Variational inference

Rico Krueger

Filipe Rodrigues

DTU Management Engineering
Department of Management Engineering

Learning objectives

- Explain the concept of deterministic approximate inference.
- Explain variational inference.
- Describe differences between variational inference and expectation propagation.
- Derive and explain the evidence lower bound.
- Apply mean-field variational inference.
- Explain and apply Variational Bayesian Expectation-Maximization.

Outline

- Introduction
- Variational inference
- Variational Bayesian Expectation-Maximization

Approximate inference

- Suppose we wish to compute the posterior distribution of the latent variables $\mathbf{z}$ given some observed data $\mathbf{x}$: $p(\mathbf{z}|\mathbf{x})$
- For many problems of interest exact posterior inference is **intractable**

$$\overbrace{p(\mathbf{z}|\mathbf{x})}^{\text{posterior}} = \frac{\overbrace{p(\mathbf{x},\mathbf{z})}^{\text{joint}}}{p(\mathbf{x})} = \frac{\overbrace{p(\mathbf{x}|\mathbf{z})}^{\text{likelihood}}\ \overbrace{p(\mathbf{z})}^{\text{prior}}}{\underbrace{p(\mathbf{x})}_{\text{evidence}}} = \frac{\overbrace{p(\mathbf{x}|\mathbf{z})}^{\text{likelihood}}\ \overbrace{p(\mathbf{z})}^{\text{prior}}}{\underbrace{\sum_{\mathbf{z}} p(\mathbf{x}|\mathbf{z})\, p(\mathbf{z})}_{\text{evidence}}}$$

  - Cannot determine the posterior distribution analytically
  - Cannot even compute expectations with respect to the posterior

Approximate inference

- Suppose we wish to compute the posterior distribution of the latent variables $\mathbf{z}$ given some observed data $\mathbf{x}$: $p(\mathbf{z}|\mathbf{x})$
- For many problems of interest exact posterior inference is **intractable**

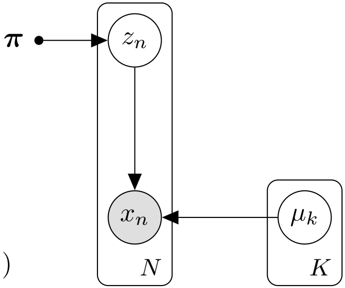
$$\overbrace{p(\mathbf{z}|\mathbf{x})}^{\text{posterior}} = \frac{\overbrace{p(\mathbf{x},\mathbf{z})}^{\text{joint}}}{p(\mathbf{x})} = \frac{\overbrace{p(\mathbf{x}|\mathbf{z})}^{\text{likelihood}}\ \overbrace{p(\mathbf{z})}^{\text{prior}}}{\underbrace{p(\mathbf{x})}_{\text{evidence}}} = \frac{\overbrace{p(\mathbf{x}|\mathbf{z})}^{\text{likelihood}}\ \overbrace{p(\mathbf{z})}^{\text{prior}}}{\underbrace{\sum_{\mathbf{z}} p(\mathbf{x}|\mathbf{z})\, p(\mathbf{z})}_{\text{evidence}}}$$

  - Cannot determine the posterior distribution analytically
  - Cannot even compute expectations with respect to the posterior

- Reasons for intractability:
  - Dimensionality of the latent space is too high to work with directly
  - Posterior distribution has a highly complex form for which expectations are not analytically tractable

- We must resort to **approximate inference** methods!

A practical example: Bayesian Gaussian mixture model



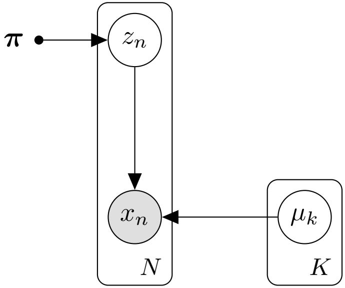
- For the sake of simplicity, assume $\boldsymbol{\pi}$ is given (fixed)

1. For each cluster k:
   a) Draw cluster center $\mu_k \sim \mathcal{N}(\mu_k | 0, \tau^2)$
2. For each data point $1, \ldots, N$:
   a) Draw cluster assignment $z_n \sim \text{Multinomial}(z_n | \boldsymbol{\pi})$
   b) Draw observed data point $x_n \sim \mathcal{N}(x_n | \mu_{z_n}, \sigma^2)$

Note

For simplicity, we assumed the cluster variances σ^2 to be fixed.

A practical example: Bayesian Gaussian mixture model

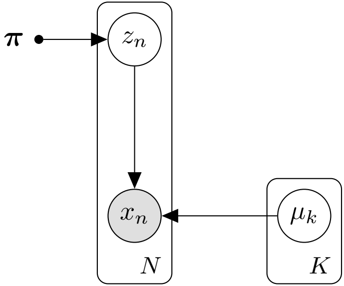


- Joint distribution is given by (ignoring the fixed parameters)

$$p(\boldsymbol{\mu}, \mathbf{z}, \mathbf{x}) = \left(\prod_{k=1}^{K} p(\mu_k) \right) \prod_{n=1}^{N} p(z_n|\boldsymbol{\pi})\, p(x_n|z_n, \boldsymbol{\mu})$$

where $\boldsymbol{\mu} = \{\mu_1, \ldots, \mu_K\}$, $\mathbf{x} = \{x_1, \ldots, x_N\}$ and $\mathbf{z} = \{z_1, \ldots, z_N\}$

A practical example: Bayesian Gaussian mixture model



- Joint distribution is given by (ignoring the fixed parameters)

$$p(\boldsymbol{\mu}, \mathbf{z}, \mathbf{x}) = \left(\prod_{k=1}^{K} p(\mu_k) \right) \prod_{n=1}^{N} p(z_n | \boldsymbol{\pi})\, p(x_n | z_n, \boldsymbol{\mu})$$

where $\boldsymbol{\mu} = \{\mu_1, \dots, \mu_K\}$, $\mathbf{x} = \{x_1, \dots, x_N\}$ and $\mathbf{z} = \{z_1, \dots, z_N\}$

- Our goal is to compute the posterior over $\boldsymbol{\mu}$ and $\mathbf{z}$

$$\color{red}{p(\boldsymbol{\mu}, \mathbf{z} | \mathbf{x})} = \frac{p(\boldsymbol{\mu}, \mathbf{z}, \mathbf{x})}{\color{green}{\int_{\boldsymbol{\mu}} \sum_{\mathbf{z}} p(\boldsymbol{\mu}, \mathbf{z}, \mathbf{x})}}$$

A practical example: Bayesian Gaussian mixture model

$$p(\boldsymbol{\mu}, \mathbf{z}|\mathbf{x}) = \frac{p(\boldsymbol{\mu}, \mathbf{z}, \mathbf{x})}{\int_{\boldsymbol{\mu}} \sum_{\mathbf{z}} p(\boldsymbol{\mu}, \mathbf{z}, \mathbf{x})}$$

- The numerator is easy to evaluate for any configuration of the hidden variables

A practical example: Bayesian Gaussian mixture model

$$p(\boldsymbol{\mu}, \mathbf{z}|\mathbf{x}) = \frac{p(\boldsymbol{\mu}, \mathbf{z}, \mathbf{x})}{\int_{\boldsymbol{\mu}} \sum_{\mathbf{z}} p(\boldsymbol{\mu}, \mathbf{z}, \mathbf{x})}$$

- The numerator is easy to evaluate for any configuration of the hidden variables
- The problem is the marginal likelihood in the denominator

$$\begin{aligned} p(\mathbf{x}) &= \int_{\boldsymbol{\mu}} \sum_{\mathbf{z}} p(\boldsymbol{\mu}, \mathbf{z}, \mathbf{x}) \\ &= \int_{\boldsymbol{\mu}} \sum_{\mathbf{z}} \left(\prod_{k=1}^{K} p(\mu_k) \right) \prod_{n=1}^{N} p(z_n|\boldsymbol{\pi})\, p(x_n|z_n, \boldsymbol{\mu}) \end{aligned}$$

A practical example: Bayesian Gaussian mixture model

$$p(\boldsymbol{\mu}, \mathbf{z}|\mathbf{x}) = \frac{p(\boldsymbol{\mu}, \mathbf{z}, \mathbf{x})}{\int_{\boldsymbol{\mu}} \sum_{\mathbf{z}} p(\boldsymbol{\mu}, \mathbf{z}, \mathbf{x})}$$

- The numerator is easy to evaluate for any configuration of the hidden variables
- The problem is the marginal likelihood in the denominator

$$\begin{aligned} p(\mathbf{x}) &= \int_{\boldsymbol{\mu}} \sum_{\mathbf{z}} p(\boldsymbol{\mu}, \mathbf{z}, \mathbf{x}) \\ &= \int_{\boldsymbol{\mu}} \sum_{\mathbf{z}} \left(\prod_{k=1}^{K} p(\mu_k) \right) \prod_{n=1}^{N} p(z_n|\boldsymbol{\pi})\, p(x_n|z_n, \boldsymbol{\mu}) \end{aligned}$$

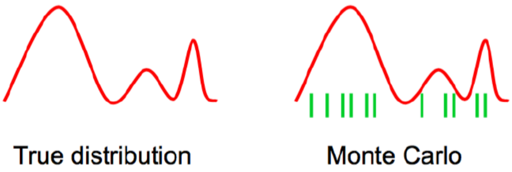
- Taking advantage of the conditional independence of the z_n's:

$$p(\mathbf{x}) = \int_{\boldsymbol{\mu}} \left(\prod_{k=1}^{K} p(\mu_k) \right) \prod_{n=1}^{N} \sum_{z_n} p(z_n|\boldsymbol{\pi})\, p(x_n|z_n, \boldsymbol{\mu})$$

- But this integral is still **intractable** to compute!

Approximate inference: stochastic methods

- Obtain a set of **samples** $\mathbf{z}^{(s)}$, for $s \in \{1, \ldots, S\}$, drawn independently from the distribution $p(\mathbf{z}|\mathbf{x})$

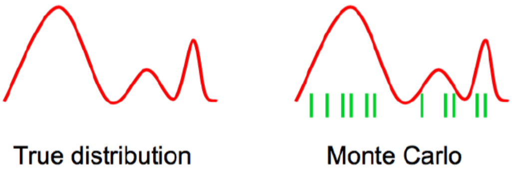


- Allows to approximate expectations as finite sums

$$\mathbb{E}_{p(\mathbf{z}|\mathbf{x})}[f(\mathbf{z})] \approx \frac{1}{S} \sum_{s=1}^{S} f(\mathbf{z}^{(s)})$$

Approximate inference: stochastic methods

- Obtain a set of **samples** $\mathbf{z}^{(s)}$, for $s \in \{1, \dots, S\}$, drawn independently from the distribution $p(\mathbf{z}|\mathbf{x})$



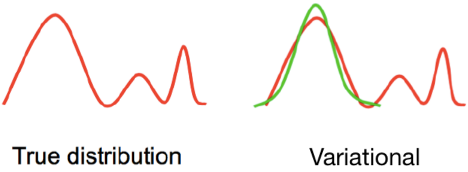
- Allows to approximate expectations as finite sums

$$\mathbb{E}_{p(\mathbf{z}|\mathbf{x})}[f(\mathbf{z})] \approx \frac{1}{S} \sum_{s=1}^{S} f(\mathbf{z}^{(s)})$$

- In the limit of infinite computational resources they can generate exact results
- **Computationally demanding** and hard to scale to large datasets
- Many practical problems: determining convergence, number of samples, burn-in size, thinning, hard to diagnose, etc.

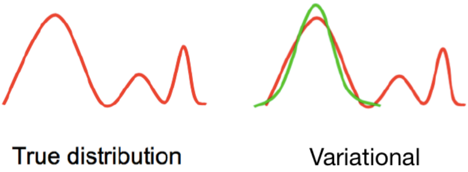
Approximate inference: variational methods

- Consider a family of tractable distributions $q(\mathbf{z}|\boldsymbol{\nu})$ - the **variational distribution**
- Find the **variational parameters** $\boldsymbol{\nu}$ that make $q(\mathbf{z}|\boldsymbol{\nu})$ as close as possible to the true posterior $p(\mathbf{z}|\mathbf{x})$



Approximate inference: variational methods

- Consider a family of tractable distributions $q(\mathbf{z}|\boldsymbol{\nu})$ - the **variational distribution**
- Find the **variational parameters** $\boldsymbol{\nu}$ that make $q(\mathbf{z}|\boldsymbol{\nu})$ as close as possible to the true posterior $p(\mathbf{z}|\mathbf{x})$



- Turns the inference problem into an **optimization** problem!
- Use $q(\mathbf{z}|\boldsymbol{\nu})$ with the fitted parameters as a proxy for the true posterior
  - To make predictions about future data
  - To investigate the posterior distribution of the hidden variables

Mean-field variational inference

- How should we choose a **tractable** family of distributions for $q(\mathbf{z})$?

Mean-field variational inference

- How should we choose a **tractable** family of distributions for $q(\mathbf{z})$?
  - Relax some constraints in the true distribution (e.g. dependencies)
  - For example, we can assume a **fully factorized** approximation

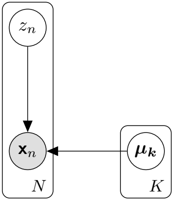
$$q(\mathbf{z}) = \prod_{m=1}^{M} q(z_m)$$

Mean-field variational inference

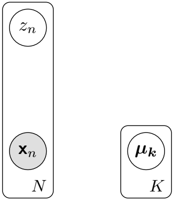
- How should we choose a **tractable** family of distributions for $q(\mathbf{z})$?
  - Relax some constraints in the true distribution (e.g. dependencies)
  - For example, we can assume a **fully factorized** approximation

$$q(\mathbf{z}) = \prod_{m=1}^{M} q(z_m)$$

  - This is called a **mean-field approximation**



True joint distribution



Fully factorized approximation

- We can also assume that the distribution factorizes in groups of latent variables

Kullback-Leibler (KL) divergence

- What criteria defines the closeness between the two distributions?
    - **Kullback-Leibler (KL) divergence**

$$\mathbb{KL}(q(\mathbf{z})||p(\mathbf{z}|\mathbf{x})) = \int_{\mathbf{z}} q(\mathbf{z}) \log \frac{q(\mathbf{z})}{p(\mathbf{z}|\mathbf{x})} = \mathbb{E}_q \left[\log \frac{q(\mathbf{z})}{p(\mathbf{z}|\mathbf{x})} \right]$$

Kullback-Leibler (KL) divergence

- What criteria defines the closeness between the two distributions?
  - **Kullback-Leibler (KL) divergence**

$$\mathbb{KL}(q(\mathbf{z})||p(\mathbf{z}|\mathbf{x})) = \int_{\mathbf{z}} q(\mathbf{z}) \log \frac{q(\mathbf{z})}{p(\mathbf{z}|\mathbf{x})} = \mathbb{E}_q\left[\log \frac{q(\mathbf{z})}{p(\mathbf{z}|\mathbf{x})}\right]$$

- Notice that the KL divergence is an asymmetric measure

$$\mathbb{KL}(q(\mathbf{z})||p(\mathbf{z}|\mathbf{x})) \neq \mathbb{KL}(p(\mathbf{z}|\mathbf{x})||q(\mathbf{z}))$$

Kullback-Leibler (KL) divergence

- What criteria defines the closeness between the two distributions?
    - **Kullback-Leibler (KL) divergence**

$$\mathbb{KL}(q(\mathbf{z})||p(\mathbf{z}|\mathbf{x})) = \int_{\mathbf{z}} q(\mathbf{z}) \log \frac{q(\mathbf{z})}{p(\mathbf{z}|\mathbf{x})} = \mathbb{E}_q\left[\log \frac{q(\mathbf{z})}{p(\mathbf{z}|\mathbf{x})}\right]$$

- Notice that the KL divergence is an asymmetric measure

$$\mathbb{KL}(q(\mathbf{z})||p(\mathbf{z}|\mathbf{x})) \neq \mathbb{KL}(p(\mathbf{z}|\mathbf{x})||q(\mathbf{z}))$$

- The reverse KL, $\mathbb{KL}(p(\mathbf{z}|\mathbf{x})||q(\mathbf{z}))$, gives rise to a different approximation inference algorithm called **expectation propagation (EP)**
    - It requires us to be able to take expectations with respect to $p(\mathbf{z}|\mathbf{x})$!
    - In general, it's more computationally expensive than variational inference
    - Watch `http://videolectures.net/mlss09uk_minka_ai/?q=minka`

Variational inference (VI) vs expectation propagation (EP)

- Different KL leads to different properties for the approximation
  - $\mathbb{KL}(p(\mathbf{z}|\mathbf{x})||q(\mathbf{z}))$ in EP is moment-matching
  - $\mathbb{KL}(q(\mathbf{z})||p(\mathbf{z}|\mathbf{x}))$ in VI is mode-seeking

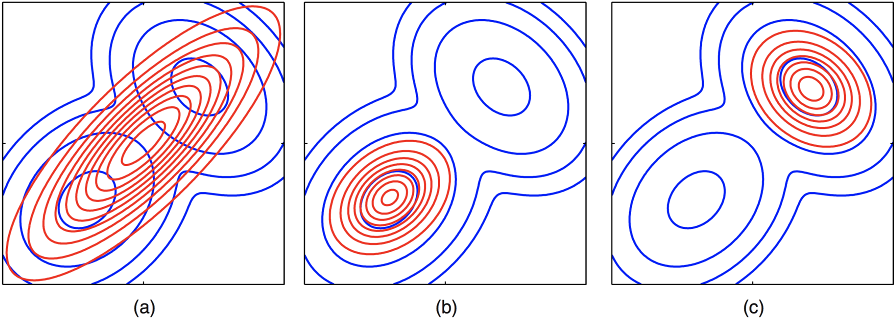


Figure: (a) $\mathbb{KL}(p||q)$. (b) $\mathbb{KL}(q||p)$. (c) Same as (b) but another local minimum. From Bishop (2006).

KL minimization

- Our goal is to find the variational parameters $\boldsymbol{\nu}^*$, such that

$$\boldsymbol{\nu}^* = \arg\min_{\boldsymbol{\nu}} \mathbb{KL}(q(\mathbf{z}|\boldsymbol{\nu})||p(\mathbf{z}|\mathbf{x})) = \arg\min_{\boldsymbol{\nu}} \int_{\mathbf{z}} q(\mathbf{z}|\boldsymbol{\nu}) \log \frac{q(\mathbf{z}|\boldsymbol{\nu})}{p(\mathbf{z}|\mathbf{x})}$$

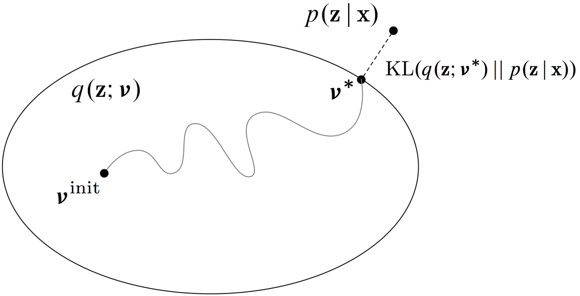


Figure: From David Blei (2017)

KL minimization

- Our goal is to find the variational parameters $\boldsymbol{\nu}^*$, such that

$$\boldsymbol{\nu}^* = \arg\min_{\boldsymbol{\nu}} \mathbb{KL}(q(\mathbf{z}|\boldsymbol{\nu})||p(\mathbf{z}|\mathbf{x})) = \arg\min_{\boldsymbol{\nu}} \int_{\mathbf{z}} q(\mathbf{z}|\boldsymbol{\nu}) \log \frac{q(\mathbf{z}|\boldsymbol{\nu})}{p(\mathbf{z}|\mathbf{x})}$$

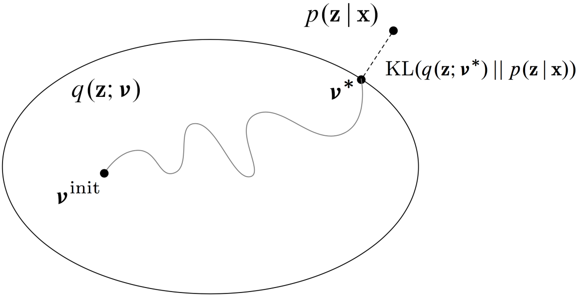


Figure: From David Blei (2017)

- Unfortunately, $\mathbb{KL}(q(\mathbf{z})||p(\mathbf{z}|\mathbf{x}))$ cannot be minimized directly!

KL minimization

- However, we can find a function that we can minimize, which is equal to $\mathbb{KL}(q(\mathbf{z})||p(\mathbf{z}|\mathbf{x}))$ up to an additive constant, as follows

$$
\begin{aligned}
\mathbb{KL}(q(\mathbf{z})||p(\mathbf{z}|\mathbf{x})) &= \mathbb{E}_q\left[\log \frac{q(\mathbf{z})}{p(\mathbf{z}|\mathbf{x})}\right] \\
&= \mathbb{E}_q[\log q(\mathbf{z})] - \mathbb{E}_q[\log p(\mathbf{z}|\mathbf{x})] \\
&= \mathbb{E}_q[\log q(\mathbf{z})] - \mathbb{E}_q\left[\log \frac{p(\mathbf{z},\mathbf{x})}{p(\mathbf{x})}\right] \\
&= -(\underbrace{\mathbb{E}_q[\log p(\mathbf{z},\mathbf{x})] - \mathbb{E}_q[\log q(\mathbf{z})]}_{\mathcal{L}(q)}) + \underbrace{\log p(\mathbf{x})}_{\text{const.}}
\end{aligned}
$$

KL minimization

- However, we can find a function that we can minimize, which is equal to $\mathbb{KL}(q(\mathbf{z})||p(\mathbf{z}|\mathbf{x}))$ up to an additive constant, as follows

$$
\begin{aligned}
\mathbb{KL}(q(\mathbf{z})||p(\mathbf{z}|\mathbf{x})) &= \mathbb{E}_q\left[\log \frac{q(\mathbf{z})}{p(\mathbf{z}|\mathbf{x})}\right] \\
&= \mathbb{E}_q[\log q(\mathbf{z})] - \mathbb{E}_q[\log p(\mathbf{z}|\mathbf{x})] \\
&= \mathbb{E}_q[\log q(\mathbf{z})] - \mathbb{E}_q\left[\log \frac{p(\mathbf{z},\mathbf{x})}{p(\mathbf{x})}\right] \\
&= -(\underbrace{\mathbb{E}_q[\log p(\mathbf{z},\mathbf{x})] - \mathbb{E}_q[\log q(\mathbf{z})]}_{\mathcal{L}(q)}) + \underbrace{\log p(\mathbf{x})}_{\text{const.}}
\end{aligned}
$$

- The $\log p(\mathbf{x})$ term does not depend on q and thus it can be ignored
- Minimizing the KL divergence is then equivalent to maximizing $\mathcal{L}(q)$
- $\mathcal{L}(q)$ is called the **evidence lower bound (ELBO)**

KL minimization

- The ELBO, $\mathcal{L}(q)$, is a **lower bound** on the log model evidence $\log p(\mathbf{x})$

$$\mathbb{KL}(q(\mathbf{z})||p(\mathbf{z}|\mathbf{x})) = -(\underbrace{\mathbb{E}_q[\log p(\mathbf{z},\mathbf{x})] - \mathbb{E}_q[\log q(\mathbf{z})]}_{\text{ELBO } \mathcal{L}(q)}) + \underbrace{\log p(\mathbf{x})}_{\text{log evidence}}$$

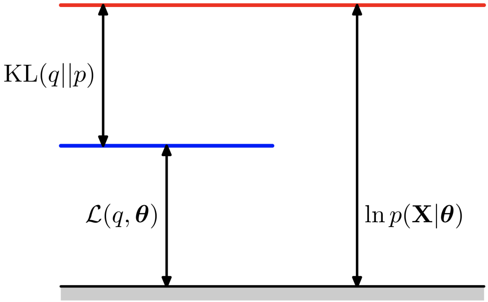


Figure: From Bishop (2006)

- The ELBO $\mathcal{L}(q)$ is tight when $q(\mathbf{z}) \approx p(\mathbf{z}|\mathbf{x})$, in which case $\mathcal{L}(q) \approx \log p(\mathbf{x})$

KL minimization

- The ELBO, $\mathcal{L}(q)$, is a **lower bound** on the log model evidence $\log p(\mathbf{x})$

$$\begin{aligned}
\log p(\mathbf{x}) &= \log \int_{\mathbf{z}} p(\mathbf{z}, \mathbf{x}) \\
&= \log \int_{\mathbf{z}} q(\mathbf{z}) \frac{p(\mathbf{z}, \mathbf{x})}{q(\mathbf{z})} \\
&= \log \mathbb{E}_q \left[\frac{p(\mathbf{z}, \mathbf{x})}{q(\mathbf{z})} \right] \\
&\geqslant \underbrace{\mathbb{E}_q[\log p(\mathbf{z}, \mathbf{x})] - \mathbb{E}_q[\log q(\mathbf{z})]}_{\mathcal{L}(q)}
\end{aligned}$$

- The last step comes from **Jensen's inequality**

$$\log \mathbb{E}[p(\mathbf{x})] \geqslant \mathbb{E}[\log p(\mathbf{x})]$$

  - This is a consequence of the concavity of the logarithmic function

Variational inference

- Our goal is to find the variational parameters $\boldsymbol{\nu}$ that maximize the ELBO $\mathcal{L}(q)$

$$\mathcal{L}(q) = \mathbb{E}_q[\log p(\mathbf{z}, \mathbf{x})] - \mathbb{E}_q[\log q(\mathbf{z}|\boldsymbol{\nu})]$$

Variational inference

- Our goal is to find the variational parameters $\boldsymbol{\nu}$ that maximize the ELBO $\mathcal{L}(q)$

$$\mathcal{L}(q) = \mathbb{E}_q[\log p(\mathbf{z}, \mathbf{x})] - \mathbb{E}_q[\log q(\mathbf{z}|\boldsymbol{\nu})]$$

- Can be maximized using a **coordinate ascent algorithm**
  - Iteratively optimize the variational parameters of each latent variable $q(z_m)$ in turn, holding the others fixed, until a convergence criteria is met
- Ensures convergence to a local maximum of $\mathcal{L}(q)$
- Remember, at convergence:

$$q(\mathbf{z}) \approx p(\mathbf{z}|\mathbf{x})$$

$$\mathcal{L}(q) \approx \log p(\mathbf{x})$$

Note

Variational inference (VI) is also often called **Variational Bayes (VB)**.

Two different perspectives on the ELBO $\mathcal{L}(q)$

- Expected value of log joint probability minus entropy terms

$$\mathcal{L}(q) = \mathbb{E}_q[\log p(\mathbf{z}, \mathbf{x})] - \underbrace{\mathbb{E}_q[\log q(\mathbf{z}|\boldsymbol{\nu})]}_{\text{entropy terms}}$$

  - Entropy terms favour less informative variational distributions $q(\mathbf{z}|\boldsymbol{\nu})$

Two different perspectives on the ELBO $\mathcal{L}(q)$

- Expected value of log joint probability minus entropy terms

$$\mathcal{L}(q) = \mathbb{E}_q[\log p(\mathbf{z}, \mathbf{x})] - \underbrace{\mathbb{E}_q[\log q(\mathbf{z}|\boldsymbol{\nu})]}_{\text{entropy terms}}$$

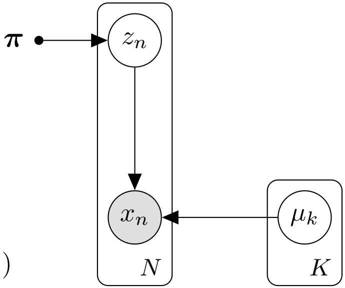
  - Entropy terms favour less informative variational distributions $q(\mathbf{z}|\boldsymbol{\nu})$

- Expected value of log likelihood minus KL divergence to prior

$$\begin{aligned}\mathcal{L}(q) &= \mathbb{E}_q[\log p(\mathbf{x}|\mathbf{z})] - \mathbb{E}_q[\log p(\mathbf{z})] - \mathbb{E}_q[\log q(\mathbf{z}|\boldsymbol{\nu})] \\ &= \mathbb{E}_q[\log p(\mathbf{x}|\mathbf{z})] - \underbrace{\mathbb{E}_q\left[\log \frac{q(\mathbf{z}|\boldsymbol{\nu})}{p(\mathbf{z})}\right]}_{\mathbb{KL}(q(\mathbf{z}|\boldsymbol{\nu})||p(\mathbf{z}))}\end{aligned}$$

  - $\mathbb{E}_q[\log p(\mathbf{x}|\mathbf{z})]$ is the reconstruction loss
  - $\mathbb{KL}(q(\mathbf{z}|\boldsymbol{\nu})||p(\mathbf{z}))$ acts like a regularization term (to stay close to the prior)

A practical example: Bayesian Gaussian mixture model



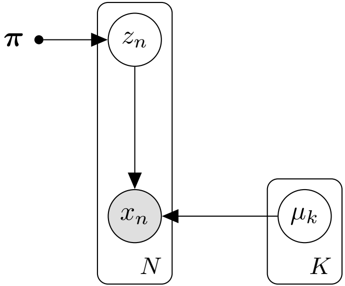
- For the sake of simplicity, assume $\boldsymbol{\pi}$ is given (fixed)

1. For each cluster k:
   a) Draw cluster center $\mu_k \sim \mathcal{N}(\mu_k|0, \tau^2)$
2. For each data point $1, \ldots, N$:
   a) Draw cluster assignment $z_n \sim \text{Multinomial}(z_n|\boldsymbol{\pi})$
   b) Draw observed data point $x_n \sim \mathcal{N}(x_n|\mu_{z_n}, \sigma^2)$

Note

For simplicity, we assumed the cluster variances σ^2 to be fixed.

A practical example: Bayesian Gaussian mixture model



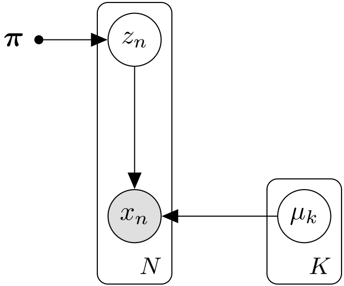
- Joint distribution:

$$p(\boldsymbol{\mu}, \mathbf{z}, \mathbf{x}) = \left(\prod_{k=1}^{K} p(\mu_k | 0, \tau^2) \right) \prod_{n=1}^{N} p(z_n | \boldsymbol{\pi}) \, p(x_n | z_n, \boldsymbol{\mu}, \sigma^2)$$

- Approximate (mean-field) distribution:

$$q(\boldsymbol{\mu}, \mathbf{z}) =$$

A practical example: Bayesian Gaussian mixture model



- Joint distribution:

$$p(\boldsymbol{\mu}, \mathbf{z}, \mathbf{x}) = \left(\prod_{k=1}^{K} p(\mu_k | 0, \tau^2) \right) \prod_{n=1}^{N} p(z_n | \boldsymbol{\pi}) \, p(x_n | z_n, \boldsymbol{\mu}, \sigma^2)$$

- Approximate (mean-field) distribution:

$$q(\boldsymbol{\mu}, \mathbf{z}) = \left(\prod_{k=1}^{K} q(\mu_k | \tilde{\mu}_k, \tilde{\sigma}_k) \right) \prod_{n=1}^{N} q(z_n | \boldsymbol{\phi}_n)$$

A practical example: Bayesian Gaussian mixture model

- Our goal is to find the variational parameters that maximize the ELBO $\mathcal{L}(q)$

$$\mathcal{L}(q) = \mathbb{E}_q[\log p(\boldsymbol{\mu}, \mathbf{z}, \mathbf{x})] - \mathbb{E}_q[\log q(\boldsymbol{\mu}, \mathbf{z})]$$

A practical example: Bayesian Gaussian mixture model

- Our goal is to find the variational parameters that maximize the ELBO $\mathcal{L}(q)$

$$\mathcal{L}(q) = \mathbb{E}_q[\log p(\boldsymbol{\mu}, \mathbf{z}, \mathbf{x})] - \mathbb{E}_q[\log q(\boldsymbol{\mu}, \mathbf{z})]$$

- Because q is fully factorized, the expectations in $\mathcal{L}(q)$ decompose into sums of simpler terms

$$\begin{aligned}\mathbb{E}_q[\log p(\boldsymbol{\mu}, \mathbf{z}, \mathbf{x})] = &\sum_{k=1}^{K} \mathbb{E}_q[\log p(\mu_k | 0, \tau^2)] + \sum_{n=1}^{N} \mathbb{E}_q[\log p(z_n | \boldsymbol{\pi})] \\ &+ \sum_{n=1}^{N} \mathbb{E}_q[\log p(x_n | z_n, \boldsymbol{\mu}, \sigma^2)]\end{aligned}$$

$$\mathbb{E}_q[\log q(\boldsymbol{\mu}, \mathbf{z})] = \sum_{k=1}^{K} \mathbb{E}_q[\log q(\mu_k | \tilde{\mu}_k, \tilde{\sigma}_k)] + \sum_{n=1}^{N} \mathbb{E}_q[\log q(z_n | \boldsymbol{\phi}_n)]$$

Expectations, expectations, expectations...

- For a discrete variable X:

$$\mathbb{E}[f(X)] = \sum_{X} p(X)\, f(X)$$

- For a continuous variable X:

$$\mathbb{E}[f(X)] = \int p(X)\, f(X)\, dX$$

- Some useful properties of expectations:

$$\begin{aligned}
\mathbb{E}[a] &= a \\
\mathbb{E}[a + bX] &= a + b\,\mathbb{E}[X] \\
\mathbb{E}[X + Y] &= \mathbb{E}[X] + \mathbb{E}[Y] \\
\mathbb{E}[XY] &= \mathbb{E}[X]\,\mathbb{E}[Y], \quad \text{only if } X \text{ and } Y \text{ are independent} \\
\mathbb{V}[X] &= \mathbb{E}[(X - \mathbb{E}[X])^2] = \mathbb{E}[X^2] - \mathbb{E}[X]^2
\end{aligned}$$

- More details on expectations:
`https://www3.nd.edu/~rwilliam/stats1/x12.pdf`

Expectations, expectations, expectations...

- Consider the approximate (mean-field) distribution:

$$q(\boldsymbol{\mu}, \mathbf{z}) = \left(\prod_{k=1}^{K} \mathcal{N}(\mu_k | \tilde{\mu}_k, \tilde{\sigma}_k) \right) \prod_{n=1}^{N} \text{Mult}(z_n | \boldsymbol{\phi}_n)$$

- Using the properties of expectations and the factorization of $q(\boldsymbol{\mu}, \mathbf{z})$:

$$\mathbb{E}_q[\mu_k] =$$

$$\mathbb{E}_q[\mu_k^2] =$$

$$\mathbb{E}_q[z_n] =$$

$$\mathbb{E}_q[z_{n,k}] =$$

$$\mathbb{E}_q[\mu_{z_n}] =$$

$$\mathbb{E}_q[\mu_{z_n}^2] =$$

Expectations, expectations, expectations...

- Consider the approximate (mean-field) distribution:

$$q(\boldsymbol{\mu}, \mathbf{z}) = \left(\prod_{k=1}^{K} \mathcal{N}(\mu_k | \tilde{\mu}_k, \tilde{\sigma}_k) \right) \prod_{n=1}^{N} \text{Mult}(z_n | \boldsymbol{\phi}_n)$$

- Using the properties of expectations and the factorization of $q(\boldsymbol{\mu}, \mathbf{z})$:

$$\mathbb{E}_q[\mu_k] = \tilde{\mu}_k$$

$$\mathbb{E}_q[\mu_k^2] =$$

$$\mathbb{E}_q[z_n] =$$

$$\mathbb{E}_q[z_{n,k}] =$$

$$\mathbb{E}_q[\mu_{z_n}] =$$

$$\mathbb{E}_q[\mu_{z_n}^2] =$$

Expectations, expectations, expectations...

- Consider the approximate (mean-field) distribution:

$$q(\boldsymbol{\mu}, \mathbf{z}) = \left(\prod_{k=1}^{K} \mathcal{N}(\mu_k | \tilde{\mu}_k, \tilde{\sigma}_k) \right) \prod_{n=1}^{N} \mathsf{Mult}(z_n | \boldsymbol{\phi}_n)$$

- Using the properties of expectations and the factorization of $q(\boldsymbol{\mu}, \mathbf{z})$:

$$\mathbb{E}_q[\mu_k] = \tilde{\mu}_k$$
$$\mathbb{E}_q[\mu_k^2] = \tilde{\mu}_k^2 + \tilde{\sigma}_k^2$$

$$\mathbb{E}_q[z_n] =$$

$$\mathbb{E}_q[z_{n,k}] =$$

$$\mathbb{E}_q[\mu_{z_n}] =$$

$$\mathbb{E}_q[\mu_{z_n}^2] =$$

Expectations, expectations, expectations...

- Consider the approximate (mean-field) distribution:

$$q(\boldsymbol{\mu}, \mathbf{z}) = \left(\prod_{k=1}^{K} \mathcal{N}(\mu_k | \tilde{\mu}_k, \tilde{\sigma}_k) \right) \prod_{n=1}^{N} \mathsf{Mult}(z_n | \boldsymbol{\phi}_n)$$

- Using the properties of expectations and the factorization of $q(\boldsymbol{\mu}, \mathbf{z})$:

$$\mathbb{E}_q[\mu_k] = \tilde{\mu}_k$$

$$\mathbb{E}_q[\mu_k^2] = \tilde{\mu}_k^2 + \tilde{\sigma}_k^2$$

$$\mathbb{E}_q[z_n] = \sum_{k=1}^{K} k\, p(z_n = k) = \sum_{k=1}^{K} k\, \phi_{n,k}$$

$$\mathbb{E}_q[z_{n,k}] =$$

$$\mathbb{E}_q[\mu_{z_n}] =$$

$$\mathbb{E}_q[\mu_{z_n}^2] =$$

Expectations, expectations, expectations...

- Consider the approximate (mean-field) distribution:

$$q(\boldsymbol{\mu}, \mathbf{z}) = \left(\prod_{k=1}^{K} \mathcal{N}(\mu_k | \tilde{\mu}_k, \tilde{\sigma}_k) \right) \prod_{n=1}^{N} \mathsf{Mult}(z_n | \boldsymbol{\phi}_n)$$

- Using the properties of expectations and the factorization of $q(\boldsymbol{\mu}, \mathbf{z})$:

$$\begin{aligned}
\mathbb{E}_q[\mu_k] &= \tilde{\mu}_k \\
\mathbb{E}_q[\mu_k^2] &= \tilde{\mu}_k^2 + \tilde{\sigma}_k^2 \\
\mathbb{E}_q[z_n] &= \sum_{k=1}^{K} k \, p(z_n = k) = \sum_{k=1}^{K} k \, \phi_{n,k} \\
\mathbb{E}_q[z_{n,k}] &= p(z_n = k) = \phi_{n,k} \\
\mathbb{E}_q[\mu_{z_n}] &= \\
\mathbb{E}_q[\mu_{z_n}^2] &=
\end{aligned}$$

Expectations, expectations, expectations...

- Consider the approximate (mean-field) distribution:

$$q(\boldsymbol{\mu}, \mathbf{z}) = \left(\prod_{k=1}^{K} \mathcal{N}(\mu_k | \tilde{\mu}_k, \tilde{\sigma}_k) \right) \prod_{n=1}^{N} \mathsf{Mult}(z_n | \boldsymbol{\phi}_n)$$

- Using the properties of expectations and the factorization of $q(\boldsymbol{\mu}, \mathbf{z})$:

$$\begin{aligned}
\mathbb{E}_q[\mu_k] &= \tilde{\mu}_k \\
\mathbb{E}_q[\mu_k^2] &= \tilde{\mu}_k^2 + \tilde{\sigma}_k^2 \\
\mathbb{E}_q[z_n] &= \sum_{k=1}^{K} k\, p(z_n = k) = \sum_{k=1}^{K} k\, \phi_{n,k} \\
\mathbb{E}_q[z_{n,k}] &= p(z_n = k) = \phi_{n,k} \\
\mathbb{E}_q[\mu_{z_n}] &= \sum_{k=1}^{K} \mathbb{E}_q[z_{n,k}]\, \mathbb{E}_q[\mu_k] = \sum_{k=1}^{K} \phi_{n,k}\, \tilde{\mu}_k \\
\mathbb{E}_q[\mu_{z_n}^2] &=
\end{aligned}$$

Expectations, expectations, expectations...

- Consider the approximate (mean-field) distribution:

$$q(\boldsymbol{\mu}, \mathbf{z}) = \left(\prod_{k=1}^{K} \mathcal{N}(\mu_k | \tilde{\mu}_k, \tilde{\sigma}_k) \right) \prod_{n=1}^{N} \mathsf{Mult}(z_n | \boldsymbol{\phi}_n)$$

- Using the properties of expectations and the factorization of $q(\boldsymbol{\mu}, \mathbf{z})$:

$$\begin{aligned}
\mathbb{E}_q[\mu_k] &= \tilde{\mu}_k \\
\mathbb{E}_q[\mu_k^2] &= \tilde{\mu}_k^2 + \tilde{\sigma}_k^2 \\
\mathbb{E}_q[z_n] &= \sum_{k=1}^{K} k\, p(z_n = k) = \sum_{k=1}^{K} k\, \phi_{n,k} \\
\mathbb{E}_q[z_{n,k}] &= p(z_n = k) = \phi_{n,k} \\
\mathbb{E}_q[\mu_{z_n}] &= \sum_{k=1}^{K} \mathbb{E}_q[z_{n,k}]\, \mathbb{E}_q[\mu_k] = \sum_{k=1}^{K} \phi_{n,k}\, \tilde{\mu}_k \\
\mathbb{E}_q[\mu_{z_n}^2] &= \sum_{k=1}^{K} \mathbb{E}_q[z_{n,k}]\, \mathbb{E}_q[\mu_k^2] = \sum_{k=1}^{K} \phi_{n,k}\, (\tilde{\mu}_k^2 + \tilde{\sigma}_k^2)
\end{aligned}$$

Playtime!

- **Calculate** the coordinate ascent updates for maximizing $\mathcal{L}(q)$

$$\mathcal{L}(q) = \mathbb{E}_q[\log p(\boldsymbol{\mu}, \mathbf{z}, \mathbf{x})] - \mathbb{E}_q[\log q(\boldsymbol{\mu}, \mathbf{z})]$$

- Hints:
  - Consider each variational parameter in turn: $\tilde{\mu}_k$, $\tilde{\sigma}_k$ and $\phi_{n,k}$

Table: Variables in the model and corresponding variational parameters

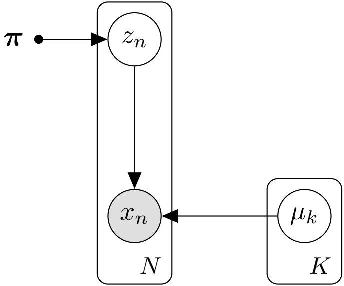
Model variable	μ_k		z_n
Variational parameter	$\tilde{\mu}_k$	$\tilde{\sigma}_k$	$\boldsymbol{\phi}_n$

  - For each variational parameter, keep the relevant terms in $\mathcal{L}(q)$
  - Take derivative with respect to the variational parameter and set it to zero
  - If necessary, use Lagrange multipliers to ensure constraints (e.g. $\sum_{k=1}^{K} \phi_{n,k} = 1$)

Note

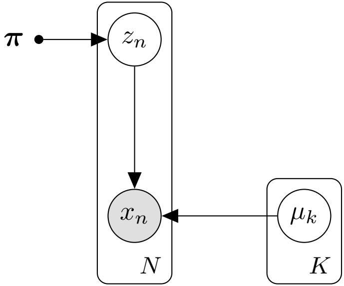
Automatic differentiation variational inference (ADVI) in STAN attempts to do all of this automatically (sometimes successfully, other times not so much).

Optimizing hyper-parameters



- What if we wanted to optimize the hyper-parameters $\boldsymbol{\pi}$ too?

Optimizing hyper-parameters



- What if we wanted to optimize the hyper-parameters $\boldsymbol{\pi}$ too?
- ELBO $\mathcal{L}(q)$ is a lower bound on the log marginal likelihood of the data $\log p(\mathbf{x})$

$$\log p(\mathbf{x}) = \log \int_{\boldsymbol{\mu}} \left(\prod_{k=1}^{K} p(\mu_k) \right) \prod_{n=1}^{N} \sum_{z_n} p(z_n|\boldsymbol{\pi})\, p(x_n|z_n, \boldsymbol{\mu}) \geq \mathcal{L}(q)$$

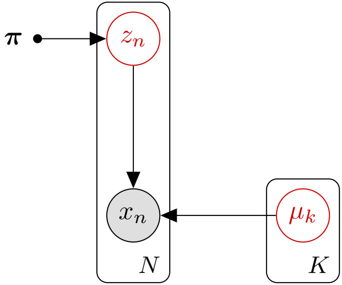
- At convergence of VI, this bound is thigh: $\mathcal{L}(q) \approx \log p(\mathbf{x})$
- We can use $\mathcal{L}(q)$ to find a maximum likelihood estimate of $\boldsymbol{\pi}$!

Variational Bayesian Expectation-Maximization (VB-EM)

- Goal is to find posterior over latent variables $\mathbf{z}$ using VI and maximum likelihood estimates for hyper-parameters $\boldsymbol{\theta}$

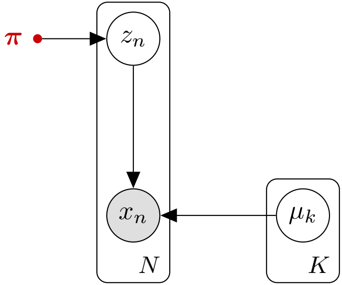
Variational Bayesian Expectation-Maximization (VB-EM)

- Goal is to find posterior over latent variables $\mathbf{z}$ using VI and maximum likelihood estimates for hyper-parameters $\boldsymbol{\theta}$
- VB-EM alternates between these 2 steps until convergence:
  - E-step: run VI to find approximate posterior $q(\mathbf{z})$, such that $q(\mathbf{z}) \approx p(\mathbf{z}|\mathbf{x})$
  - M-step: use $\mathcal{L}(q) \approx \log p(\mathbf{x})$ to find maximum likelihood estimates of $\boldsymbol{\theta}$



Variational Bayesian Expectation-Maximization (VB-EM)

- Goal is to find posterior over latent variables $\mathbf{z}$ using VI and maximum likelihood estimates for hyper-parameters $\boldsymbol{\theta}$
- VB-EM alternates between these 2 steps until convergence:
  - E-step: run VI to find approximate posterior $q(\mathbf{z})$, such that $q(\mathbf{z}) \approx p(\mathbf{z}|\mathbf{x})$
  - M-step: use $\mathcal{L}(q) \approx \log p(\mathbf{x})$ to find maximum likelihood estimates of $\boldsymbol{\theta}$



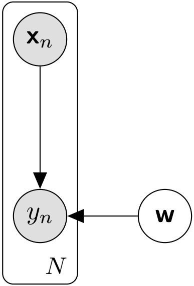
Relation with standard Expectation-Maximization (EM)

- Goal is to find posterior over latent variables $\mathbf{z}$ using VI and maximum likelihood estimates for hyper-parameters $\boldsymbol{\theta}$
- VB-EM alternates between these 2 steps until convergence:
  - E-step: run VI to find approximate posterior $q(\mathbf{z})$, such that $q(\mathbf{z}) \approx p(\mathbf{z}|\mathbf{x})$
  - M-step: use $\mathcal{L}(q) \approx \log p(\mathbf{x})$ to find maximum likelihood estimates of $\boldsymbol{\theta}$

Note

If you are familiar with the popular Expectation-Maximization (EM) algorithm, it is a special case of VB-EM, when the E-step is exact (exact inference instead of VI).

VI example: Automatic Relevance Determination (ARD)

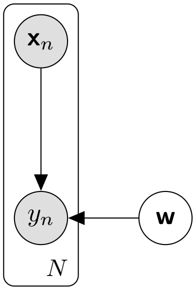


- Recall the Bayesian linear regression model:

1. Draw weights $\mathbf{w} \sim \mathcal{N}(\mathbf{w}|\mathbf{0}, \alpha^{-1}\mathbf{I})$
2. For each data point $n \in \{1, \ldots, N\}$
   a. Draw target $y_n \sim \mathcal{N}(y_n|\mathbf{w}^\mathsf{T}\mathbf{x}_n, \sigma^2)$

- It assumes independent priors on the weights $\mathbf{w}$ with the same variance (α^{-1}) for all input dimensions $d \in \{1, \ldots, D\}$

VI example: Automatic Relevance Determination (ARD)



- Recall the Bayesian linear regression model:

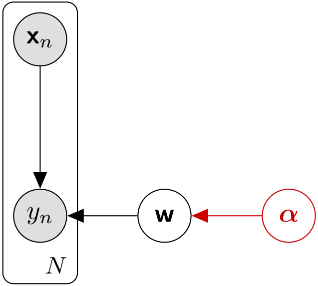
1. Draw weights $\mathbf{w} \sim \mathcal{N}(\mathbf{w}|\mathbf{0}, \alpha^{-1}\mathbf{I})$
2. For each data point $n \in \{1, \ldots, N\}$
   a. Draw target $y_n \sim \mathcal{N}(y_n|\mathbf{w}^\mathsf{T}\mathbf{x}_n, \sigma^2)$

- It assumes independent priors on the weights $\mathbf{w}$ with the same variance (α^{-1}) for all input dimensions $d \in \{1, \ldots, D\}$
- Alternatively, we consider a different variance α_d^{-1} for each input dimension
- We can then specify a prior on the precisions $\alpha_d \sim \text{Gamma}(\alpha_d|a_0, b_0)$
- This is called an **hyper-prior**!

VI example: Automatic Relevance Determination (ARD)

- New generative process and PGM:

1. For each input dimension $d \in \{1, \ldots, D\}$
   a. Draw weight precision $\alpha_d \sim \text{Gamma}(\alpha_d | a_0, b_0)$
2. Draw weights $\mathbf{w} \sim \mathcal{N}(\mathbf{w}|\mathbf{0}, \boldsymbol{\alpha}^{-1}\mathbf{I})$
3. For each data point $n \in \{1, \ldots, N\}$
   a. Draw target $y_n \sim \mathcal{N}(y_n|\mathbf{w}^\mathsf{T}\mathbf{x}_n, \sigma^2)$

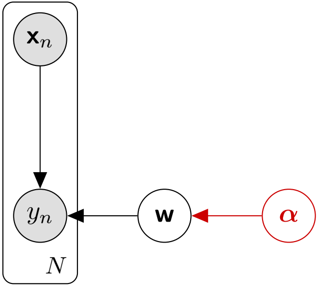


- Prior on α_d specifies how uncertain we are a-priori that these weights are small

VI example: Automatic Relevance Determination (ARD)

- New generative process and PGM:

1. For each input dimension $d \in \{1, \ldots, D\}$
   a. Draw weight precision $\alpha_d \sim \text{Gamma}(\alpha_d | a_0, b_0)$
2. Draw weights $\mathbf{w} \sim \mathcal{N}(\mathbf{w}|\mathbf{0}, \boldsymbol{\alpha}^{-1}\mathbf{I})$
3. For each data point $n \in \{1, \ldots, N\}$
   a. Draw target $y_n \sim \mathcal{N}(y_n|\mathbf{w}^\mathsf{T}\mathbf{x}_n, \sigma^2)$



- Prior on α_d specifies how uncertain we are a-priori that these weights are small
- Posterior on α_d tells us how relevant the d^{th} dimension is - hence the name ARD!
- Added flexibility allows some weights w_d to be further pushed towards zero
- It allows to do **automatic feature selection**!

VI example: Automatic Relevance Determination (ARD)

- The joint distribution is given by:

$$p(\mathbf{y}, \mathbf{w}, \boldsymbol{\alpha}|\mathbf{X}) = \mathcal{N}(\mathbf{w}|\mathbf{0}, \boldsymbol{\alpha}^{-1}\mathbf{I}) \left(\prod_{j=1}^{D} \mathrm{Ga}(\alpha_j|a_0, b_0) \right) \prod_{n=1}^{N} \mathcal{N}(y_n|\mathbf{w}^T \mathbf{x}_n, \sigma^2)$$

where $\mathbf{y} = \{y_1, \ldots, y_N\}$ and $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$

VI example: Automatic Relevance Determination (ARD)

- The joint distribution is given by:

$$p(\mathbf{y}, \mathbf{w}, \boldsymbol{\alpha}|\mathbf{X}) = \mathcal{N}(\mathbf{w}|\mathbf{0}, \boldsymbol{\alpha}^{-1}\mathbf{I}) \left(\prod_{j=1}^{D} \mathrm{Ga}(\alpha_j|a_0, b_0) \right) \prod_{n=1}^{N} \mathcal{N}(y_n|\mathbf{w}^T\mathbf{x}_n, \sigma^2)$$

  where $\mathbf{y} = \{y_1, \ldots, y_N\}$ and $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$

- Exact inference is unfortunately now intractable

$$p(\mathbf{w}, \boldsymbol{\alpha}|\mathbf{y}, \mathbf{X}) = \frac{p(\mathbf{y}, \mathbf{w}, \boldsymbol{\alpha}|\mathbf{X})}{\int_{\mathbf{w}} \int_{\boldsymbol{\alpha}} p(\mathbf{y}, \mathbf{w}, \boldsymbol{\alpha}|\mathbf{X})}$$

VI example: Automatic Relevance Determination (ARD)

- The joint distribution is given by:

$$p(\mathbf{y}, \mathbf{w}, \boldsymbol{\alpha}|\mathbf{X}) = \mathcal{N}(\mathbf{w}|\mathbf{0}, \boldsymbol{\alpha}^{-1}\mathbf{I}) \left(\prod_{j=1}^{D} \text{Ga}(\alpha_j|a_0, b_0) \right) \prod_{n=1}^{N} \mathcal{N}(y_n|\mathbf{w}^T\mathbf{x}_n, \sigma^2)$$

  where $\mathbf{y} = \{y_1, \dots, y_N\}$ and $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$

- Exact inference is unfortunately now intractable

$$p(\mathbf{w}, \boldsymbol{\alpha}|\mathbf{y}, \mathbf{X}) = \frac{p(\mathbf{y}, \mathbf{w}, \boldsymbol{\alpha}|\mathbf{X})}{\int_{\mathbf{w}} \int_{\boldsymbol{\alpha}} p(\mathbf{y}, \mathbf{w}, \boldsymbol{\alpha}|\mathbf{X})}$$

- But we can use (mean-field) variational inference!

$$q(\mathbf{w}, \boldsymbol{\alpha}) = \mathcal{N}(\mathbf{w}|\mathbf{m}_N, \mathbf{V}_N) \prod_{j=1}^{D} \text{Ga}(\alpha_j|a_{Nj}, b_{Nj})$$

- Our goal is to find the variational parameters $\mathbf{m}_N$, $\mathbf{V}_N$, a_{Nj} and b_{Nj} that maximize the ELBO $\mathcal{L}(q)$:

$$\mathcal{L}(q) = \mathbb{E}_q[\log p(\mathbf{y}, \mathbf{w}, \boldsymbol{\alpha}|\mathbf{X})] - \mathbb{E}_q[\log q(\mathbf{w}, \boldsymbol{\alpha})]$$

Playtime!

- **Implement** the coordinate ascent updates for maximizing $\mathcal{L}(q)$

$$\mathcal{L}(q) = \mathbb{E}_q[\log p(\mathbf{y}, \mathbf{w}, \boldsymbol{\alpha}|\mathbf{X})] - \mathbb{E}_q[\log q(\mathbf{w}, \boldsymbol{\alpha})]$$

- Jupyter notebook: "12 - Variational inference - ARD.ipynb"
- Hints:
  - Consider each variational parameter in turn: $\mathbf{m}_N$, $\mathbf{V}_N$, a_{Nj} and b_{Nj}

Table: Variables in the model and corresponding variational parameters

Model variable	$\mathbf{w}$		α_j	
Variational parameter	$\mathbf{m}_N$	$\mathbf{V}_N$	a_{Nj}	b_{Nj}

Note

Notice how fast this VI algorithm is when compared to MCMC inference in STAN!