boldsymbol{x}};\boldsymbol{H}^{\mathrm{old}})$.

M-step: the maximization step

$$\max_{\boldsymbol{H}} \mathcal{Q}(\boldsymbol{H};\boldsymbol{H}^{\mathrm{old}}).$$

The problem (40) can be reformulated as

$$\max_{\boldsymbol{H}} \mathcal{Q}(\boldsymbol{H};\boldsymbol{H}^{\mathrm{old}}) = \underbrace{\max_{\boldsymbol{q}} \mathcal{Q}(\boldsymbol{q};\boldsymbol{H}^{\mathrm{old}})}_{\text{Initial State}} + \underbrace{\max_{\boldsymbol{A}} \mathcal{Q}(\boldsymbol{A};\boldsymbol{H}^{\mathrm{old}})}_{\text{Transition}} + \underbrace{\max_{\boldsymbol{\Phi}} \mathcal{Q}(\boldsymbol{\Phi};\boldsymbol{H}^{\mathrm{old}})}_{\text{Generators}}, \qquad (1)$$

Maximizing a lower bound of $\log\prod_i p(\underline{\boldsymbol{x}}^{(i)};\boldsymbol{H})$.

# APPLICATION TO SPEECH RECOGNITION

Configuration of generators of GenHMM in Experiments on TIMIT

| | |
|---|---|
| Latent distribution $p_{s,\kappa}(\boldsymbol{z})$<br>$s \in \mathcal{S}, \kappa = 1, 2, \cdots, K$ | Standard Gaussian |
| Number of flow blocks | 4 |
| Non-linear mapping $\boldsymbol{m}_a$, $\boldsymbol{m}_b$ | Multiple layer perception<br>3 layers and with hidden dimension 24 |

Phoneme classification / recognition

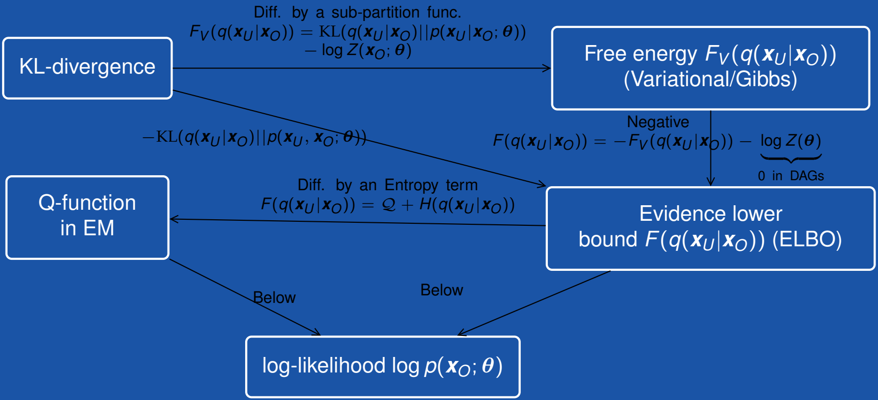
| Model | Criterion | K=1 | K=3 | K=5 |
|---|---|---|---|---|
| GMM-HMM | Accuracy | 62.3 | 68.0 | 68.7 |
| | Precision | 67.9 | 72.6 | 73.0 |
| | F1 | 63.7 | 69.1 | 69.7 |
| GenHMM | Accuracy | 76.7 | **77.7** | 77.7 |
| | Precision | 76.9 | **78.1** | 78.0 |
| | F1 | 76.1 | **77.1** | 77.0 |

Robustness to perturbation of noise.

| Model | Criterion | White Noise SNR | | | |
|---|---|---|---|---|---|
| | | 15dB | 20dB | 25dB | 30dB |
| GMM-HMM | Accuracy | 36.6 | 44.2 | 50.8 | 57.1 |
| | Precision | 59.2 | 64.2 | 68.4 | 70.6 |
| | F1 | 39.9 | 47.7 | 53.9 | 59.9 |
| GenHMM | Accuracy | 52.4 | 62.0 | 69.7 | **74.3** |
| | Precision | 60.0 | 65.9 | 71.7 | **74.8** |
| | F1 | 52.5 | 62.0 | 69.3 | **73.5** |

For our conducted experiments of applying GenHMM to sepsis detection for infants, see Section 7.5.

# What Have We been Talking About?



KL-divergence

Diff. by a sub-partition func.

$F_V(q(\boldsymbol{x}_U|\boldsymbol{x}_O)) = \mathrm{KL}(q(\boldsymbol{x}_U|\boldsymbol{x}_O)||p(\boldsymbol{x}_U|\boldsymbol{x}_O;\boldsymbol{\theta})) - \log Z(\boldsymbol{x}_O;\boldsymbol{\theta})$

Free energy $F_V(q(\boldsymbol{x}_U|\boldsymbol{x}_O))$ (Variational/Gibbs)

$-\mathrm{KL}(q(\boldsymbol{x}_U|\boldsymbol{x}_O)||p(\boldsymbol{x}_U,\boldsymbol{x}_O;\boldsymbol{\theta}))$

Negative

$F(q(\boldsymbol{x}_U|\boldsymbol{x}_O)) = -F_V(q(\boldsymbol{x}_U|\boldsymbol{x}_O)) - \underbrace{\log Z(\boldsymbol{\theta})}_{0 \text{ in DAGs}}$

Q-function in EM

Diff. by an Entropy term

$F(q(\boldsymbol{x}_U|\boldsymbol{x}_O)) = \mathcal{Q} + H(q(\boldsymbol{x}_U|\boldsymbol{x}_O))$

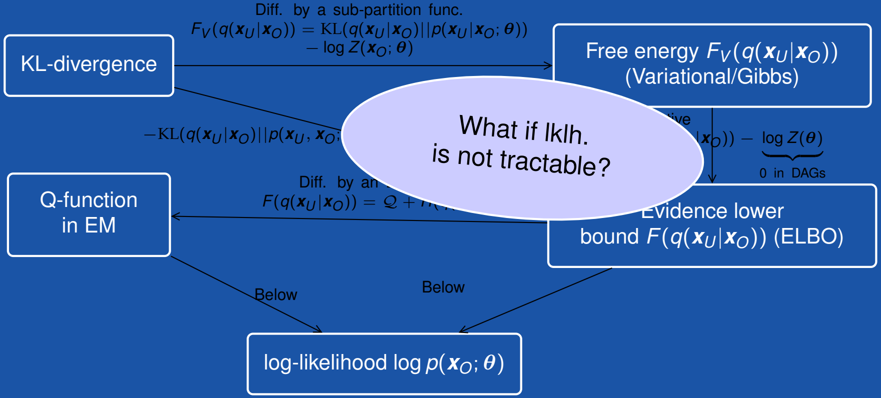
Evidence lower bound $F(q(\boldsymbol{x}_U|\boldsymbol{x}_O))$ (ELBO)

Below

Below

log-likelihood $\log p(\boldsymbol{x}_O;\boldsymbol{\theta})$

# What Have We been Talking About?



KL-divergence

Diff. by a sub-partition func.

$F_V(q(\boldsymbol{x}_U|\boldsymbol{x}_O)) = \mathrm{KL}(q(\boldsymbol{x}_U|\boldsymbol{x}_O)||p(\boldsymbol{x}_U|\boldsymbol{x}_O;\boldsymbol{\theta})) - \log Z(\boldsymbol{x}_O;\boldsymbol{\theta})$

Free energy $F_V(q(\boldsymbol{x}_U|\boldsymbol{x}_O))$ (Variational/Gibbs)

$-\mathrm{KL}(q(\boldsymbol{x}_U|\boldsymbol{x}_O)||p(\boldsymbol{x}_U, \boldsymbol{x}_O;$

tive

$|\boldsymbol{x}_O)) - \underbrace{\log Z(\boldsymbol{\theta})}_{0 \text{ in DAGs}}$

What if lklh.
is not tractable?

Diff. by an

$F(q(\boldsymbol{x}_U|\boldsymbol{x}_O)) = \mathcal{Q} + H($

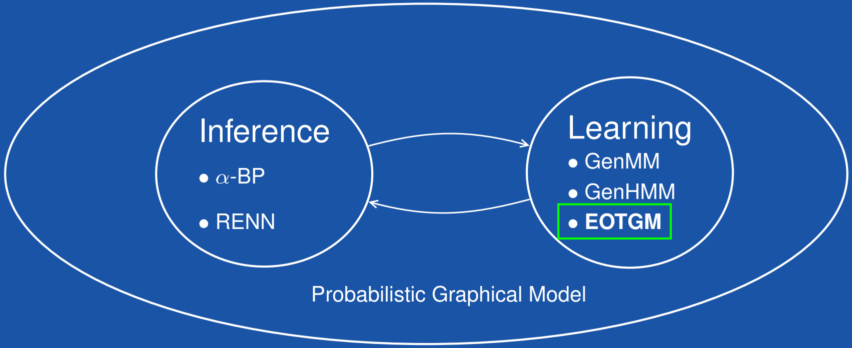
Q-function in EM

Evidence lower bound $F(q(\boldsymbol{x}_U|\boldsymbol{x}_O))$ (ELBO)

Below

Below

log-likelihood $\log p(\boldsymbol{x}_O;\boldsymbol{\theta})$


Inference
• $\alpha$-BP
• RENN
Learning
• GenMM
• GenHMM
• **EOTGM**
Probabilistic Graphical Model

# SUMMARY

- Brief on probabilistic graphic models
- Overview of inference methods
- A focus on the message-passing
- Transition to inference methods with NN

Thank you for your attention.
Q&A.