Collapsed Variational Inference for Sum-Product Networks

Han Zhao\textsuperscript{1}, Tameem Adel\textsuperscript{2}, Geoff Gordon\textsuperscript{1}, Brandon Amos\textsuperscript{1}
Presented by: Han Zhao

Carnegie Mellon University\textsuperscript{1}, University of Amsterdam\textsuperscript{2}

June. 20th, 2016
Outline

Background
  Sum-Product Networks
  Variational Inference

Collapsed Variational Inference
  Motivations and Challenges
  Efficient Marginalization
  Logarithmic Transformation

Experiments

Summary
Sum-Product Networks

Definition

A Sum-Product Network (SPN) is a

- Rooted directed acyclic graph of univariate distributions, sum nodes and product nodes.
- Value of a product node is the product of its children.
- Value of a sum node is the weighted sum of its children, where the weights are nonnegative.
- Value of the network is the value at the root.
Sum-Product Networks

Mixture of Trees

Each SPN can be decomposed as a mixture of trees:

- Each tree is a product of univariate distributions.
- Number of mixture components is $\Omega(2^\text{Depth})$.
- Each network computes a positive polynomial (posynomial) function of model parameters:

$$V_{\text{root}}(x \mid w) = \sum_{t=1}^{\tau_S} \prod_{(k,j) \in T_t} w_{kj} \prod_{i=1}^{n} p_t(X_i = x_i)$$
Sum-Product Networks

Bayesian Network

Alternatively, each SPN $S$ is equivalent to a Bayesian network $B$ with bipartite structure.

▶ Number of sum nodes in $S = \text{Number of hidden variables in } B = \Theta(|S|)$. $|B| = O(n|S|)$

▶ Number of observable variables in $B = \text{Number of variables modeled by } S$.

▶ Typically number of hidden variables $\gg$ number of observable variables.
Variational Inference

Brief Introduction

Bayesian Inference:

\[
\frac{p(w \mid x)}{p(x \mid w)} \propto \frac{p(w)}{p(x \mid w)}
\]

Often intractable because of:

- No analytical solution.
- Expensive numerical integration.

General idea: find the best approximation in a tractable family of distributions \( Q \):

\[
\text{minimize}_{q \in Q} \quad KL[q(w) \mid \mid p(w \mid x)]
\]

Typical choice of approximation families: Mean-field, structured mean-field, etc.
Variational Inference

Brief Introduction

Variational method: Optimization-based, deterministic approach for approximate Bayesian inference.

\[ \inf_{q \in Q} KL[q(w) \parallel p(w \mid x)] \iff \sup_{q \in Q} \mathbb{E}_q[\log p(w, x)] + H[q] \]

Evidence Lower Bound \( \hat{\mathcal{L}} \):

\[ \log p(x) \geq \sup_{q \in Q} \mathbb{E}_q[\log p(w, x)] + H[q] =: \hat{\mathcal{L}} \]
Collapsed Variational Inference
Motivations and Challenges

Bayesian inference algorithms for SPNs:
- Flexible at incorporating prior knowledge about the structure of SPNs.
- More robust to overfitting.
Collapsed Variational Inference
Motivations and Challenges

- $W$ – Model parameters, global hidden variables.
- $H$ – Assignments of sum nodes, local hidden variables.
- $X$ – Observable variables.
- $D$ – Number of instances.

Challenges for standard VB:
- Large number of local hidden variables: number of local hidden variables $= \text{Number of sum nodes} = \Theta(|\mathcal{S}|)$. 

\[ W_1 \rightarrow W_2 \rightarrow W_3 \cdots \rightarrow W_m \]
\[ H_1 \rightarrow H_2 \rightarrow H_3 \cdots \rightarrow H_m \]
\[ X_1 \rightarrow X_2 \rightarrow X_3 \cdots \rightarrow X_n \]
Collapsed Variational Inference
Motivations and Challenges

\[ W_1 \rightarrow W_2 \rightarrow W_3 \cdots \rightarrow W_m \]
\[ H_1 \rightarrow H_2 \rightarrow H_3 \cdots \rightarrow H_m \]
\[ X_1 \rightarrow X_2 \rightarrow X_3 \cdots \rightarrow X_n \]

- \( W \) – Model parameters, global hidden variables.
- \( H \) – Assignments of sum nodes, local hidden variables.
- \( X \) – Observable variables.
- \( D \) – Number of instances.

Challenges for standard VB:

- Large number of local hidden variables: number of local hidden variables = Number of sum nodes = \( \Theta(|S|) \).
- Memory overhead: space complexity \( O(D|S|) \).
Collapsed Variational Inference
Motivations and Challenges

- $W$ – Model parameters, global hidden variables.
- $H$ – Assignments of sum nodes, local hidden variables.
- $X$ – Observable variables.
- $D$ – Number of instances.

Challenges for standard VB:
- Large number of local hidden variables: number of local hidden variables $=$ Number of sum nodes $= \Theta(|S|)$.
- Memory overhead: space complexity $O(D|S|)$.
- Time complexity: $O(nD|S|)$. 
Our contributions:

▶ We obtain better ELBO $\mathcal{L}$ to optimize than $\hat{\mathcal{L}}$, the one obtained by mean-field.

▶ Reduced space complexity: $O(D|S|) \Rightarrow O(|S|)$, space complexity is independent of training size.

▶ Reduced time complexity: $O(nD|S|) \Rightarrow O(D|S|)$, removing the explicit dependency on the dimension.
Collapsed Variational Inference

Efficient Marginalization

Recall ELBO in standard VI:

$$\hat{\mathcal{L}} := \mathbb{E}_{q(w,h)}[\log p(w, h, x)] + \mathbb{H}[q(w, h)]$$

Consider the new ELBO in Collapsed VI:

$$\mathcal{L} := \mathbb{E}_{q(w)}[\log p(w, x)] + \mathbb{H}[q(w)]$$

$$= \mathbb{E}_{q(w)}[\log \sum_h p(w, h, x)] + \mathbb{H}[q(w)]$$

We can establish the following inequality:

$$\log p(x) \geq \mathcal{L} \geq \hat{\mathcal{L}}$$

The new ELBO in Collapsed VI leads to a better lower bound than the one used in standard VI!
Collapsed Variational Inference

Comparisons

Standard Variational Inference
Mean-field assumption: \( q(w, h) = \prod_i q(w_i) \prod_j q(h_j) \)
ELBO: \( \hat{\mathcal{L}} := \mathbb{E}_{q(w,h)}[\log p(w, h, x)] + \mathbb{H}[q(w, h)] \)

Collapsed Variational Inference for LDA, HDP
Collapsed out global hidden variables: \( q(h) = \int_w q(w, h) \, dw \)
ELBO: \( \mathcal{L}_h := \mathbb{E}_{q(h)}[\log p(h, x)] + \mathbb{H}[q(h)] \)
Better lower bound: \( \mathcal{L}_h \geq \hat{\mathcal{L}} \)

Collapsed Variational Inference for SPN
Collapsed out local hidden variables: \( q(w) = \sum_h q(w, h) \)
ELBO: \( \mathcal{L}_w := \mathbb{E}_{q(w)}[\log p(w, x)] + \mathbb{H}[q(w)] \)
Better lower bound: \( \mathcal{L}_w \geq \hat{\mathcal{L}} \)
Collapsed Variational Inference
Efficient Marginalization

Time complexity of the exact marginalization incurred in computing \( \sum_h p(w, h, x) \):

- Time complexity of marginalization in graphical model \( \mathcal{G} \):
  \( O(D \cdot 2^{tw(\mathcal{G})}) \).

Space complexity reduction:
Collapsed Variational Inference
Efficient Marginalization

Time complexity of the exact marginalization incurred in computing $\sum_h p(w, h, x)$:

- Time complexity of marginalization in graphical model $G$: $O(D \cdot 2^{tw(G)})$.
- Exact marginalization in BN $B$ with algebraic decision diagram as local factors: $O(D|B|) = O(nD|S|)$.

Space complexity reduction:
Collapsed Variational Inference
Efficient Marginalization

Time complexity of the exact marginalization incurred in computing $\sum_h p(w, h, x)$:

- Time complexity of marginalization in graphical model $G$: $O(D \cdot 2^{tw(G)})$.

- Exact marginalization in BN $B$ with algebraic decision diagram as local factors: $O(D|B|) = O(nD|S|)$.

- Exact marginalization in SPN $S$: $O(D|S|)$.

Space complexity reduction:
Collapsed Variational Inference
Efficient Marginalization

Time complexity of the exact marginalization incurred in computing $\sum_{h} p(w, h, x)$:

- Time complexity of marginalization in graphical model $G$: $O(D \cdot 2^{\text{tw}(G)})$.
- Exact marginalization in BN $B$ with algebraic decision diagram as local factors: $O(D|B|) = O(nD|S|)$.
- Exact marginalization in SPN $S$: $O(D|S|)$.

Space complexity reduction:
- No posterior over $h$ to approximate anymore.
Collapsed Variational Inference
Efficient Marginalization

Time complexity of the exact marginalization incurred in computing \( \sum_h p(w, h, x) \):

- Time complexity of marginalization in graphical model \( G \): \( O(D \cdot 2^{tw(G)}) \).
- Exact marginalization in BN \( B \) with algebraic decision diagram as local factors: \( O(D|B|) = O(nD|S|) \).
- Exact marginalization in SPN \( S \): \( O(D|S|) \).

Space complexity reduction:

- No posterior over \( h \) to approximate anymore.
- No variational variables over \( h \) needed: \( O(D|S|) \Rightarrow O(|S|) \).
Collapsed Variational Inference

Logarithmic Transformation

New optimization objective:

$$\text{maximize}_{q \in Q} \mathbb{E}_{q(w)} [\log \sum_h p(w, h, x)] + \mathbb{H}[q(w)]$$

which is equivalent to

$$\text{minimize}_{q \in Q} \text{KL}[q(w) \| p(w)] - \mathbb{E}_{q(w)} [\log p(x \mid w)]$$

- $p(w)$ – prior distribution over $w$, product of Dirichlets.
- $q(w)$ – variational posterior over $w$, product of Dirichlets.
- $p(x \mid w)$ – likelihood, not multinomial anymore after marginalization.

Non-conjugate $q(w)$ and $p(x \mid w)$, no analytical solution for $\mathbb{E}_{q(w)} [\log p(x \mid w)]$. 
Collapsed Variational Inference

Logarithmic Transformation

Key observation:

\[ p(x \mid w) = V_{\text{root}}(x \mid w) = \sum_{t=1}^{\tau_S} \prod_{(k,j) \in T_t} w_{kj} \prod_{i=1}^{n} p_t(X_i = x_i) \]

is a posynomial function of \( w \).

Make a bijective mapping (change of variable): \( w' = \log(w) \).

- Dates back to the literature of geometric programming.
- The new objective after transformation is convex in \( w' \).

\[
\log p(x \mid w) = \log \left( \sum_{t=1}^{\tau_S} \exp \left( c_t + \sum_{(k,j) \in T_t} w'_{kj} \right) \right)
\]

Jensen’s inequality to obtain further lower bound.
Collapsed Variational Inference

Logarithmic Transformation

Further lower bound:

\[ \mathbb{E}_{q(w)}[\log p(x \mid w)] = \mathbb{E}_{q(w')}[\log p(x \mid w')] \geq \log p(x \mid \mathbb{E}_{q'(w')}[w']) \]

Relaxed objective:

\[
\text{minimize}_{q \in Q} \quad \underbrace{\text{KL}[q(w) \mid \mid p(w)]}_{\text{Regularity}} - \underbrace{\log p(x \mid \mathbb{E}_{q'(w')}[w'])}_{\text{Data fitting}}
\]

Roughly, \( \log p(x \mid \mathbb{E}_{q'(w')}[w']) \) corresponds the log-likelihood by setting the weights of SPN as the posterior mean of \( q(w) \).

Optimized by projected GD.
Collapsed Variational Inference

Algorithm

Algorithm 1 CVB-SPN

**Input:** Initial $\beta$, prior hyperparameter $\alpha$, training instances $\{x_d\}_{d=1}^{D}$.

**Output:** Locally optimal $\beta^*$.

1: while not converged do
2: Update $w = \exp(\mathbb{E}_{q'(w'|\beta)}[w'])$ with Eq. 10.
3: Set $\nabla_\beta = 0$.
4: for $d = 1$ to $D$ do
5: Bottom-up evaluation of $\log p(x_d|w)$.
6: Top-down differentiation of $\frac{\partial}{\partial w} \log p(x_d|w)$.
7: Update $\nabla_\beta$ based on $x_d$.
8: end for
9: Update $\nabla_\beta$ based on $\text{KL}(q(w|\beta) \parallel p(w|\alpha))$.
10: Update $\beta$ with projected GD.
11: end while

- Line 4 – 8 easily parallelizable, distributed version.
- Sample minibatch in Line 4 – 8, stochastic version.
Experiments

- Experiments on 20 data sets, report average log-likelihoods, Wilcoxon ranked test.
- Compared with (O)MLE-SPN and OBMM-SPN.
Experiments

OBMM vs OCVB, Avg. log-likelihoods

OBMM-Moment Matching
OCVB-Projected

Carnegie Mellon University
Thanks

Q & A