Chapter 19

Approximate Inference

Inference

- Inference refers to computing conditional distributions
- For example, in probabilistic models with latent $\boldsymbol{h}$ and visible $\boldsymbol{v}$ units, we usually wish to compute

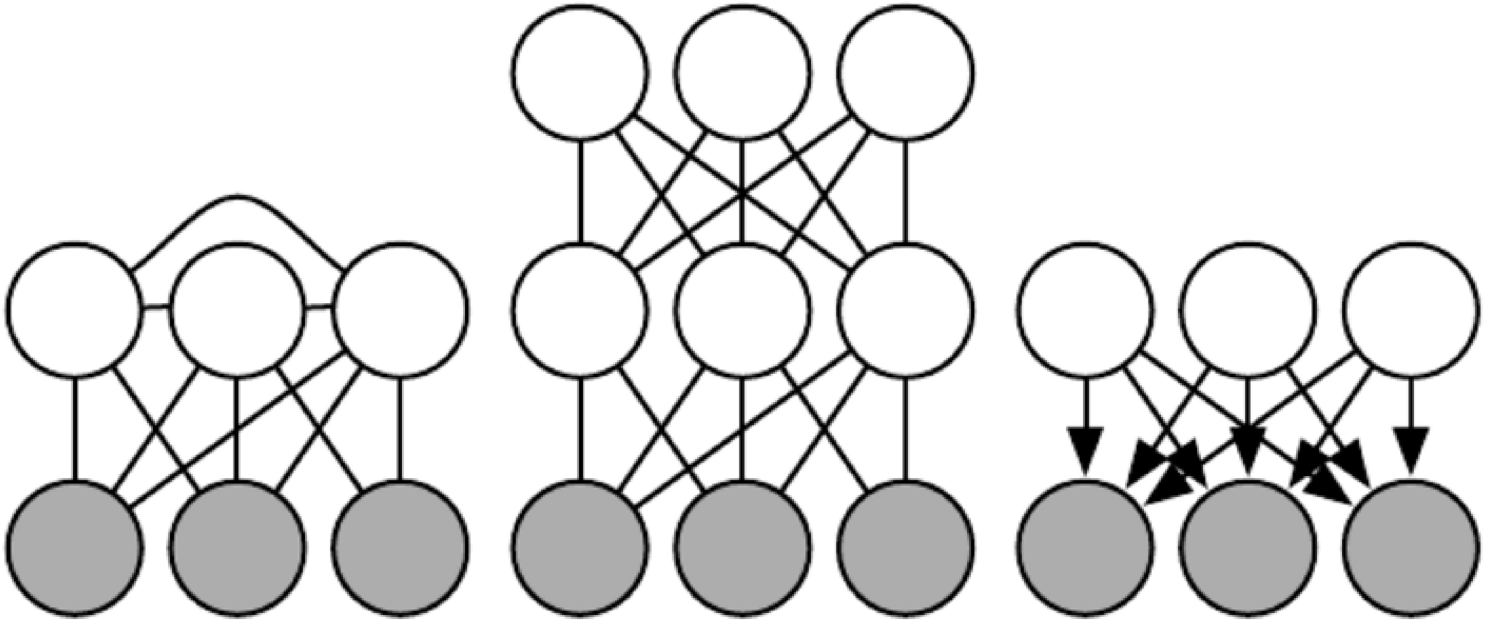
$$p(\boldsymbol{h}|\boldsymbol{v})$$

- An alternative form of inference is to compute

$$\boldsymbol{h}^* = \arg\max_{\boldsymbol{h}} p(\boldsymbol{h}|\boldsymbol{v})$$

- In graphical models with multiple layers of hidden units, it is usually difficult to evaluate them exactly

- The difficulty arises from the interactions between latent variables $\boldsymbol{h}$ in computing the posterior $p(\boldsymbol{h}|\boldsymbol{v})$



Inference as Optimization

- Recall from the EM algorithm

$$\log p(\boldsymbol{v}; \boldsymbol{\theta}) = \mathcal{L}(\boldsymbol{v}, q, \boldsymbol{\theta}) + \mathsf{KL}(q(\boldsymbol{h}) || p(\boldsymbol{h}|\boldsymbol{v}; \boldsymbol{\theta}))$$

where

$$\begin{aligned} \mathcal{L}(\boldsymbol{v}, q, \boldsymbol{\theta}) &= \int q(\boldsymbol{h}) \log p(\boldsymbol{v}, \boldsymbol{h}; \boldsymbol{\theta}) d\boldsymbol{h} - \int q(\boldsymbol{h}) \log q(\boldsymbol{h}) d\boldsymbol{h} \\ &= E_{\boldsymbol{h} \sim q} \log p(\boldsymbol{v}, \boldsymbol{h}; \boldsymbol{\theta}) - H(q) \\ \mathsf{KL}(q(\boldsymbol{h}) || p(\boldsymbol{h}|\boldsymbol{v}; \boldsymbol{\theta})) &= \int q(\boldsymbol{h}) \log \frac{q(\boldsymbol{h})}{p(\boldsymbol{h}|\boldsymbol{v}; \boldsymbol{\theta})} d\boldsymbol{h} \end{aligned}$$

- Some changes of variables (e.g. $\boldsymbol{X} \to \boldsymbol{v}$ and $\boldsymbol{Z} \to \boldsymbol{h}$) have been made for consistent notation

- Since the KL divergence is non-negative, $\mathrm{KL}(q||p) \geq 0$, it follows that

$$\log p(\boldsymbol{v};\boldsymbol{\theta}) \geq \mathcal{L}(\boldsymbol{v}, q, \boldsymbol{\theta})$$

with equality if and only if

$$q(\boldsymbol{h}) = p(\boldsymbol{h}|\boldsymbol{v};\boldsymbol{\theta})$$

- In other words, $\mathcal{L}(\boldsymbol{v}, q, \boldsymbol{\theta})$ is a lower bound on $\log p(\boldsymbol{v};\boldsymbol{\theta})$ for any choice of $q(\boldsymbol{h})$

- The bound $\mathcal{L}$ is closer to $\log p(\boldsymbol{v};\boldsymbol{\theta})$ when $q(\boldsymbol{h})$ is closer to $p(\boldsymbol{h}|\boldsymbol{v};\boldsymbol{\theta})$

- Inference
  - Exact inference maximizes $\mathcal{L}(\boldsymbol{v}, q, \boldsymbol{\theta})$ perfectly by searching over a family of functions $q(\boldsymbol{h})$ that includes $p(\boldsymbol{h}|\boldsymbol{v};\boldsymbol{\theta})$
  - Approximate inference makes optimization approximate by restricting the $q(\boldsymbol{h})$'s to search over, or by imperfect optimization
- Learning
  - Maximize $\mathcal{L}(\boldsymbol{v}, q, \boldsymbol{\theta})$ w.r.t. $\boldsymbol{\theta}$ for a given $q(\boldsymbol{h})$
- Combined inference and learning is possible via algorithms such as coordinate ascent

Revisiting Kullback-Leibler (KL) Divergence

- In maximizing $\mathcal{L}(\boldsymbol{v}, q, \boldsymbol{\theta})$, we are minimizing $\mathsf{KL}(q||p)$, which yields an effect different from minimizing $\mathsf{KL}(p||q)$

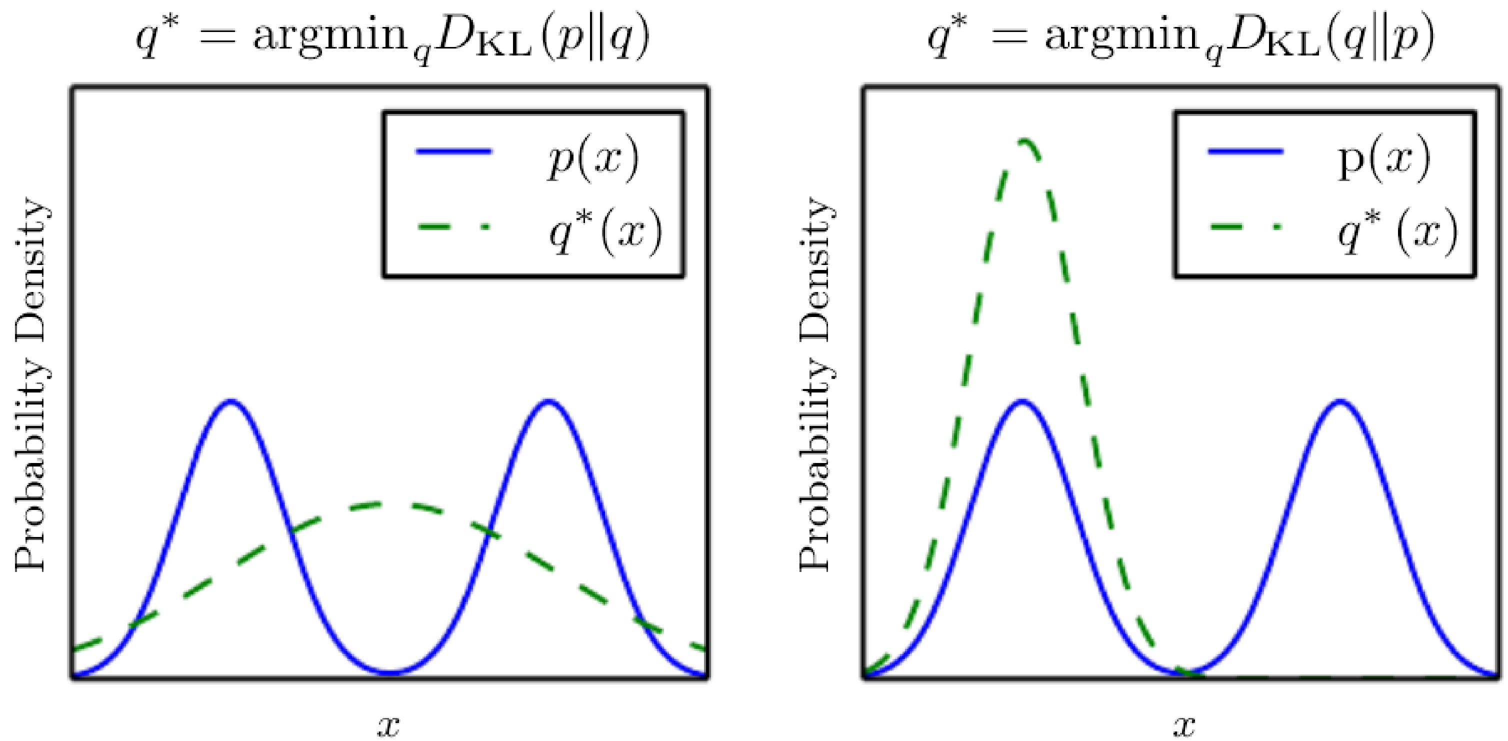
$$q^* = \arg\min_q \mathsf{KL}(p||q)$$

v.s.

$$q^* = \arg\min_q \mathsf{KL}(q||p)$$

where

$$\mathsf{KL}(p||q) = \int p(x)(\log \frac{1}{q(x)} - \log \frac{1}{p(x)})dx$$
$$\mathsf{KL}(q||p) = \int q(x)(\log \frac{1}{p(x)} - \log \frac{1}{q(x)})dx$$



– KL$(p||q)$ places high probability where data occur

– KL$(q||p)$ places low probability where data do not occur

Calculus of Variations

- Learning algorithms are mainly based on minimizing a function $J(\boldsymbol{\theta})$ by solving for the critical points

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = \mathbf{0}$$

- Calculus of variations solves for a function $f(\boldsymbol{x})$ to maximize a functional $J(f(\boldsymbol{x}))$ by taking functional derivatives, a.k.a variational derivatives, w.r.t. the value of the function f at point $\boldsymbol{x}$

$$\frac{\delta}{\delta f(\boldsymbol{x})} J$$

- Example: To find a $p(x)$ that maximizes differential entropy

$$\arg\max_{p(x)} H[p(x)] = -\int p(x)\log p(x)dx$$

subject to

$$\int p(x)dx = 1$$
$$E[x] = \mu$$
$$E[(x-\mu)^2] = \sigma^2$$

- This constrained optimization problem can be solved by setting up a Lagrangian functional

$$\mathcal{L}(p,\lambda_1,\lambda_2,\lambda_3) = -\int p(x)\log p(x)dx + \lambda_1\left(\int p(x)dx - 1\right) + \lambda_2\left(\int p(x)dx - \mu\right) + \lambda_3\left(\int (x-\mu)^2 p(x)dx - \sigma^2\right)$$

- We then set the functional derivative w.r.t. $p(x)$ to 0

$$\forall x, \frac{\delta}{\delta p(x)}\mathcal{L} = -\log p(x) - 1 + \lambda_1 + \lambda_2 + \lambda_3(x-\mu)^2 = 0$$

- It is seen that the optimal $p(x)$ has a form of Gaussian and can be solved exactly as a Gaussian after all the λ's are solved

$$p(x) = \exp(\lambda_1 + \lambda_2 x + \lambda_3(x-\mu)^2 - 1) = \mathcal{N}(x; \mu, \sigma^2)$$

Variational Inference

- Variational inference is to find $p(\boldsymbol{h}|\boldsymbol{v})$ by maximizing $\mathcal{L}(\boldsymbol{v}, q, \boldsymbol{\theta})$ w.r.t. $q(\boldsymbol{h})$, which can be any function of $\boldsymbol{h}$

$$\mathcal{L}(\boldsymbol{v}, q, \boldsymbol{\theta}) = E_{\boldsymbol{h}\sim q} \log p(\boldsymbol{v}, \boldsymbol{h}; \boldsymbol{\theta}) - H(q)$$

- An approximate framework (known as the **mean field approximation**) of variational inference assumes $q(\boldsymbol{h})$ is factorial

$$q(\boldsymbol{h}) = \prod_{i=1}^{M} q_i(\boldsymbol{h}_i)$$

where the elements of $\boldsymbol{h}$ is partitioned into disjoint groups $\boldsymbol{h}_i$'s

- **No further assumption is made about the forms of** $q_i(\boldsymbol{h}_i)$

- We now seek the $q_i(\boldsymbol{h}_i)$ for which the lower bound is the largest by making a free form (variational) optimization w.r.t. each of them

$$
\begin{aligned}
&\mathcal{L}(\boldsymbol{v}, q, \boldsymbol{\theta}) \\
&= \int q(\boldsymbol{h}) \log p(\boldsymbol{v}, \boldsymbol{h}; \boldsymbol{\theta}) d\boldsymbol{h} - \int q(\boldsymbol{h}) \log q(\boldsymbol{h}) d\boldsymbol{h} \\
&= \int \prod_i q_i(\boldsymbol{h}_i) \left\{ \log p(\boldsymbol{v}, \boldsymbol{h}; \boldsymbol{\theta}) - \sum_i \log q_i(\boldsymbol{h}_i) \right\} d\boldsymbol{h} \\
&= \int q_j(\boldsymbol{h}_j) \left\{ \int \log p(\boldsymbol{v}, \boldsymbol{h}; \boldsymbol{\theta}) \prod_{i \neq j} q_i(\boldsymbol{h}_i) d\boldsymbol{h}_i \right\} d\boldsymbol{h}_j \\
&\quad - \int q_j(\boldsymbol{h}_j) \log q_j(\boldsymbol{h}_j) d\boldsymbol{h}_j + \mathsf{const} \\
&= \int q_j(\boldsymbol{h}_j) \log \tilde{p}(\boldsymbol{v}, \boldsymbol{h}_j) d\boldsymbol{h}_j - \int q_j(\boldsymbol{h}_j) \log q_j(\boldsymbol{h}_j) d\boldsymbol{h}_j + \mathsf{const} \\
&= -\mathsf{KL}(q_j(\boldsymbol{h}_j) || \tilde{p}(\boldsymbol{v}, \boldsymbol{h}_j)) + \mathsf{const}
\end{aligned}
$$

where we have defined

$$\log \tilde{p}(\boldsymbol{v}, \boldsymbol{h}_j) = \int \log p(\boldsymbol{v}, \boldsymbol{h}; \boldsymbol{\theta}) \prod_{i \neq j} q_i(\boldsymbol{h}_i) d\boldsymbol{h}_i$$
$$= E_{i \neq j}(\log p(\boldsymbol{v}, \boldsymbol{h}; \boldsymbol{\theta})) + \text{const}$$

- It is seen that $\mathcal{L}$ is maximized when

$$q_j(\boldsymbol{h}_j) = \tilde{p}(\boldsymbol{v}, \boldsymbol{h}_j) = \frac{1}{Z} \exp(E_{i \neq j}(\log p(\boldsymbol{v}, \boldsymbol{h}; \boldsymbol{\theta})))$$

- Alternatively, we have the unnormalized distribution $\tilde{q}_j(\boldsymbol{h}_j)$

$$\tilde{q}_j(\boldsymbol{h}_j) = \exp(E_{i \neq j}(\log p(\boldsymbol{v}, \boldsymbol{h}; \boldsymbol{\theta})))$$

- Carrying out the expectation $E_{i \neq j}(\log p(\boldsymbol{v}, \boldsymbol{h}; \boldsymbol{\theta}))$ will yield the correct functional form for $\tilde{q}_j(\boldsymbol{h}_j)$

- Toy problem: Consider a probabilistic model with latent variables $\boldsymbol{h} \in R^2$ and one visible variable $v \in R^1$

$$p(\boldsymbol{h}) = \mathcal{N}(\boldsymbol{h}; \boldsymbol{0}, \boldsymbol{I})$$

$$p(v|\boldsymbol{h}) = \mathcal{N}(v; \boldsymbol{w}^T \boldsymbol{h}, 1)$$

- The true posterior is given by

$$\begin{aligned} p(\boldsymbol{h}|v) &= \frac{p(\boldsymbol{h})p(v|\boldsymbol{h})}{p(v)} \\ &= \mathcal{N}(\boldsymbol{h}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) \end{aligned}$$

where

$$\boldsymbol{\Sigma}^{-1} = \boldsymbol{I} + \boldsymbol{w}\boldsymbol{w}^T$$

$$\boldsymbol{\mu} = v\boldsymbol{\Sigma}\boldsymbol{w}$$

- Obviously, $p(\boldsymbol{h}|v)$ is not factorial

- Now, applying variational inference, we have

$$p(\boldsymbol{h}|v) \approx q(\boldsymbol{h}|v) = q(h_1|v)q(h_2|v)$$

$$\begin{aligned}
\tilde{q}(h_1|v) \quad &= \exp(E_{h_2 \sim q(h_2|v)}(\log p(\boldsymbol{v}, \boldsymbol{h}; \boldsymbol{\theta}))) \\
&= \exp(-\frac{1}{2} E_{h_2 \sim q(h_2|v)}[h_1^2 + h_2^2 + v^2 + h_1^2 w_1^2 + h_2^2 w_2^2 \\
&\quad - 2vh_1w_1 - 2vh_2w_2 + 2h_1w_1h_2w_2]) \\
&= \exp(-\frac{1}{2}[h_1^2 + \langle h_2^2 \rangle + v^2 + h_1^2 w_1^2 + \langle h_2^2 \rangle w_2^2 \\
&\quad - 2vh_1w_1 - 2v\langle h_2 \rangle w_2 + 2h_1w_1\langle h_2 \rangle w_2]) \\
&= \exp(-\frac{1}{2}[(1 + w_1^2)h_1^2 - 2(vw_1 - w_1\langle h_2 \rangle w_2)h_1 + \mathsf{const}])
\end{aligned}$$

- From the above, we can deduce

$$q(h_1|v) = \mathcal{N}(h_1; \mu_1, \sigma_1^2)$$

where

$$\sigma_1^2 = \frac{1}{1+w_1^2}$$

$$\mu_1 = \frac{vw_1 - w_1\langle h_2\rangle w_2}{1+w_1^2} = \frac{vw_1 - w_1\mu_2 w_2}{1+w_1^2}$$

- On the symmetry ground, we can readily arrive at

$$q(h_2|v) = \mathcal{N}(h_2; \mu_2, \sigma_2^2)$$

where

$$\sigma_2^2 = \frac{1}{1+w_2^2}$$

$$\mu_2 = \frac{vw_2 - w_2\langle h_1\rangle w_1}{1+w_2^2} = \frac{vw_2 - w_2\mu_1 w_1}{1+w_2^2}$$

- Note that we do not assume $q(\boldsymbol{h}|v) = q(h_1|v)q(h_2|v)$ is Gaussian; its Gaussian form is derived automatically

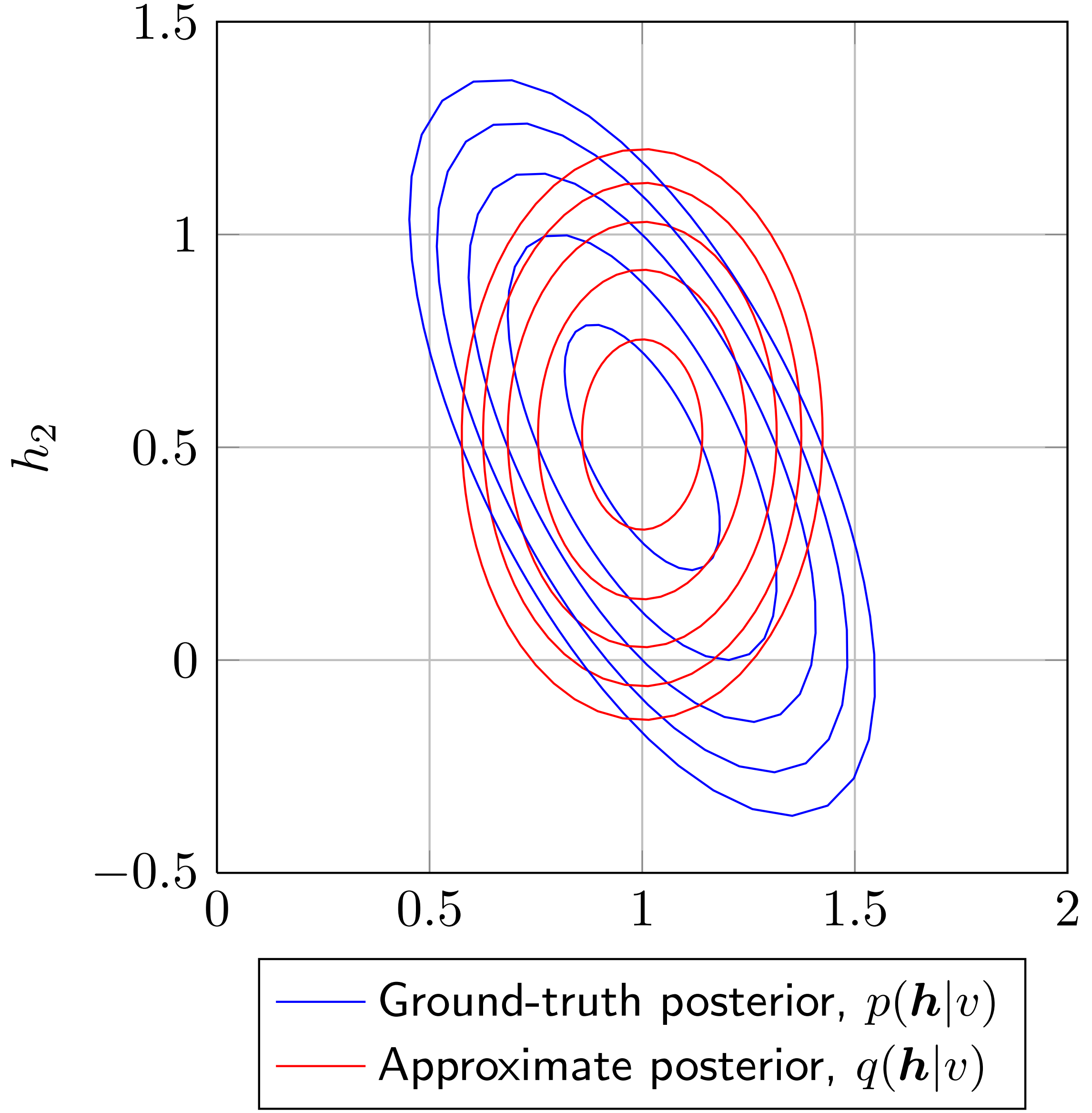
- Moreover, given the data v and the model parameters w_1, w_2, we have the following fixed-point update equations to estimate u_1, u_2 iteratively

$$\mu_1 = \frac{vw_1 - w_1\mu_2 w_2}{1 + w_1^2}$$
$$\mu_2 = \frac{vw_2 - w_2\mu_1 w_1}{1 + w_2^2}$$

- Note that the process needs to be carried out for each new value of v
- The following compares the ground-true posterior $p(\boldsymbol{h}|v)$ with the variational approximation $q(\boldsymbol{h}|v)$, with

$$\boldsymbol{w} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}, \; v = 3$$

and the fixed-point update equations run for 10 iterations


1.5
1
0.5
0
−0.5
h_2
0
0.5
1
1.5
2
Ground-truth posterior, $p(\boldsymbol{h}|v)$
Approximate posterior, $q(\boldsymbol{h}|v)$

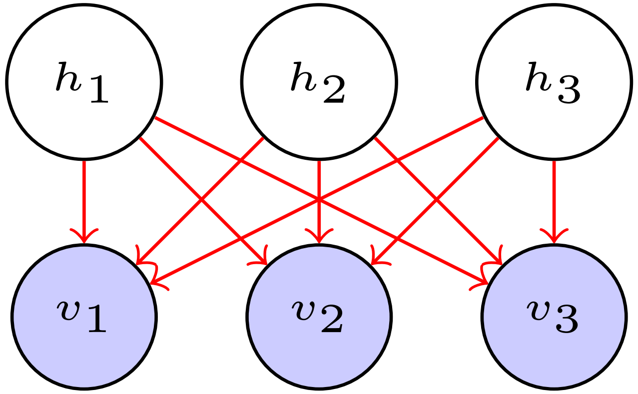
Learned Approximate Inference

- **Idea:** To replace the iterative optimization with a neural network $f(\boldsymbol{v};\boldsymbol{\theta}')$ to form an approximate inference distribution $q(\boldsymbol{h}|\boldsymbol{v})$
- **Wake-Sleep algorithm** (in sleep mode) is to learn the mapping from $\boldsymbol{v}$ to $\boldsymbol{h}$ by drawing $\boldsymbol{v}, \boldsymbol{h}$ samples from the model distribution $p(\boldsymbol{v}, \boldsymbol{h}; \boldsymbol{\theta})$ while the model is still evolving in the learning process
- One drawback is that we will only be able to train $f(\boldsymbol{v};\boldsymbol{\theta}')$ on $\boldsymbol{v}$'s that have high probability under the evolving model $p(\boldsymbol{v}, \boldsymbol{h}; \boldsymbol{\theta})$

Sparse Coding and MAP Inference

- Sparse coding can be interpreted as a special case of combined variational inference and learning
- Combined variational inference and learning
  1. Find approximate inference q by maximizing $\mathcal{L}(\boldsymbol{v}, q, \boldsymbol{\theta})$ w.r.t. q
  2. Find model parameters $\boldsymbol{\theta}$ by maximizing $\mathcal{L}(\boldsymbol{v}, q, \boldsymbol{\theta})$ w.r.t. $\boldsymbol{\theta}$
  3. Repeat steps 1, 2 until the stopping criteria are met

- Linear factor model for sparse coding



$$p(\boldsymbol{h}) = \prod_i \frac{\lambda}{2} e^{-\lambda |h_i|}$$

$$p(\boldsymbol{v}|\boldsymbol{h}) = \mathcal{N}(\boldsymbol{v}; \boldsymbol{W}\boldsymbol{h} + b, \beta^{-1}\boldsymbol{I})$$

where λ, β are hyper parameters and are assumed to be known

- In learning the model parameters $\boldsymbol{W}, b$, the EM does not work because $p(\boldsymbol{h}|\boldsymbol{v})$ is intractable

- We thus resort to approximate variational inference by restricting $q(\boldsymbol{h}) \approx p(\boldsymbol{h}|\boldsymbol{v})$ to the following family

$$q(\boldsymbol{h}; \boldsymbol{\mu}) = \delta(\boldsymbol{h} - \boldsymbol{\mu})$$

  where $\delta(\cdot)$ is the Dirac delta function

- Maximizing $\mathcal{L}(\boldsymbol{v}, q, \boldsymbol{\theta}) = E_{\boldsymbol{h}\sim q} \log p(\boldsymbol{v}, \boldsymbol{h}; \boldsymbol{\theta}) - H(q)$ w.r.t. q arrives at

$$\begin{aligned}
\boldsymbol{u}^* &= \arg\max_{\boldsymbol{\mu}} \log p(\boldsymbol{v}, \boldsymbol{h} = \boldsymbol{u}; \boldsymbol{\theta}) = \underbrace{\arg\max_{\boldsymbol{\mu}} p(\boldsymbol{h} = \boldsymbol{u}|\boldsymbol{v}; \boldsymbol{\theta})}_{\text{MAP inference}} \\
&= \arg\max_{\boldsymbol{\mu}} -\lambda \sum_i |u_i| - \frac{\beta}{2} \|\boldsymbol{v} - \boldsymbol{W}\boldsymbol{u} - \boldsymbol{b}\|_2^2 \\
&= \arg\min_{\boldsymbol{\mu}} \lambda \sum_i |u_i| + \frac{\beta}{2} \|\boldsymbol{v} - \boldsymbol{W}\boldsymbol{u} - \boldsymbol{b}\|_2^2
\end{aligned}$$

- With this $q(\boldsymbol{h};\boldsymbol{\mu}^*) = \delta(\boldsymbol{h} - \boldsymbol{\mu}^*)$, maximizing $\mathcal{L}(\boldsymbol{v}, q, \boldsymbol{\theta})$ w.r.t. $\boldsymbol{\theta}$ gives

$$\begin{aligned}
&\arg\max_{\boldsymbol{\theta}} \log p(\boldsymbol{v}, \boldsymbol{h} = \boldsymbol{\mu}^*; \boldsymbol{\theta}) \\
&= \arg\min_{\boldsymbol{W},\boldsymbol{b}} \lambda \sum_i |u_i^*| + \frac{\beta}{2} \|\boldsymbol{v} - \boldsymbol{W}\boldsymbol{u}^{\boldsymbol{*}} - \boldsymbol{b}\|_2^2
\end{aligned}$$

- To summarize, we alternate between minimization w.r.t. $\boldsymbol{u}$ and $\boldsymbol{W}, b$

$$J(\boldsymbol{u}, \boldsymbol{W}, \boldsymbol{b}) = \lambda \sum_i |u_i| + \frac{\beta}{2} \|\boldsymbol{v} - \boldsymbol{W}\boldsymbol{u} - \boldsymbol{b}\|_2^2$$

  where $\boldsymbol{u}$ is known as the sparse code for the input $\boldsymbol{v}$

- The procedure corresponds to the conventional sparse coding and a form of coordinate ascent on $\mathcal{L}(\boldsymbol{v}, q, \boldsymbol{\theta})$

Interaction between Learning and Inference

- Using approximate inference as part of the learning algorithm affects the learning process; this in turn affects the inference accuracy
- In learning the model parameters, variational learning increases

$$E_{\boldsymbol{h}\sim q(\boldsymbol{h})} p(\boldsymbol{v}, \boldsymbol{h})$$

- For a specific $\boldsymbol{v}$, this has an effect of learning a model that
  - Increases $p(\boldsymbol{h}|\boldsymbol{v})$ where $q(\boldsymbol{h}|\boldsymbol{v})$ has high probability
  - Decreases $p(\boldsymbol{h}|\boldsymbol{v})$ where $q(\boldsymbol{h}|\boldsymbol{v})$ has low probability
- As a result, the approximate assumptions (e.g. the mean field assumption) become self-fulfilling prophecies
- The true amount of harm due to approximate inference is generally hard to estimate

Review

- Inference
- Inference as optimization
- KL$(q||p)$ vs KL$(p||q)$
- Calculus of variations
- Variational inference
- Combined variational inference and learning