Approximate Inference and Learning: From Message-Passing to Neural Network based Methods

Dong Liu

Information Science and Engineering

KTH - Royal Institute of Technology

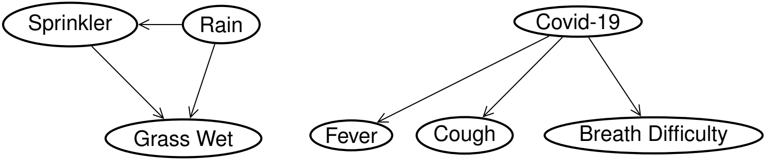
CONTENT

- Background: Probabilistic graphical models (PGM)
- Common usage of PGMs
- High-level view of inference
- Play inference with neural networks
- Summary

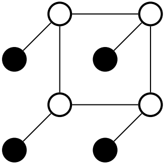
A few words on Probabilistic Graphical Models

PROBABILISTIC GRAPHICAL MODELS

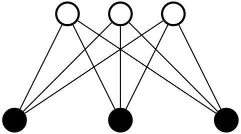
Directed: Bayesian Networks



Undirected: Markov Random Field



Computer vision



Communication system

Two key aspects to encode in a graphical model:

- attributes of our interests in a system $\rightarrow$ variable nodes
- relationship of these factors (dependencies or indepedencies) $\rightarrow$ structures of a graph

Markov Random Field

Markov Random Field (MRF) and Fundamental Inference Problems

Let us walk through via MRF

- An MRF can be represented by a graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ with each node $i \in \mathcal{V}$ is associated with a random variable X_i
- The probability distribution (Gibbs distribution) is

$$p(\boldsymbol{x}; \boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})} \prod_a \psi_a(\boldsymbol{x}_a; \boldsymbol{\theta}_a),$$

where $\boldsymbol{x} = \{x_1, x_2, \cdots, x_N\}$, a indexes potential functions $\mathcal{I} = \{\psi_A, \psi_B, \cdots, \psi_M\}$ and $\boldsymbol{\theta}$ is set of potential function parameters. $Z(\boldsymbol{\theta}) = \sum_{\boldsymbol{x}} \prod_a \psi_a(\boldsymbol{x}_a; \boldsymbol{\theta}_a)$.

What do we do with graphical models?

USAGE OF GRAPHICAL MODELS

In general:

- Representation
  - In place of real systems
  - Abstraction of complex problems or systems (with subjective bias)
- Answer queries
  Evidence (observation) $\rightarrow$?? $\rightarrow$ Answers

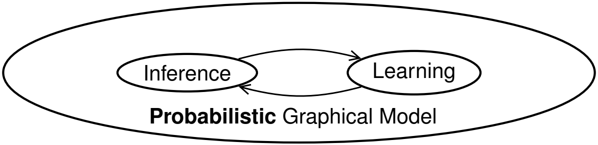
Two components interacting with each other:

USAGE OF GRAPHICAL MODELS

In general:

- Representation
    - In place of real systems
    - Abstraction of complex problems or systems (with subjective bias)
- Answer queries
  Evidence (observation) $\rightarrow$?? $\rightarrow$ Answers

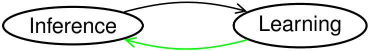
Two components interacting with each other:



USAGE OF GRAPHICAL MODELS

Why impact in two direction?

- Learning to Inference:



  A graphical model
  - 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.

USAGE OF GRAPHICAL MODELS

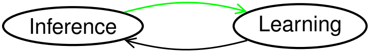
Why impact in two direction?

- Learning to Inference:

A graphical model

  - 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.

Common Inference Problems

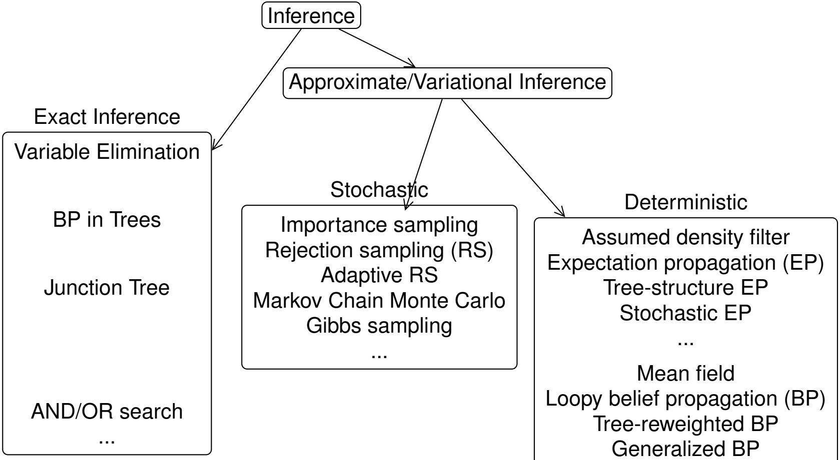
The common inference problems in a MRF $\mathcal{G}(\mathcal{V}, \mathcal{E})$:

- 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 most likely configuration $\operatorname{argmax}_{\boldsymbol{x}} p(\boldsymbol{x})$

Common Inference Problems

The common inference problems in a MRF $\mathcal{G}(\mathcal{V}, \mathcal{E})$:

- 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 most likely configuration $\arg\max_{\boldsymbol{x}} p(\boldsymbol{x})$



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:

$$\max_{\boldsymbol{\theta}} \log p(\boldsymbol{x};\boldsymbol{\theta}) = \max_{\boldsymbol{\theta}} \sum_a \log \psi_a(\boldsymbol{x}_a;\boldsymbol{\theta}_a) \quad \underbrace{-\log Z(\boldsymbol{\theta})}_{Helmholtz\ free\ energy} \quad ,$$

An alternative view:

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

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:

$$\max_{\boldsymbol{\theta}} \log p(\boldsymbol{x};\boldsymbol{\theta}) = \max_{\boldsymbol{\theta}} \sum_a \log \psi_a(\boldsymbol{x}_a;\boldsymbol{\theta}_a) \quad \underbrace{-\log Z(\boldsymbol{\theta})}_{Helmholtz\ free\ energy} \quad ,$$

An alternative view:

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

Remark:

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

Play with **Gibbs (variational) free energy**

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

with trial $b(\boldsymbol{x})$.

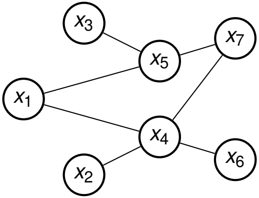
What is the state of x?

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.

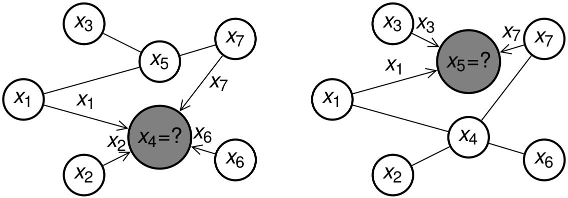


what is the state of x_4

WHAT IS THE STATE OF x?

GIBBS SAMPLING: LET US GUESS BY SAMPLING

We can approximately sample iteratively: $x_i \sim p(x_i|\boldsymbol{x}_{-i}) \sim p(x_i, \boldsymbol{x}_{-i})$



This coordinate-wise sampling algorithm is called Gibbs sampling, which answers queries by collected samples $\{\boldsymbol{x}^n\}_1^N$.

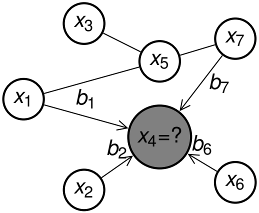
Gibbs sampling is named after the physicist Josiah Willard Gibbs, which was described by brothers Stuart and Donald Geman in 1984, some eight decades after the death of Gibbs.

$$x_i \sim \exp\{\sum_{a \in \mathrm{ne}_i} \log \varphi_a(x_i, \boldsymbol{x}_{a-i}; \boldsymbol{\theta}_a)\}$$

where ne_i gives the neighboring potential factors of node i.

WHAT IS THE STATE OF x?

Naive Mean Field: **message in form of sample values** $\rightarrow$ **message in form of belief**



Corresponding to minimization of **variational free energy** $F_v(b)$ **with trial** b **in fully-factorized form for univariant** $\{b_i\}$.

Iterative sampling $\rightarrow$ iterative belief update via

$$\log b_i(x_i) \propto \sum_{a \in \mathrm{ne}_i} \sum_{\boldsymbol{x}_a \setminus x_i} \prod_{j \in a \setminus i} b_j(x_j) \log \varphi_a(\boldsymbol{x}_a; \boldsymbol{\theta}_a).$$

WHAT IS THE STATE OF x?

BELIEF PROPAGATION (BP): LET US GUESS BY PROPAGATING BELIEF

Proposed by Pearl (1982) for Bayesian networks (tree-structured graphs), which then widely used for general graphs (loopy BP).
Yedidia, et al, connected the loopy BP with stationary points of **Bethe free energy**

$$F_{Bethe}(b) = \sum_{a\in\mathcal{F}} \sum_{\boldsymbol{x}_a} b_a(\boldsymbol{x}_a) \log \frac{b_a(\boldsymbol{x}_a)}{\varphi_a(\boldsymbol{x}_a)} - \sum_{i=1}^{N} (|\mathrm{ne}_i| - 1) \sum_{x_i} b_i(x_i) \log b_i(x_i),$$

Corresponding to minimization of approximated **variational free energy $F_v(b)$ with trial b includes $\{b_i\}$ and $\{b_a\}$**.

msg : factor to variable $m_{a\to i}(x_i) \propto \sum_{\boldsymbol{x}_a\setminus x_i} \varphi_a(\boldsymbol{x}_a) \prod_{j\in a\setminus i} m_{j\to a}(x_j),$

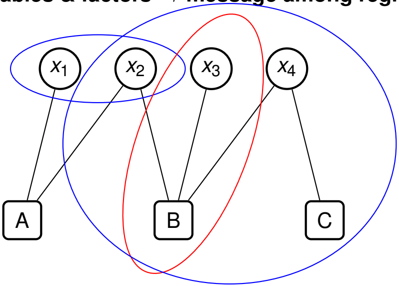
msg : variable to factor $m_{j\to a}(x_j) \propto \prod_{a'\in \mathrm{ne}_j\setminus a} m_{a'\to j}(x_j)$

See, D. Liu, M. T. Vu, Z. Li, and Lars K. Rasmussen. α belief propagation for approximate inference. 2020
D. Liu, N. N. Moghadam, L. K. Rasmussen, etc. α belief propagation as fully factorized approximation. In GlobalSIP, 2019. for alternative view to loopy BP.

WHAT IS THE STATE OF x?

YEDIDIA, FREEMAN, WEISS: A STEP TO GENERALIZATION

Message among variables & factors → message among regions



Generalized belief propagation (GBP) generalizes loopy BP

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

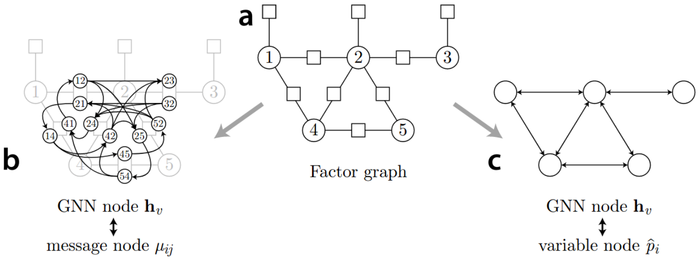
Corresponding to minimization of approximated **variational free energy** $F_v(b)$ **with trial** b **including** $\{b_R\}$.

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.

Attempts with neural networks: an imitation game of message passing, or trials under free energy?

LEARN THE MESSAGE UPDATE RULE BY NN

An end-to-end learning process: Factor graph → converted graph representation → GNN → Output



- sum-product update rule (in BP) → NN, to learn
- pseudo probability (belief aggregation) → NN, to learn
- end-to-end learning that requires true marginal probability, which BP, GPB and mean field do no require

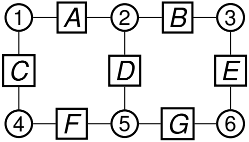
For related methods, see:
Heess et al, Learning to Pass Expectation Propagation Messages
Yoon, et al, 2019, Inference in Probabilistic Graphical Models by Graph Neural Networks
Gilmer, et al, 2017, Neural message passing for quantum chemistry.
Battaglia, et al, 2018, Relational inductive biases, deep learning, and graph networks

RENN

REGION REVISITED

- If you cannot collect true targets ($p(x_i)$)
- If you are unwilling to be restricted to pre-defined inference

Factor graph representation of MRF (2-by-3 grid) with factor nodes.
MRF $\rightarrow$ region graph:



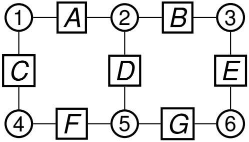
An alternative region graph of the same MRF:

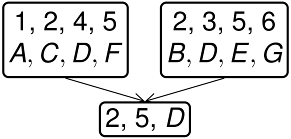
RENN

REGION REVISITED

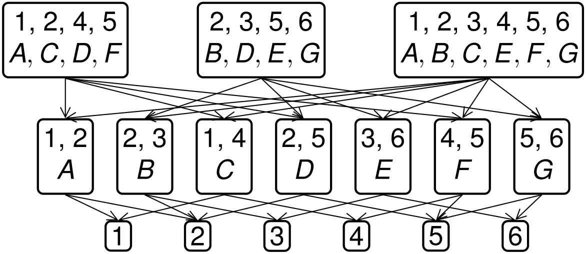
- If you cannot collect true targets ($p(x_i)$)
- If you are unwilling to be restricted to pre-defined inference

Factor graph representation of MRF (2-by-3 grid) with factor nodes.
MRF $\rightarrow$ region graph:





An alternative region graph of the same MRF:



RENN

The region-based free energy of a region graph is

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

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

Denote

by

Amortizing root beliefs:

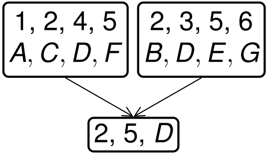
RENN

The region-based free energy of a region graph is

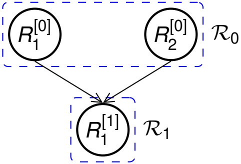
$$F_R(\mathcal{B};\boldsymbol{\theta}) = \sum_{R\in\mathcal{R}} \underbrace{c_R}_{counting\ number} \sum_{\boldsymbol{x}_R} b_R(\boldsymbol{x}_R)(\underbrace{E_R(\boldsymbol{x}_R;\boldsymbol{\theta}_R)}_{region\ average\ energy} + \ln b_R(\boldsymbol{x}_R)),$$

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

Denote



by



Amortizing root beliefs:

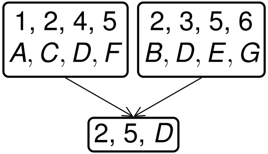
RENN

The region-based free energy of a region graph is

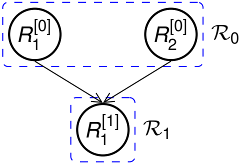
$$F_R(\mathcal{B};\boldsymbol{\theta}) = \sum_{R\in\mathcal{R}} \underbrace{c_R}_{counting\ number} \sum_{\boldsymbol{x}_R} b_R(\boldsymbol{x}_R)\Big(\underbrace{E_R(\boldsymbol{x}_R;\boldsymbol{\theta}_R)}_{region\ average\ energy} + \ln b_R(\boldsymbol{x}_R)\Big),$$

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

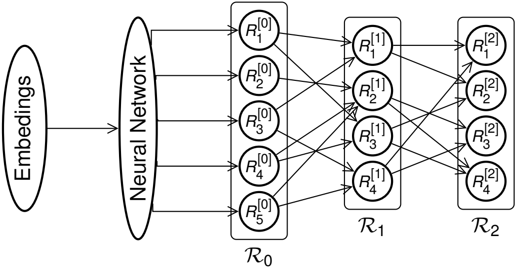
Denote



by



Amortizing root beliefs:



RENN

Objective of RENN[1]:

$$\min \textbf{region-based free energy}(F_R) + \underbrace{\textbf{panelty on belief consistency}}_{\textit{along region graph struture}}$$

Inference only

Learning alternatives of MRFs

learn with customized optm.

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

learn with auto-grads

$$\max_{\boldsymbol{\theta}} \log p(\boldsymbol{x};\boldsymbol{\theta}) = \max_{\boldsymbol{\theta}} \sum_a \log \psi_a(\boldsymbol{x}_a;\boldsymbol{\theta}_a) - \underbrace{\log Z(\boldsymbol{\theta})}_{ets.\ free\ energy},$$

by $-\log Z(\boldsymbol{\theta}) \simeq F_R$.

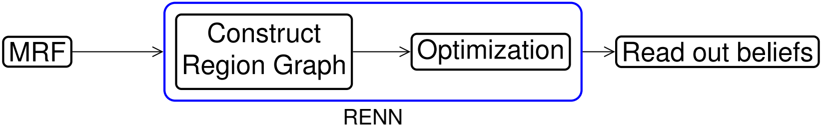
[1] More detail on RENN? Refer to, Dong Liu, Ragnar Thobaben, and Lars K. Rasmussen. Region-based energy neural network for approximate inference. arxiv, 2020

RENN

Objective of RENN[1]:

$$\min \textbf{region-based free energy}(F_R) + \underbrace{\textbf{panelty on belief consistency}}_{\textit{along region graph struture}}$$

Inference only



Learning alternatives of MRFs

learn with customized optm.

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

learn with auto-grads

$$\max_{\boldsymbol{\theta}} \log p(\boldsymbol{x};\boldsymbol{\theta}) = \max_{\boldsymbol{\theta}} \sum_a \log \psi_a(\boldsymbol{x}_a;\boldsymbol{\theta}_a) - \underbrace{\log Z(\boldsymbol{\theta})}_{\textit{ets. free energy}},$$

by $-\log Z(\boldsymbol{\theta}) \simeq F_R$.

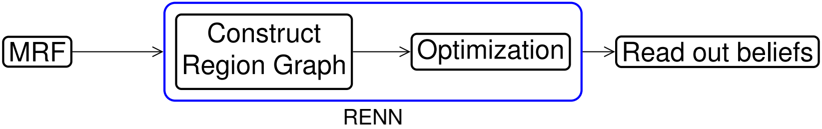
[1] More detail on RENN? Refer to, Dong Liu, Ragnar Thobaben, and Lars K. Rasmussen. Region-based energy neural network for approximate inference. arxiv, 2020

RENN

Objective of RENN[1]:

$$\min \textbf{region-based free energy}(F_R) + \underbrace{\textbf{panelty on belief consistency}}_{along\ region\ graph\ struture}$$

Inference only



Learning alternatives of MRFs

learn with customized optm.

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

learn with auto-grads

$$\max_{\boldsymbol{\theta}} \log p(\boldsymbol{x};\boldsymbol{\theta}) = \max_{\boldsymbol{\theta}} \sum_a \log \psi_a(\boldsymbol{x}_a;\boldsymbol{\theta}_a) - \underbrace{\log Z(\boldsymbol{\theta})}_{ets.\ free\ energy},$$

by $-\log Z(\boldsymbol{\theta}) \simeq F_R$.

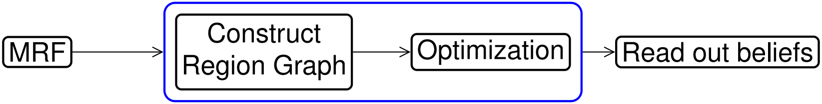
[1] More detail on RENN? Refer to, Dong Liu, Ragnar Thobaben, and Lars K. Rasmussen. Region-based energy neural network for approximate inference. arxiv, 2020

RENN

Objective of RENN[1]:

$$\min \textbf{region-based free energy}(F_R) + \underbrace{\textbf{panelty on belief consistency}}_{\textit{along region graph struture}}$$

Inference only



RENN

Learning alternatives of MRFs

learn with customized optm.

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

learn with auto-grads

$$\max_{\boldsymbol{\theta}} \log p(\boldsymbol{x};\boldsymbol{\theta}) = \max_{\boldsymbol{\theta}} \sum_a \log \psi_a(\boldsymbol{x}_a;\boldsymbol{\theta}_a) - \underbrace{\log Z(\boldsymbol{\theta})}_{\textit{ets. free energy}},$$

by $-\log Z(\boldsymbol{\theta}) \simeq F_R$.

[1] More detail on RENN? Refer to, Dong Liu, Ragnar Thobaben, and Lars K. Rasmussen. Region-based energy neural network for approximate inference. arxiv, 2020

INFERENCE RESULTS

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
ℓ_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 ρ	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

- ℓ_1 error of beliefs v.s. true
- correlation ρ between true and approximate marginals,
- $\log Z$ error, true v.s. free energy approximation.

Inference Net: Wiseman, Kim, Amortized Bethe Free Energy Minimization for Learning MRFs, 2019.

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**

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.