Machine Learning Notes

Yuanxun Zhang

February 14, 2019

Contents

Chapter 1

Probability Theory

1.1 Dirichlet Distribution

Dirichlet distribution, often denoted $\mathbf{Dir}(\boldsymbol{\alpha})$ or $\mathbf{Dir}(\alpha_1, ..., \alpha_K)$, is a family of continuous multivariate probability distributions parameterized by a vector $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_K)$, where $K \geqslant 2$. It is a multivariate generalization of the Beta distribution, and it's equal to Beta distribution at $K = 2$.

Dirichlet distribution is a density of $\theta = (\theta_1, ..., \theta_K)$ in K dimensions, and $\sum_{i=1}^{K} \theta_i = 1$. So, Dirichlet distribution is commonly used to sample Multinomial distribution. Its probability density function is,

$$p(\theta|\boldsymbol{\alpha}) = \frac{\Gamma(\sum_{i=1}^{K} \alpha_i)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \prod_{i=1}^{K} \theta_i^{\alpha_i - 1} \tag{1.1}$$

Or, it can be written as,

$$p(\theta|\boldsymbol{\alpha}) = \frac{1}{B(\boldsymbol{a})} \prod_{i=1}^{K} \theta_i^{\alpha_i - 1}, \qquad B(\alpha) = \frac{\prod_{i=1}^{K} \Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^{K} \alpha_i)} \quad \text{(Beta function)} \tag{1.2}$$

For example, we draw a Multinomial distribution θ from from a Dirichlet with parameters $\boldsymbol{\alpha}$ and then then sample a sequence of N discrete variables $x_1, x_2, ..., x_N$. Then, the probability of $\boldsymbol{x}$ given θ is $\prod_{i=1}^{K} \theta_i^{n_i}$. Combining with Equation 1.1, we have

$$\begin{aligned} p(\boldsymbol{x}, \theta|\boldsymbol{\alpha}) &= p(\theta|\boldsymbol{\alpha}) p(\boldsymbol{x}|\theta) \\ &= \frac{\Gamma(\sum_{i=1}^{K} \alpha_i)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \prod_{i=1}^{K} \theta_i^{\alpha_i - 1} \times \prod_{i=1}^{K} \theta_i^{n_i} \\ &= \frac{\Gamma(\sum_{i=1}^{K} \alpha_i)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \prod_{i=1}^{K} \theta_i^{n_i + \alpha_i - 1} \end{aligned} \tag{1.3}$$

In Equation 1.3, we can integrate out θ to get the marginal probability $p(\boldsymbol{x}|\boldsymbol{\alpha})$, hence

$$\begin{aligned} p(\boldsymbol{x}|\boldsymbol{\alpha}) &= \int p(\boldsymbol{x}, \theta|\boldsymbol{\alpha}) d\theta \\ &= \int \frac{\Gamma(\sum_{i=1}^{K} \alpha_i)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \prod_{i=1}^{K} \theta_i^{n_i + \alpha_i - 1} d\theta \end{aligned} \tag{1.4}$$

Because the integral of distribution is equal to 1, then applying integral of Dirichlet distribution in Equation 1.1, we have

$$\int \frac{\Gamma(\sum_{i=1}^{K} \alpha_i)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \prod_{i=1}^{K} \theta_i^{\alpha_i - 1} = 1 \tag{1.5}$$

Moving the terms not depending on θ outside the integral, we get

$$\frac{\Gamma(\sum_{i=1}^{K} \alpha_i)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \int \prod_{i=1}^{K} \theta_i^{\alpha_i - 1} = 1 \tag{1.6}$$

Then,

$$\int \prod_{i=1}^{K} \theta_i^{\alpha_i - 1} = \frac{\prod_{i=1}^{K} \Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^{K} \alpha_i)} \tag{1.7}$$

Applying Equation 1.7 to Equation 1.4, we have

$$\begin{aligned} p(\boldsymbol{x}|\boldsymbol{\alpha}) &= \int p(\boldsymbol{x}, \theta|\boldsymbol{\alpha}) d\theta && (1.8) \\ &= \int \frac{\Gamma(\sum_{i=1}^{K} \alpha_i)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \prod_{i=1}^{K} \theta_i^{n_i + \alpha_i - 1} d\theta \\ &= \frac{\Gamma(\sum_{i=1}^{K} \alpha_i)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \int \prod_{i=1}^{K} \theta_i^{n_i + \alpha_i - 1} d\theta \\ &= \frac{\Gamma(\sum_{i=1}^{K} \alpha_i)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \frac{\prod_{i=1}^{K} \Gamma(\alpha_i + n_i)}{\Gamma(\sum_{i=1}^{K} (\alpha_i + n_i))} && (1.9) \end{aligned}$$

Chapter 2

Variational Inference

2.1 Problem Definition

The goal of variational inference is to approximate a conditional density of latent variables given observed variables. The key idea is to solve this problem with optimization. We use a family of densities over the latent variables, parameterized by free "variational parameters".

Let $\mathbf{x} = \{x_1, x_2, ..., x_n\}$ be a set of observed variables, $\mathbf{z} = \{z_1, z_2, ..., z_m\}$ be a set of latent (or hidden) variables, and θ be the model parameter. In Bayesian inference, we are usually interested in the posterior of latent variables $\mathbf{z}$ given $\mathbf{x}$ and θ, which is $p(\mathbf{z}|\mathbf{x}, \theta)$. According to Bayes theory, the computation of the posterior is

$$p(\mathbf{z}|\mathbf{x}, \theta) = \frac{p(\mathbf{z}, \mathbf{x}|\theta)}{p(\mathbf{x}|\theta)} = \frac{p(\mathbf{z}, \mathbf{x}|\theta)}{\int p(\mathbf{z}, \mathbf{x}|\theta) d\mathbf{z}}$$

However, the integral of marginal distribution of the observations $p(\mathbf{x}|\theta) = \int p(\mathbf{z}, \mathbf{x}|\theta) d\mathbf{z}$ is usually computationally intractable. The $p(\mathbf{x}|\theta)$ are also called *evidence*.

2.2 The evidence lower bound

The $p(\mathbf{x}|\theta)$ is usually computationally intractable, and according to **Jensens inequality**,

$$\begin{aligned}
\log p(\mathbf{x}|\theta) &= \log \int p(\mathbf{z}, \mathbf{x}|\theta) d\mathbf{z} \\
&= \log \int q(\mathbf{z}) \frac{p(\mathbf{z}, \mathbf{x}|\theta)}{q(\mathbf{z})} d\mathbf{z} \\
&\geqslant \int q(\mathbf{z}) \log \frac{p(\mathbf{z}, \mathbf{x}|\theta)}{q(\mathbf{z})} d\mathbf{z} \quad \text{(using Jensen's inequality)} \\
&= \int q(\mathbf{z}) \log p(\mathbf{z}, \mathbf{x}|\theta) d\mathbf{z} - \int q(\mathbf{z}) \log q(\mathbf{z}) d\mathbf{z} \\
&= \mathbb{E}_q[\log p(\mathbf{z}, \mathbf{x}|\theta)] - \mathbb{E}_q[\log q(\mathbf{z})] = \mathcal{L}(q) \qquad (2.1)
\end{aligned}$$

Hence, $\log(p(\mathbf{x}|\theta)) \geqslant \mathcal{L}(q)$, and $\mathcal{L}(q)$ is called **evidence lower bound (or ELBO)** of $\log p(\mathbf{x}|\theta)$. In addition, the second term $\mathbb{E}_q[\log q(\mathbf{z})]$ is entropy. The $\mathcal{L}(q)$ can be also

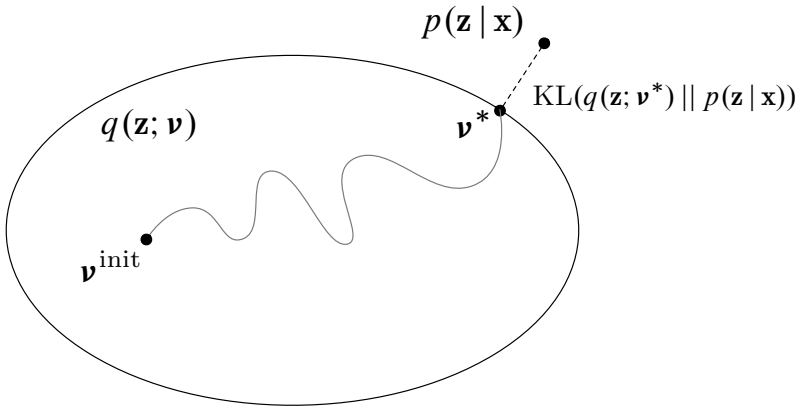

Figure 2.1: Variational Inference is solved through optimization, the $\mathbf{v}$ denotes the variational parameters.

written as $\mathrm{ELBO}(q)$ as,

$$\mathrm{ELBO}(q) = \mathbb{E}_q[\log p(\mathbf{z}, \mathbf{x}|\theta)] - \mathbb{E}_q[\log q(\mathbf{z})] \tag{2.2}$$

Recall **Kullback-Leibler (KL) divergence** as,

$$\begin{aligned}
\mathrm{KL}(q||p) &= \mathbb{E}\Big[\log \frac{q(\mathbf{z})}{p(\mathbf{z}|\mathbf{x}, \theta)}\Big] \\
&= \mathbb{E}[\log q(\mathbf{z})] - \mathbb{E}[\log p(\mathbf{z}|\mathbf{x}, \theta)] \\
&= \mathbb{E}[\log q(\mathbf{z})] - \mathbb{E}\Big[\log \frac{p(\mathbf{z}, \mathbf{x}|\theta)}{p(\mathbf{x}|\theta)}\Big] \\
&= \mathbb{E}[\log q(\mathbf{z})] - \mathbb{E}[\log p(\mathbf{z}, \mathbf{x}|\theta)] + \log p(x|\theta) \\
&= -(\mathbb{E}[\log p(\mathbf{z}, \mathbf{x}|\theta)] - \mathbb{E}[\log q(\mathbf{z})]) + \log p(x|\theta)
\end{aligned} \tag{2.3}$$

Where all expectations are taken with respect to $q(z)$. Combined with Equation(2.1), and the Equation(2.3) can be written as,

$$\mathrm{KL}(q||p) = -\mathcal{L}(q) + \log p(x|\theta)$$

Hence, the $\log p(x|\theta)$ can be decomposed as,

$$\begin{aligned}
\log p(\mathbf{x}|\theta) &= \mathcal{L}(q) + \mathrm{KL}(q||p) \\
\text{or, } \log p(\mathbf{x}|\theta) &= \mathrm{ELBO}(q) + \mathrm{KL}(q||p)
\end{aligned} \tag{2.4}$$

The goal of variational inference is to find the best candidate, the one closest in KL divergence to the exact conditional. In Figure 2.1, Inference now amounts to solving the following optimization problem,

$$q^*(\mathbf{z}) = \arg\min \mathbf{KL}(q(\mathbf{z})||q(\mathbf{z}|\mathbf{x}, \theta)) \tag{2.5}$$

With Equation 2.4, we can also infer that minimizing the KL divergence is equivalent to maximizing the ELBO.

2.3 The mean-field variational family

We now describe a variational family, to complete the specification of the optimization problem. The complexity of the family determines the complexity of the optimization; it is more difficult to optimize over a complex family than a simple family.

Hence, we focus on the mean-field variational family, where the latent variables are mutually *independent*, and each governed by a distinct factor in the variational density. A generic member of the mean-field variational family is defined as,

$$q(\mathbf{z}) = \prod_{j=1}^{m} q_j(z_j) \tag{2.6}$$

Each latent variable z_j is governed by its own variational factor, the density $q_j(z_j)$. In optimization, these variational factors are chosen to maximize the ELBO of Equation.

2.4 Coordinate ascent mean-field variational inference

Using the ELBO and the mean-field family, we have cast the approximate conditional inference as an optimization problem. To solve this optimization problem, we introduce the **coordinate ascent variational inference (CAVI)** method. CAVI iteratively optimizes each factor of the mean-field variational density, while holding the other fixed. It climbs the ELBO to a local optimum.

In the Equation 2.2, the $\mathrm{ELBO}(q)$ or $\mathcal{L}(q)$ is defined as,

$$\mathrm{ELBO}(q) = \mathbb{E}_q[\log p(\mathbf{z}, \mathbf{x}|\theta)] - \mathbb{E}_q[\log q(\mathbf{z})] \tag{2.7}$$

In oder to simplify the equation, we ignore the model parameters θ, which is written as,

$$\mathrm{ELBO}(q) = \mathbb{E}_q[\log p(\mathbf{z}, \mathbf{x})] - \mathbb{E}_q[\log q(\mathbf{z})] \tag{2.8}$$

According to chain rule we have,

$$p(z_{1:m}, x_{1:n}) = p(x_{1:n}) \prod_{j=1}^{m} p(z_j|z_{1:(j-1)}, x_{1:n}) \tag{2.9}$$

And,

$$\mathbb{E}_q[\log q(z_{1:m})] = \sum_{j=1}^{m} \mathbb{E}_{q_j}[\log p(z_j)] \tag{2.10}$$

Hence, combining Equation 2.8, 2.9, and , we got,

$$\begin{aligned}
\text{ELBO}(q) &= \mathbb{E}_q[\log\left(p(x_{1:n})\prod_{j=1}^{m} p(z_j|z_{1:(j-1)}, x_{1:n})\right)] - \mathbb{E}_q[\log q(\mathbf{z})] \\
&= \mathbb{E}_q\left[\log p(x_{1:n}) + \sum_{j=1}^{m} \log p(z_j|z_{1:(j-1)}, x_{1:n})\right] - \mathbb{E}_q[\log q(\mathbf{z})] \\
&= \mathbb{E}_q[\log p(x_{1:n})] + \sum_{j=1}^{m}\left[\mathbb{E}_q[\log p(z_j|z_{1:(j-1)}, x_{1:n})] - \mathbb{E}_{q_j}[\log q(z_j)]\right] \\
&= \log p(x_{1:n}) + \sum_{j=1}^{m}\left[\mathbb{E}_q[\log p(z_j|z_{1:(j-1)}, x_{1:n})] - \mathbb{E}_{q_j}[\log q(z_j)]\right] \qquad (2.11)
\end{aligned}$$

We can consider the $\mathbb{E}_q[\log p(z_j|z_{1:(j-1)}, x_{1:n})]$ as,

$$\mathbb{E}_q[\log p(z_j|z_{1:(j-1)}, x_{1:n})] = \int_{\mathbf{z}} \prod_{i=1}^{m} q_i(z_i) \log p(z_j|z_{1:(j-1)}, \mathbf{x}) d\mathbf{z} \qquad (2.12)$$

Hence, the ELBO can be decomposed based on different $q(z_j)$, then we can remove the components that do not depend on $q(z_j)$. Hence, supposing the chain rule with the variable z_j as the last variable in the list, we can write the objective function,

$$\mathcal{L}_j = \mathbb{E}_q[\log p(z_j|z_{-j}, \mathbf{x})] - \mathbb{E}_{q_j}[\log q(z_j)] + \text{const} \qquad (2.13)$$

Next, because we only consider $q(z_j)$, we can rewrite the objective function of $q(z_j)$,

$$\begin{aligned}
\mathcal{L}_j &= \int q(z_j)\mathbb{E}_{-j}[\log p(z_j|z_{-j}, \mathbf{x})]dz_j - \int q(z_j)\log q(z_j)dz_j \qquad (2.14) \\
&= q(z_j)\mathbb{E}_{-j}[\log p(z_j|z_{-j}, \mathbf{x})] - q(z_j)\log q(z_j) \qquad (2.15)
\end{aligned}$$

Here, the notation $\mathbb{E}_{-j}[\cdot]$ denotes an expectation with respect to q distributions over all variables $\mathbf{z}$ expect for the variable z_j.

So, to **maximize** the ELBO ($\mathcal{L}_j$), we take the derivative of $\mathcal{L}_j$ with respect to $q(z_j)$ and set the derivative to zero,

$$\frac{d\mathcal{L}_j}{dq(z_j)} = \mathbb{E}_{-j}[\log p(z_j|z_{-j}, \mathbf{x})] - \log q(z_j) - 1 = 0 \qquad (2.16)$$

And this leads to our CAVI updates for $q^*(z_j)$,

$$q^*(z_j) \propto \exp\{\mathbb{E}_{-j}[\log p(z_j|z_{-j}, \mathbf{x})]\} \qquad (2.17)$$

Since the denominator of the conditional does not depend on z_j , we can equivalently write:

$$q^*(z_j) \propto \exp\{\mathbb{E}_{-j}[\log p(z_j, z_{-j}, \mathbf{x})]\} \qquad (2.18)$$

In addition, we can also use KL divergence to derive the CAVI from the Equation 2.14, we replace $\mathbb{E}_{-j}[\log p(z_j|z_{-j}, \mathbf{x})]$ with $\log(\tilde{p}_j(z_j|z_{-j}, \mathbf{x}))$, then the Equation 2.14 could be written as,

$$\begin{aligned}\mathcal{L}_j &= \int q(z_j)\log(\tilde{p}_j(z_j|z_{-j}, \mathbf{x}))dz_j - \int q(z_j)\log q(z_j)dz_j \\ &= \int q(z_j)\log\frac{\tilde{p}_j(z_j|z_{-j}, \mathbf{x})}{q(z_j)}dz_j \\ &= -KL(q||p) \end{aligned} \tag{2.19}$$

Hence, from the Equation 2.19, to **maximize** the ELBO ($\mathcal{L}_j$) is equal to **minimize** the KL divergence, which is $q = p$. Hence, we got $q(z_j) = \tilde{p}_j(z_j|z_{-j}, \mathbf{x})$. Thus, we obtain a general expression for the optimal solution $q^*(z_j)$, then

$$\log q^*(z_j) = \mathbb{E}_{-j}[\log p(z_j|z_{-j}, \mathbf{x})] + \text{const} \tag{2.20}$$

Hence,

$$q^*(z_j) \propto exp\{\mathbb{E}_{-j}[\log p(z_j|z_{-j}, \mathbf{x})]\} \tag{2.21}$$

The coordinate ascent algorithm is to iteratively update each $q(z_j)$, so the ELBO converges to a *local minimum*. These equations underlie the CAVI algorithm, presented as Algorithm 1.

Algorithm 1: Coordinate ascent variational inference (CAVI)

Input: A model $p(\mathbf{z}, \mathbf{x})$, and data $\mathbf{x}$
Output: Variational density $q(z) = \prod_{j=1}^{m} q_j(z_j)$
Initialize: Variational factors $q_j(z_j)$
while *the ELBO has not converged* **do**
  for $j \in \{1, ..., m\}$ **do**
    $q(z_j) \propto exp\{\mathbb{E}_{-j}[\log p(z_j, |z_{-j}, \mathbf{x})]\}$
  end
  Compute $ELBO(q) = \mathbb{E}_q[\log p(\mathbf{z}, \mathbf{x}|\theta)] - \mathbb{E}_q[\log q(\mathbf{z})]$
end
return $q(z)$

Hence, the CAVI algorithm is closely related to Gibbs sampling method. The Gibbs sampling maintains a realization of the latent variables and iteratively samples from each variables complete conditional. The Equation 2.17 uses the same complete conditional. It takes the expected log, and uses this quantity to iteratively set each variables variational factor.

2.5 Case Study: Bayesian mixture of Gaussian

As an example, we return to the simple mixture of Gaussian mixture model. In Figure 2.2, it shows that there are K mixture components and n real valued data point $x_{1:n}$. The latent variables are K real value mean parameters $\mu_{1:K}$ and n latent class assignments $\mathbf{c} = c_{1:n}$. The assignment c_i indicates which latent cluster c_i comes from. In detail, c_i is an indicator K-vector, all zeros except for a one in position corresponding to x_i's cluster. There is a

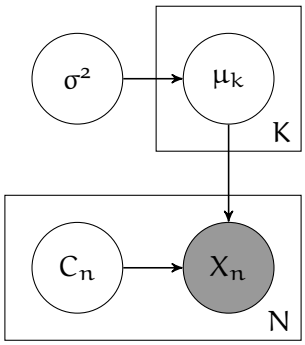

Figure 2.2: Graphical model representation of the K component Bayesian mixture of Gaussian model

fixed hyper-parameter σ^2, the variance of the normal prior on the μ_k's. We assume the observation variance is one and take a uniform prior over the mixture components. Hence, the full hierarchical model is

$$\begin{aligned} \mu_k &\sim \mathcal{N}(0, \sigma^2) \\ c_i &\sim Categorical(1/K, ..., 1/K) \\ x_i|c_i, \boldsymbol{\mu} &\sim \mathcal{N}(c_i^T \boldsymbol{\mu}, 1) \end{aligned}$$

For a sample of size n, the joint density of latent and observed variables is,

$$\begin{aligned} p(\boldsymbol{\mu}, \boldsymbol{c}, \boldsymbol{x}) &= p(\boldsymbol{\mu}) \prod_{i=1}^{n} p(c_i) p(x_i|c_i, \boldsymbol{\mu}) \\ &= \prod_{i=1}^{K} p(\mu_k) \prod_{i=1}^{n} p(c_i) p(x_i|c_i, \boldsymbol{\mu}) \end{aligned} \tag{2.22}$$

As discussed in Section 2.3, each latent variable has its own variational factor. The first factor $q(\mu_k; m_k, s_k^2)$ is a Gaussian distribution on the k-th mixture components mean, parameterized by its own mean m_k and variance s_k^2 (note these are not the same as the means of the cluster Gaussian as these are completely different distributions!). The second factor $q(c_i; \varphi_i)$ is a distribution on the i-th observations mixture assignment with assignment probabilities given by a K-vector φ_i, and c_i being the bit-vector (with one 1) associated with data point i.

$$q(\boldsymbol{\mu}, \boldsymbol{c}) = \prod_{k=1}^{K} q(\mu_k; m_k, s_k^2) \prod_{i=1}^{n} q(c_i; \varphi_i) \tag{2.23}$$

Having specified the joint distribution and now the mean-field family, we have now completely specified the variational inference problem for the mixture of Gaussians. The optimization will now focus on maximizing the ELBO with respect to the variational parameters for each latent variable.

According to Equation 2.7 and 2.22, the ELBO in the Gaussian mixture model is written

as,

$$
\begin{aligned}
\text{ELBO}(m, s^2, \varphi) &= \mathbb{E}_q[\log p(\mu, \boldsymbol{c}, \boldsymbol{x})] - \mathbb{E}_q[\log q(\boldsymbol{\mu}, \boldsymbol{c})] \\
&= \mathbb{E}_q[\log \Big\{ \prod_{i=1}^{K} p(\mu_k) \prod_{i=1}^{n} p(c_i) p(x_i|c_i, \boldsymbol{\mu}) \Big\}] \\
&\quad - \mathbb{E}_q[\log \Big\{ \prod_{k=1}^{K} q(\mu_k; m_k, s_k^2) \prod_{i=1}^{n} q(c_i; \varphi_i) \Big\}] \\
&= \sum_{i=1}^{K} \underbrace{\mathbb{E}_q[\log p(\mu_k); m_k, s_k^2]}_{\text{part 1}} \\
&\quad + \sum_{i=1}^{n} \Big\{ \underbrace{\mathbb{E}_q[\log p(c_i); \varphi_i]}_{\text{part 2}} + \underbrace{\mathbb{E}_q[\log p(x_i|c_i, \boldsymbol{\mu}); \varphi_i, m, s^2]}_{\text{part 3}} \Big\} \\
&\quad - \sum_{i=1}^{n} \Big\{ \underbrace{\mathbb{E}_q[\log q(c_i; \varphi_i)]}_{\text{part 4}} \Big\} - \sum_{i=1}^{K} \Big\{ \underbrace{\mathbb{E}_q[\log q(\mu_k; m_k, s_k^2)]}_{\text{part 5}} \Big\}
\end{aligned} \tag{2.24}
$$

Hence, the ELBO can computed in five parts shown in Equation 2.24,

1. **part 1**:

$$
\begin{aligned}
\mathbb{E}_q[\log p(\mu_k)] &= \mathbb{E}_q[\log \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{\mu_k^2}{2\sigma^2}}] \\
&= \mathbb{E}_q[\log\Big(\frac{1}{\sqrt{2\pi\sigma^2}}\Big) - \frac{\mu_k^2}{2\sigma^2}] \\
&= \log\Big(\frac{1}{\sqrt{2\pi\sigma^2}}\Big) - \frac{E[\mu_k^2]}{2\sigma^2} \\
&= \underbrace{\log\Big(\frac{1}{\sqrt{2\pi\sigma^2}}\Big) - \frac{Var[\mu_k] + \mathbb{E}[\mu_k]^2}{2\sigma^2}}_{\mathbb{E}[\mu_k] = m_k, Var[\mu_k] = s_k^2}
\end{aligned} \tag{2.25}
$$

2. **part 2**:

$$
\mathbb{E}_q[\log p(c_i)] = \log(1/K) \tag{2.26}
$$

3. **part 3**:

$$
\begin{aligned}
\mathbb{E}_q[\log p(x_i|c_i, \boldsymbol{\mu})] &= \mathbb{E}_q[\log \underbrace{\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_i - \mu_{c_i})^2}{2\sigma^2}}}_{\sigma=1}] \\
&= \mathbb{E}_q[\log(\frac{1}{\sqrt{2\pi}}) e^{\frac{-(x_i - \mu_{c_i})^2}{2}}] \\
&= \mathbb{E}_q[\log(1/\sqrt{2\pi}) - (x_i^2 - 2x_i\mu_{c_i} + \mu_{c_i}^2)/2] \\
&= \log(1/\sqrt{2\pi}) - (x_i^2 - 2x_i E_q[\mu_{c_i}] + E_q[\mu_{c_i}^2])/2 \\
&= \log(1/\sqrt{2\pi}) - \Big(x_i^2 - 2x_i \sum_k c_{ik} E_q[\mu_k]
\end{aligned}
$$

$$
\begin{aligned}
&+ \sum_k c_{ik}(Var_q[\mu_k] + E_q[\mu_k]^2)\Big)/2 \\
&= \log(1/\sqrt{2\pi}) - \Big(x_i^2 - 2x_i \sum_k \varphi_{ik} m_k \\
&+ \sum_k \varphi_{ik}(s_k^2 + m_k^2)\Big)/2
\end{aligned} \tag{2.27}
$$

4. **part 4**:

$$
\begin{aligned}
\mathbb{E}_q[\log q(c_i)] &= \underbrace{\int_{c_i} q(c_i)\log(q(c_i))dc_i}_{\text{negative entropy}} \\
&= \sum_{k=1}^{K} \varphi_{ik}\log\varphi_{ik}
\end{aligned} \tag{2.28}
$$

5. **part 5**:

$$
\begin{aligned}
\mathbb{E}_q[\log q(\mu_k)] &= \int_{\mu_k} q(\mu_k)\log\, q(\mu_k)d\mu_k \\
&= \underbrace{\int_{\mu_k} (2\pi s_k^2)^{-\frac{1}{2}} e^{-(\mu_k - m_k)^2/2s_k^2} \log\Big((2\pi s_k^2)^{-\frac{1}{2}} e^{-(\mu_k - m_k)^2/2s_k^2}\Big) d\mu_k}_{m_k,\, s_k^2 \text{ are variational factor(mean, variance) for } \mu_k} \\
&= \int_{\mu_k} (2\pi s_k^2)^{-\frac{1}{2}} e^{-(\mu_k - m_k)^2/2s_k^2} \Big(\log(2\pi s_k^2)^{-\frac{1}{2}} - (\mu_k - m_k)^2/2s_k^2\Big) d\mu_k \\
&= -\frac{1}{2}\log(2\pi s_k^2) \underbrace{\int_{\mu_k} (2\pi s_k^2)^{-\frac{1}{2}} e^{-(\mu_k - m_k)^2/2s_k^2} d\mu_k}_{=1} \\
&\quad - \frac{1}{2s_k^2} \underbrace{\int_{\mu_k} (\mu_k - m_k)^2 (2\pi s_k^2)^{-\frac{1}{2}} e^{-(\mu_k - m_k)^2/2s_k^2} d\mu_k}_{=s^2, \text{ because } E(x-\mu)^2 = \sigma^2} \\
&= -\frac{1}{2}\log(2\pi s_k^2) - \frac{1}{2}
\end{aligned} \tag{2.29}
$$

Next, in each term, we have made explicit the dependence on the variational parameters. Each expectation can be computed in closed form. The CAVI algorithm updates each variational parameter in turn. We first derive the update for the variational cluster assignment factor; we then derive the update for the variational mixture component factor.

2.5.1 The variational density of the mixture assignments

We first derive the variational update for the cluster assignment c_i. According to the CAVI algorithm Equation 2.17, we can write down variational update for cluster assignment c,

$$
q^*(c;\varphi) \propto \exp\Big\{\mathbb{E}_{q_{-c}}[\log p(\mu, c, x)]\Big\} \tag{2.30}
$$

And the log joint distribution of $\log p(\mu, c, x)$ is,

$$\log p(\mu, c, x) = \sum_{i=1}^{K} \log p(\mu_k) + \sum_{i=1}^{n} \Big(\log p(c_i) + \log p(x_i | c_i, \boldsymbol{\mu}) \Big) \tag{2.31}$$

Then variational term $q^*(c; \varphi)$ can be written in Equation 2.32, which remove terms do not contain c.

$$q^*(c; \varphi) \propto exp\Big\{ \mathbb{E}_{q_{-c}} [\sum_{i=1}^{n} \Big(\log p(c_i) + \log p(x_i | c_i, \boldsymbol{\mu}) \Big)] \Big\} \tag{2.32}$$

Hence, the variational update for the each cluster assignment c_i is,

$$\begin{aligned} q^*(c_i; \varphi) &\propto exp\Big\{ \mathbb{E}_{q_{-c}} [\log p(c_i) + \log p(x_i | c_i, \boldsymbol{\mu})] \Big\} \\ &= exp\Big\{ \log p(c_i) + \mathbb{E}_{q_{-c}} [\log p(x_i | c_i, \boldsymbol{\mu})] \Big\} \end{aligned} \tag{2.33}$$

The terms in the exponent are the components of the joint density that depend on c_i. The expectation in the second term is over the mixture components μ.

The first term of Equation 2.33 is log prior of c_i. It is the same for all possible values of c_i, then,

$$\log p(c_i) = \log(1/K) = -\log(K) \tag{2.34}$$

Next, we consider the second of Equation 2.33, which is expected log of the c_i-th Gaussian density. Recalling that the c_i is an indicator vector, we can write as,

$$p(x_i | c_i, \boldsymbol{\mu}) = \prod_{k=1}^{K} p(x_i | \mu_k)^{c_{ik}} \tag{2.35}$$

Then the expected log probability is,

$$\begin{aligned} \mathbb{E}_{q_{-c}} [\log p(x_i | c_i, \boldsymbol{\mu})] &= \sum_{k=1}^{K} c_{ik} \mathbb{E}_{q_\mu} [\log p(x_i | \mu_k); m_k, s_k^2] \\ &= \sum_{k=1}^{K} c_{ik} \mathbb{E}_{q_\mu} [-(x_i - \mu_k)^2/2; m_k, s_k^2] + \text{const} \\ &= \sum_{k=1}^{K} c_{ik} \mathbb{E}_{q_\mu} [-x_i^2/2 + x_i \mu_k - \mu_k^2/2); m_k, s_k^2] + \text{const} \\ &= \sum_{k=1}^{K} c_{ik} (\mathbb{E}_{q_\mu} [\mu_k; m_k, s_k^2] x_i - \mathbb{E}_{q_\mu} [\mu_k^2; m_k, s_k^2]/2) + \text{const} \end{aligned} \tag{2.36}$$

In each line we remove terms that are constant with respect to c_i. This calculation requires $\mathbb{E}[\mu_k]$ and $\mathbb{E}[\mu_k^2]$ for each mixture component, both computable from the variational Gaussian on the kth mixture component.

Thus, the variational update for the ith cluster assignment is

$$\varphi_{ik} = q^*(c_i = k) \propto exp\Big\{ \mathbb{E}[\mu_k; m_k, s_k^2] x_i - \mathbb{E}[\mu_k^2; m_k, s_k^2]/2 \Big\} \tag{2.37}$$

2.5.2 The variational density of the mixture-component means

We can use same logic to derive the variational density $q^*(\mu; m, s^2)$ of the mixture component,

$$q(\mu; m, s^2) \propto exp\Big\{\mathbb{E}_{q_{-\mu}}[\log p(\mu, \boldsymbol{c}, \boldsymbol{x})]\Big\} \tag{2.38}$$

Because, the log joint distribution of $\log p(\mu, \boldsymbol{c}, \boldsymbol{x})$ is,

$$\log p(\mu, \boldsymbol{c}, \boldsymbol{x}) = \sum_{k=1}^{K} \log p(\mu_k) + \sum_{i=1}^{n} \Big(\log p(c_i) + \log p(x_i|c_i, \boldsymbol{\mu})\Big) \tag{2.39}$$

Then, variational term $q^*(\mu; m, s^2)$ can be written in Equation 2.40, which remove terms do not contain μ.

$$q^*(\mu; m, s^2) \propto exp\Big\{\mathbb{E}_{q_{-\mu}}[\sum_{k=1}^{K} \log p(\mu_k) + \sum_{i=1}^{n} \Big(\log p(x_i|c_i, \boldsymbol{\mu})\Big)]\Big\} \tag{2.40}$$

We turn to the variational density of the kth mixture component. According to the Equation 2.40, we have

$$q^*(\mu_k; m_k, s_k^2) \propto exp\Big\{\log p(\mu_k) + \sum_{i=1}^{n} \mathbb{E}_{q_{-\mu}}[\log p(x_i|c_i, \boldsymbol{\mu})]\Big\} \tag{2.41}$$

We now calculate the unnormalized log of this coordinate-optimal $q(\mu_k; m_k, s_k^2)$. Recall φ_{ik} is the probability that ith observation comes from the kth cluster. Because c_i is an indicator vector, then $\varphi_{ik} = \mathbb{E}[c_{ik}; \varphi_i]$, then,

$$\log q(\mu_k) = \log p(\mu_k) + \sum_{i=1}^{n} \mathbb{E}_{q_c}[\log p(x_i|c_i, \mu_k)] + \text{const} \tag{2.42}$$

$$= \log p(\mu_k) + \sum_{i=1}^{n} \mathbb{E}_{q_c}[c_{ik} \log p(x_i|\mu_k); \varphi_i] + \text{const} \tag{2.43}$$

$$= -\frac{\mu_k^2}{2\sigma^2} + \sum_{i=1}^{n} \mathbb{E}_{q_c}[c_{ik}; \varphi_i] \log p(x_i|\mu_k) + \text{const} \tag{2.44}$$

$$= -\frac{\mu_k^2}{2\sigma^2} + \sum_{i=1}^{n} \varphi_{ik}\Big(-\frac{(x_i - \mu_k)^2}{2}\Big) + \text{const} \tag{2.45}$$

$$= -\frac{\mu_k^2}{2\sigma^2} + \sum_{i=1}^{n} \Big(\varphi_{ik} x_i \mu_k - \frac{\varphi_{ik}\mu_k^2}{2}\Big) + \text{const} \tag{2.46}$$

$$= \Big(\sum_{i=1}^{n} \varphi_{ik} x_i\Big)\mu_k - \Big(\frac{1}{2\sigma^2} + \sum_{i=1}^{n} \frac{\varphi_{ik}}{2}\Big)\mu_k^2 + \text{const} \tag{2.47}$$

This calculation reveals that the coordinate-optimal variational density of μ_k is an exponential family with sufficient statistics $\{\mu_k, \mu_k^2\}$, and natural parameters $\{\sum_{i=1}^{n} \varphi_{ik} x_i, -\frac{1}{2\sigma^2} - \sum_{i=1}^{n} \frac{\varphi_{ik}}{2}\}$, which could be expressed as Gaussian in terms of mean and variance for updating $q(\mu_k)$,

$$m_k = \frac{\sum_{i=1}^{n} \varphi_{ik} x_i}{1/\sigma^2 + \sum_{i=1}^{n} \varphi_{ik}}, \qquad s_k^2 = \frac{1}{1/\sigma^2 + \sum_{i=1}^{n} \varphi_{ik}} \tag{2.48}$$

2.5.3 CAVI for the mixture of Gaussians

Algorithm 2 presents coordinate-ascent variational inference for the Bayesian mixture of Gaussians. It combines the variational updates in Equation 2.37 and Equation 2.48. The algorithm requires computing the ELBO of Equation 2.24. We use the ELBO to track the progress of the algorithm and asses when it has converged.

Algorithm 2: Coordinate ascent variational inference (CAVI)

Input: Data $x_{1:n}$ number of components K, prior variance of component means σ^2
Output: Variational densities $q(\mu_k; m_k, s_k^2)$ (Gaussian) and $q(c_i; \varphi_i)$ (K-categorical)
Initialize: Variational parameters $\boldsymbol{m} = m_{1:k}, s^2 = s^2_{1:k}, \text{and} \boldsymbol{\varphi} = \varphi_{1:n}$
while *the ELBO has not converged* **do**
  for $i \in \{1, ..., n\}$ **do**
    Set $\varphi_{ik} \propto \exp\Big\{\mathbb{E}[\mu_k; m_k, s_k^2]x_i - \mathbb{E}[\mu_k^2; m_k, s_k^2]/2\Big\}$
  end
  for $j \in \{1, ..., n\}$ **do**
    Set $m_k \leftarrow \frac{\sum_{i=1}^n \varphi_{ik} x_i}{1/\sigma^2 + \sum_{i=1}^n \varphi_{ik}}$
    Set $s_k^2 \leftarrow \frac{1}{1/\sigma^2 + \sum_{i=1}^n \varphi_{ik}}$
  end
  Compute $\text{ELBO}(\boldsymbol{m}, \boldsymbol{s}^2, \boldsymbol{\varphi})$
end
return $q(z)$

2.6 Variational Inference with Exponential Family

2.6.1 Exponential Family

A probability density from exponential family can be expressed in a general form as:

$$p(x|\eta) = h(x)exp\{\eta^T T(x) - A(\eta)\}$$

Where the parameter vector η is referred to as the *natural parameter* for given function T. The function $T(x)$ is referred to as the *sufficient statistic*. And the $A(\eta)$ is referred to as the *log normalizer*, because

$$1 = \int h(x)exp\{\eta^T T(x) - A(\eta)\}dx$$

then,

$$\begin{aligned} 1 &= \int h(x)exp\{\eta^T T(x)\}exp\{-A(\eta)\}dx \\ \frac{1}{exp\{-A(\eta)\}} &= \int h(x)exp\{\eta^T T(x)\}dx \\ exp\{A(\eta)\} &= \int h(x)exp\{\eta^T T(x)\}dx \\ A(\eta) &= \log \int h(x)exp\{\eta^T T(x)\}dx \end{aligned}$$

Chapter 3

Topics Model

3.1 Background

TBD

3.2 Latent Dirichlet Allocation

Latent Dirichlet allocation (LDA) [1] is a generative probabilistic model of a corpus. The basic idea is that documents are represented as random mixtures over latent topics, where each topic is characterized by a distribution over words.

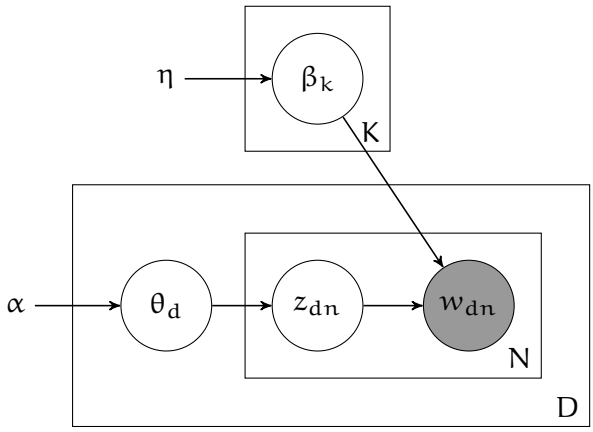

Figure 3.1: Graphical model representation of LDA. The outer plate represents documents, while the inner plate represents the repeated choice of topics and words within a document, and the "D" means the total D number of documents, and "N" means the total N number of words in a documents. The "grey" color circle is observed variable; and the others are unobserved variable. And the "α, γ" are hyper-parameters for Dirichlet distribution.

As shown in Figure 3.1, let K be a specific number of topics, the D denotes the number of documents, and the N denotes the number of words in each documents. Hence, LDA defines the following generative process:

1. For each topics in $k \in K$,

(a) draw a distribution over words $\beta_k \sim Dirichlet(\eta)$

2. For each document $d \in D$:

   (a) draw a vector of topic proportions $\theta_d \sim$ Dirchlet(α).

   (b) For each word $w_n \in N$:

      i. Choose a topic $z_{dn} \sim$ Multinomial(θ_d)

      ii. Choose a word $w_{dn} \sim$ Categorical(βz_{dn})

3.3 Inference Methods

In Figure 3.1, the latent variables in LDA are $\boldsymbol{\beta}, \boldsymbol{\theta}$, and $\boldsymbol{z}$. Hence, the goal of inference algorithm is to infer the latent variables.
The posterior distribution of hidden variables given a document:

$$p(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{z}|\boldsymbol{w}, \alpha, \eta) = \frac{p(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{z}, \boldsymbol{w}, |\alpha, \eta)}{p(\boldsymbol{w}|\alpha, \eta)}$$

Unfortunately, this distribution is intractable to compute in general. However, a wide variety of approximate inference algorithm can be used in LDA, such as MCMC, variational inference.

3.3.1 Variational Inference

Because the original distribution $p(x)$ is intractable, the goal of variational inference is to find a tractable distribution $q(x)$ to approximate the original. Hence, the computation becomes optimization problem which minimize the difference between original distribution and approximate distribution. KL divergence helps to calculate the distance between original distribution and approximate distribution.

In LDA model, the joint density of latent variables ($\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{z}$) observed variables is,

$$p(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{z}, \boldsymbol{w}|\alpha, \eta) = \prod_{k=1}^{K} p(\beta_k|\eta) \prod_{d=1}^{D} \Big[p(\theta_d|\alpha) \prod_{n=1}^{N} [p(z_{dn}|\theta_d) p(w_{dn}|\beta_{1:K}, z_{dn})] \Big]$$

Then, for echo latent variable, we posit a mean-field variational family.

$$q(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{z}) = \prod_{k=1}^{K} q(\beta_k; \lambda_k) \prod_{d=1}^{D} \Big[q(\theta_d; \gamma_d) \prod_{n=1}^{N} q(z_{dn}; \phi_{dn}) \Big]$$

And,

$$\begin{aligned} q(\beta_k) &\sim Dirichlet(\lambda_k) \\ q(\theta_d) &\sim Dirichlet(\gamma_d) \\ q(z_{dn} = k) &\sim Categorical(\phi_{dn}^k) \end{aligned}$$

Then, then log likelihood of documents is,

$$\begin{aligned}
\log p(\boldsymbol{w}|\alpha,\eta) &= \log \int_{\boldsymbol{\theta}}\int_{\boldsymbol{\beta}}\int_{\boldsymbol{z}} p(\boldsymbol{\beta},\boldsymbol{\theta},\boldsymbol{z},\boldsymbol{w}|\alpha,\eta)\,d\boldsymbol{\theta}\,d\boldsymbol{\beta}\,d\boldsymbol{z} \\
&= \log \int_{\boldsymbol{\theta}}\int_{\boldsymbol{\beta}}\int_{\boldsymbol{z}} \frac{p(\boldsymbol{\beta},\boldsymbol{\theta},\boldsymbol{z},\boldsymbol{w}|\alpha,\eta)q(\boldsymbol{\beta},\boldsymbol{\theta},\boldsymbol{z})}{q(\boldsymbol{\beta},\boldsymbol{\theta},\boldsymbol{z})}\,d\boldsymbol{\theta}\,d\boldsymbol{\beta}\,d\boldsymbol{z} \\
&\geqslant \int_{\boldsymbol{\theta}}\int_{\boldsymbol{\beta}}\int_{\boldsymbol{z}} \log\Big(\frac{p(\boldsymbol{\beta},\boldsymbol{\theta},\boldsymbol{z},\boldsymbol{w}|\alpha,\eta)}{q(\boldsymbol{\beta},\boldsymbol{\theta},\boldsymbol{z})}\Big) q(\boldsymbol{\beta},\boldsymbol{\theta},\boldsymbol{z})\,d\boldsymbol{\theta}\,d\boldsymbol{\beta}\,d\boldsymbol{z} \\
&= \int_{\boldsymbol{\theta}}\int_{\boldsymbol{\beta}}\int_{\boldsymbol{z}} \log\Big(p(\boldsymbol{\beta},\boldsymbol{\theta},\boldsymbol{z},\boldsymbol{w}|\alpha,\eta)\Big) q(\boldsymbol{\beta},\boldsymbol{\theta},\boldsymbol{z})\,d\boldsymbol{\theta}\,d\boldsymbol{\beta}\,d\boldsymbol{z} \\
&\quad - \int_{\boldsymbol{\theta}}\int_{\boldsymbol{\beta}}\int_{\boldsymbol{z}} \log\Big(q(\boldsymbol{\beta},\boldsymbol{\theta},\boldsymbol{z})\Big) q(\boldsymbol{\beta},\boldsymbol{\theta},\boldsymbol{z})\;d\boldsymbol{\theta}\,d\boldsymbol{\beta}\,d\boldsymbol{z} \\
&= \underbrace{\mathbb{E}_q[\log p(\boldsymbol{\beta},\boldsymbol{\theta},\boldsymbol{z},\boldsymbol{w}|\eta,\alpha)] - \mathbb{E}_q[\log q(\boldsymbol{\beta},\boldsymbol{\theta},\boldsymbol{z})]}_{\mathrm{ELBO}(\boldsymbol{\beta},\boldsymbol{\theta},\boldsymbol{z})}
\end{aligned}$$

The $\mathrm{ELBO}(\boldsymbol{\beta},\boldsymbol{\theta},\boldsymbol{z})$ is called evidence lower bound, because,

$$\log p(\boldsymbol{w}|\alpha,\eta) = \mathrm{ELBO}(\boldsymbol{\beta},\boldsymbol{\theta},\boldsymbol{z}) + \mathbb{KL}(q\|p)$$

Hence, to minimize the KL divergence is equal to maximize $\mathrm{ELBO}(\boldsymbol{\beta},\boldsymbol{\theta},\boldsymbol{z})$, which is easier.
According to the Equation 2.8, the ELBO for LDA can be defined as,

$$\begin{aligned}
\mathrm{ELBO}(\boldsymbol{\beta},\boldsymbol{\theta},\boldsymbol{z}) &= \mathbb{E}_q[\log p(\boldsymbol{\beta},\boldsymbol{\theta},\boldsymbol{z},\boldsymbol{w}|\eta,\alpha)] - \mathbb{E}_q[\log q(\boldsymbol{\beta},\boldsymbol{\theta},\boldsymbol{z})] \\
&= \mathbb{E}_q[\log\Big\{\prod_{k=1}^{K} p(\beta_{1:K}|\eta)\prod_{d=1}^{D}\Big[p(\theta_d|\alpha)\prod_{n=1}^{N}[p(z_{dn}|\theta_d)p(w_{dn}|\beta_{1:k},z_{dn})]\Big]\Big\}] \\
&\quad - \mathbb{E}_q[\log\Big\{\prod_{k=1}^{K} q(\beta_k;\lambda_k)\prod_{d=1}^{D}\Big[q(\theta_d;\gamma_d)\prod_{n=1}^{N} q(z_{dn};\phi_{dn})\Big]\Big\}] \\
&= \sum_{K}\mathbb{E}_q[\log p(\beta_k|\eta)] + \sum_{D}\Big\{\mathbb{E}_q[\log p(\theta_d|\alpha)] + \sum_{N}\Big\{\mathbb{E}_q[\log p(z_{dn}|\theta_d)] \\
&\quad + \mathbb{E}_q[\log p(w_{dn}|\beta_{1:K},z_{dn})]\Big\}\Big\} - \sum_{K}\mathbb{E}_q[\log q(\beta_k;\lambda_k)] \\
&\quad - \sum_{D}\Big\{\mathbb{E}_q[\log q(\theta_d;\gamma_d)] - \sum_{N}\mathbb{E}_q[\log q(z_{dn};\phi_{dn})]\Big\} \qquad (3.1)
\end{aligned}$$

Then, we can derive the variational update for each variational parameters according to CAVI algorithm form Equation 2.17 and 2.18.

First, deriving variational update for β, which remove terms do not contain β, is

$$\begin{aligned}
q^*(\beta;\lambda) &\propto \exp\Big\{\mathbb{E}_q[\log p(\boldsymbol{\beta},\boldsymbol{\theta},\boldsymbol{z},\boldsymbol{w}|\eta)]\Big\} \\
&\propto \exp\Big\{\mathbb{E}_q[\log\Big(\prod_{k=1}^{K} p(\beta_{1:K}|\eta)\prod_{d=1}^{D}\prod_{n=1}^{N} p(z_{dn}|\theta_d)p(w_{dn}|\beta_{1:k},z_{dn})\Big)]\Big\} \\
&\propto \exp\Big\{\mathbb{E}_q[\log\Big(\prod_{k=1}^{K} p(\beta_{1:K}|\eta)\prod_{d=1}^{D}\prod_{n=1}^{N} p(w_{dn}|\beta_{1:k})^{\delta(z_{dn},k)}\Big)]\Big\}
\end{aligned}$$

Hence, the variational update for particular topic β_k is,

$$
\begin{aligned}
q^*(\beta_k;\lambda_k) &\propto \exp\Big\{\mathbb{E}_q[\log\Big(p(\beta_k|\eta)\prod_{d=1}^{D}\prod_{n=1}^{N}p(w_{dn}|\beta_k)^{\delta(z_{dn},k)}\Big)]\Big\} \\
&= \exp\Big\{\mathbb{E}_q[\log\Big(\underbrace{\mathrm{Dir}_\nu(\eta)\prod_{d=1}^{D}\prod_{n=1}^{N}\beta_{k,w_{dn}}^{\delta(z_{dn},k)}}_{\text{Dir(post)}\propto\text{ Dir(prior)}\times\text{ Cat(likelihood)}}\Big)] \\
&\propto \exp\Big\{\mathbb{E}_q[\log\Big(\mathrm{Dir}_\nu(\eta+\sum_{d=1}^{D}\sum_{n=1}^{N}w_{dn}^{\delta(z_{dn},k)})\Big)] \\
&\propto \exp\Big\{\mathbb{E}_q[\log\Big(\beta_k^{\eta-1+\sum_{d=1}^{D}\sum_{n=1}^{N}w_{dn}^{\delta(z_{dn},k)}}\Big)]\Big\} \\
&= \exp\Big\{\mathbb{E}_q[\Big(\eta-1+\sum_{d=1}^{D}\sum_{n=1}^{N}w_{dn}^{\delta(z_{dn},k)}\Big)\times\log(\beta_k)]\Big\} \\
&= \exp\Big\{\mathbb{E}_{\prod_{d=1}^{D}\prod_{n=1}^{N}q(z_{dn})}[\eta-1+\sum_{d=1}^{D}\sum_{n=1}^{N}w_{dn}^{\delta(z_{dn},k)}]\times\log(\beta_k)\Big\} \\
&= \exp\Big\{\underbrace{\Big[\eta-1+\sum_{d=1}^{D}\sum_{n=1}^{N}w_{dn}\phi_{dn}^{k}\Big]}_{\text{natural parameter}}\times\underbrace{\log(\beta_k)}_{\text{sufficient statistic}}\Big\}
\end{aligned}
$$

Hence, $q^*(\beta_k;\lambda_k)$ is Dirichlet distribution with natural parameter$[\eta-1+\sum_{d=1}^{D}\sum_{n=1}^{N}w_{dn}\phi_{dn}^{k}\Big]$. So,

$$
\lambda_k = \eta+\sum_{d=1}^{D}\sum_{n=1}^{N}w_{dn}\phi_{dn}^{k} \tag{3.2}
$$

Second, deriving variational update for θ, which remove terms do not contain θ, is

$$
\begin{aligned}
q^*(\theta;\gamma) &\propto \exp\Big\{\mathbb{E}_q[\log p(\boldsymbol{\beta},\boldsymbol{\theta},\boldsymbol{z},\boldsymbol{w}|\eta)]\Big\} \\
&\propto \exp\Big\{\mathbb{E}_q[\log\Big(\prod_{d=1}^{D}\Big[p(\theta_d|\alpha)\prod_{n=1}^{N}p(z_{dn}|\theta_d)\Big]\Big)]\Big\}
\end{aligned}
$$

Hence, the variational update for particular topic θ_d is,

$$
\begin{aligned}
q^*(\theta_d;\gamma_d) &\propto \exp\Big\{\mathbb{E}_q[\log\Big(p(\theta_d|\alpha)\prod_{n=1}^{N}p(z_{dn}|\theta_d)\Big)]\Big\} \\
&\propto \exp\Big\{\mathbb{E}_q[\log\Big(\underbrace{\mathrm{Dir}_k(\alpha)\times\prod_{n=1}^{N}\mathrm{Mult}(z_{dn}|\theta_d)}_{\text{Dir(post)}\propto\text{ Dir(prior)}\times\text{ Mult(likelihood)}}\Big)]\Big\} \\
&\propto \exp\Big\{\mathbb{E}_q[\log\Big(\mathrm{Dir}_k(\alpha+\sum_{n=1}^{N}z_{dn}^{\delta(z_{dn},k)})\Big)]\Big\}
\end{aligned}
$$

$$
\begin{aligned}
&\propto \exp\Big\{\mathbb{E}_q[\log\Big(\prod_{k=1}^{K}\theta_{dk}^{\alpha_k-1+\sum_{n=1}^{N}\delta(z_{dn},k)}\Big)\Big\}\\
&= \exp\Big\{\mathbb{E}_q[\sum_{k=1}^{K}\log\Big(\theta_{dk}^{\alpha_k-1+\sum_{n=1}^{N}\delta(z_{dn},k)}\Big)]\Big\}\\
&= \exp\Big\{\mathbb{E}_q[\sum_{k=1}^{K}\Big((\alpha_k-1+\sum_{n=1}^{N}\delta(z_{dn},k))\times\log(\theta_{dk})\Big)]\Big\}\\
&= \exp\Big\{\mathbb{E}_{q(z_{dn})}[\sum_{k=1}^{K}\Big((\alpha_K-1+\sum_{n=1}^{N}\delta(z_{dn},k))\times\log(\theta_{dk})]\Big)\Big\}\\
&= \exp\Big\{\Big[(\alpha_1-1+\sum_{n=1}^{N}\delta(z_{dn},1)\phi_{dn}^{1})\times\log(\theta_{d1})\Big]+\ldots\\
&\quad+\Big[(\alpha_k-1+\sum_{n=1}^{N}\delta(z_{dn},k)\phi_{dn}^{k})\times\log(\theta_{dk})\Big]\Big\}\\
&= \exp\Big\{\underbrace{[(\alpha_1-1+\sum_{n=1}^{N}\delta(z_{dn},1)\phi_{dn}^{1})\ldots(\alpha_k-1+\sum_{n=1}^{N}\delta(z_{dn},k)\phi_{dn}^{k})]^T}_{\text{natural parameter}}\\
&\quad\underbrace{[\log(\theta_{d1})\ldots\log(\theta_{dk})]}_{\text{sufficient statistics}}\Big\}
\end{aligned}
$$

Hence, $q^*(\theta_d;\gamma_d)$ is Dirichlet distribution with natural parameter $[(\alpha_1-1+\sum_{n=1}^{N}\delta(z_{dn},1)\phi_{dn}^{1})\ldots(\alpha_k-1+\sum_{n=1}^{N}\delta(z_{dn},k)\phi_{dn}^{k})]$. Hence,

$$
\begin{aligned}
\gamma_d &= [(\alpha_1+\sum_{n=1}^{N}\delta(z_{dn},1)\phi_{dn}^{1})\ldots(\alpha_k+\sum_{n=1}^{N}\delta(z_{dn},k)\phi_{dn}^{k})]\\
&= \alpha+\sum_{n=1}^{N}\phi_{dn} \qquad (3.3)
\end{aligned}
$$

Third, deriving variational update for $q^*(z_{dn}=k;\phi_{dn})$, which remove terms do not contain z, is

$$
\begin{aligned}
q^*(z_{dn}=k;\phi_{dn}) &\propto \exp\Big\{\mathbb{E}_q[\log p(\boldsymbol{\beta},\boldsymbol{\theta},\boldsymbol{z},\boldsymbol{w}|\alpha,\eta)]\Big\}\\
&\propto \exp\Big\{\mathbb{E}_q[\log\Big(p(z_{dn}=k|\theta_d)p(w_{dn}|\beta_k,z_{dn})\Big)]\Big\}\\
&= \exp\Big\{\mathbb{E}_q[\log\Big(\theta_{dk}\times\beta_{k,w_{dn}}\Big)]\Big\}\\
&= \exp\Big\{\mathbb{E}_{q(\beta_k)q(\theta_d)}[\log(\theta_{dk})+\log(\beta_{k,w_{dn}})]\Big\}\\
&= \exp\Big\{\mathbb{E}_{q(\theta_d)}[\log(\theta_{dk})]+\mathbb{E}_{q(\beta_k)}[\log(\beta_{k,w_{dn}})]\Big\}\\
&= \exp\Big\{\Psi(\gamma_{dk})-\underbrace{\Psi\sum_{k}(\gamma_{dk})}_{\text{constant}}+\Psi(\lambda_{k,w_{dn}})-\Psi\sum_{v}(\lambda_{kv})\Big\}
\end{aligned}
$$

$$\propto \exp\Big\{\Psi(\gamma_{dk}) + \Psi(\lambda_{k,w_{dn}}) - \Psi(\sum_{v}(\lambda_{kv}))\Big\}$$
$$= \underbrace{\exp\Big\{\Psi(\gamma_{dk}) + \Psi(\lambda_{k,w_{dn}}) - \Psi(\sum_{v}(\lambda_{kv}))\Big\}}_{\text{nature parameter}} \times \underbrace{1}_{\text{sufficient statistics}}$$

Because, $q^*(z_{dn} = k; \phi_{dn})$ is categorical distribution with sufficient statistics 1, and nature parameter $\exp\Big\{\Psi(\gamma_{dk}) + \Psi(\lambda_{k,w_{dn}}) - \Psi(\sum_v(\lambda_{kv}))\Big\}$
Hence,

$$\phi^k_{dn} \propto \exp\Big\{\Psi(\gamma_{dk}) + \Psi(\lambda_{k,w_{dn}}) - \Psi(\sum_{v}\lambda_{kv})\Big\} \tag{3.4}$$

Finally, we got our CAVI algorithm for LDA model for updating variational parameters.

Algorithm 3: Coordinate ascent variational inference for LDA algorithm

Input: A set of words w in documents.
Output: Variational parameters $\boldsymbol{\lambda}, \boldsymbol{\gamma}, \boldsymbol{\phi}$
Initialize: Variational parameters $\boldsymbol{\lambda}, \boldsymbol{\gamma}, \boldsymbol{\phi}$ randomly
while *the ELBO (Equation 3.1) has not converged* **do**
  repeat
    for *each document d* **do**
      for *each word w* **do**
        Compute updates to ϕ and γ via Equations 3.4 and 3.3
      end
    end
  until $\boldsymbol{\gamma}, \boldsymbol{\phi}$ *have converged*;
  Compute update to λ via Equation 3.2.
end
return $\boldsymbol{\gamma}, \boldsymbol{\phi}, \boldsymbol{\lambda}$

3.3.2 Collapsed Gibbs Sampling

Recall the goal of LDA is to infer posterior distribution of hidden variables given a document:

$$p(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{z}|\boldsymbol{w}, \alpha, \eta) = \frac{p(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{z}, \boldsymbol{w}, |\alpha, \eta)}{p(\boldsymbol{w}|\alpha, \eta)}$$

And the original Gibbs sampling will sequentially sampling all variables from their distributions when conditioned on the current values of all other variables and data. And the Collapsed Gibbs Sampling sample only some variables by integrating other variables.

So, in this LDA scenario, we apply Collapsed Gibbs Sampling by integrating out β and θ by taking advantage of the fact that the Dirichlet is the conjugate prior of the multinomial. Then, the objective Collapse Gibbs sampling will be,

$$\begin{aligned} p(z_{dn}|\mathbf{z}_{-dn}, \mathbf{w}) &\propto p(z_{dn}, \mathbf{z}_{-dn}, w_{dn}, \mathbf{w}_{-dn}) \\ &= p(w_{dn}|\mathbf{z}_{-dn}, z_{dn}, \mathbf{w}_{-dn}) p(\mathbf{z}_{-dn}, z_{dn}, \mathbf{w}_{-dn}) \\ &= p(w_{dn}|\mathbf{z}_{-dn}, z_{dn}, \mathbf{w}_{-dn}) p(z_{dn}|\mathbf{z}_{-dn}, \mathbf{w}_{-dn}) p(\mathbf{z}_{-dn}, \mathbf{w}_{-dn}) \\ &\propto p(w_{dn}|\mathbf{z}_{-dn}, z_{dn}, \mathbf{w}_{-dn}) p(z_{dn}|\mathbf{z}_{-dn}) \end{aligned}$$

For the first term, we have,

$$\begin{aligned} p(w_{dn}|z_{dn} = k, \mathbf{z}_{-dn}, \mathbf{w}_{-dn}) &= \int_{\beta} p(w_{dn}, \beta|z_{dn} = k, \mathbf{z}_{-dn}, \mathbf{w}_{-dn}) d\beta \\ &= \int_{\beta} p(w_{dn}|\beta, z_{dn} = k, \mathbf{z}_{-dn}, \mathbf{w}_{-dn}) p(\beta|z_{dn} = k, \mathbf{z}_{-dn}, \mathbf{w}_{-dn}) d\beta \\ &= \int_{\beta_k} p(w_{dn}|\beta_k, z_{dn} = k) p(\beta_k|\mathbf{z}_{-dn}, \mathbf{w}_{-dn}) d\beta_k \end{aligned} \tag{3.5}$$

In Equation 3.5, the first term is,

$$p(w_{dn}|\beta_k, z_{dn} = k) = \beta_{k,w_{dn}}^{\delta(z_{dn},k)} \tag{3.6}$$

and the second term is,

$$\begin{aligned} p(\beta_k|\mathbf{z}_{-dn}, \mathbf{w}_{-dn}) &\propto p(\mathbf{w}_{-dn}|\beta_k, \mathbf{z}_{-dn}) p(\beta_k) \\ &= \prod_{k=1}^{K} \prod_{i=1, i\neq w_{dn}}^{N} \beta_{k,i}^{\delta(z_{dn},k)} \times \frac{\Gamma(\sum_{k=1}^{K} \eta)}{\prod_{i=1}^{k} \Gamma(\eta)} \prod_{i=1}^{N} \beta_{k,i}^{\eta-1} \end{aligned} \tag{3.7}$$

Chapter 4

Recommendation System

4.1 Matrix Factorization

4.1.1 Probabilistic Matrix Factorization

Chapter 5

Neural Network and Deep Learning

5.1 Neural Network Notation

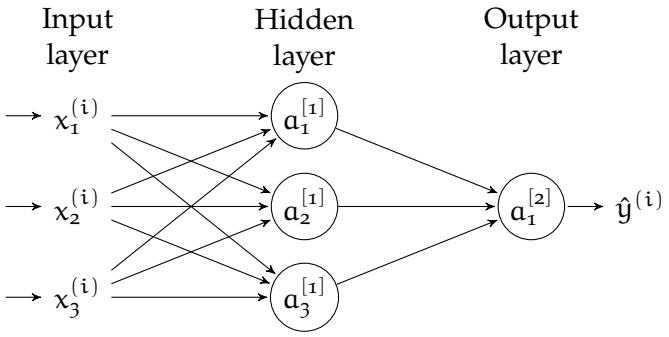

Figure 5.1: A simple representation of 2-layer neural network

A simple two-layer neural network is shown in Figure 5.1, the superscript (i) denotes the i^{th} training example while the superscript $[l]$ denotes the l^{th} layer.

- $x^{(n)}$: the n^{th} input training example
- $g^{[l]}$: the activation function g at l^{th} hidden layer
- $a_n^{[l]}$: the n^{th} activating value at l^{th} hidden layer
- $\hat{y}^{(i)}$: the predicted output vector for the i^{th} training example x

5.2 Forward and Backward Propagation

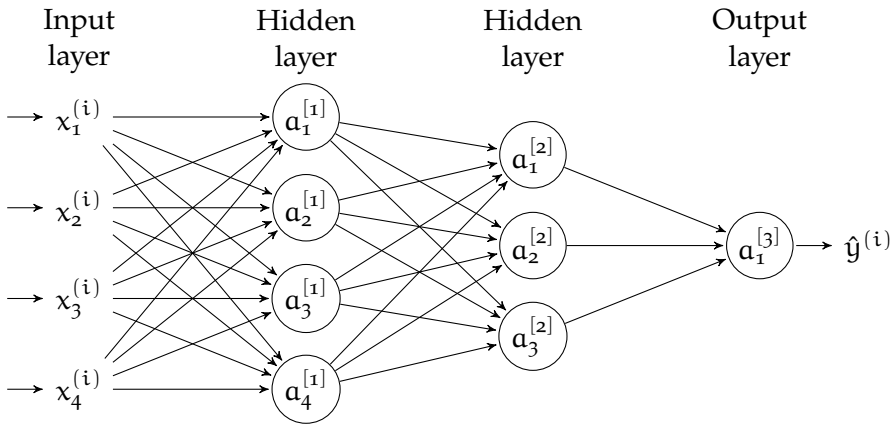

Figure 5.2: A representation of 3-layer neural network

We will use two-layer neural network structure shown in Figure 5.2 to derive forward and backward propagation of neural network. Supposing the activation function $g(\dot{})$ for all hidden layers is $sigmoid(\dot{})$ function for binary classification, and using single training example.

Forward Propagation: predict output label $\hat{y}$ for each training example x, and compute loss (or error) based on true label.

1. At 1^{th} hidden layer,

$$z^{[1]} = W^{[1]}x + b^{[1]}$$
$$a^{[1]} = sigmoid(z^{[1]})$$

2. At 2^{th} hidden layer,

$$z^{[2]} = W^{[2]}a^{[1]} + b^{[2]}$$
$$a^{[2]} = sigmoid(z^{[2]})$$

3. At output layer,

$$z^{[3]} = W^{[3]}a^{[2]} + b^{[3]}$$
$$\hat{y} = a^{[3]} = sigmoid(z^{[3]})$$

4. Compute loss,

$$\mathcal{L}(a^{[3]} - y) = -y\log(a^{[3]}) - (1-y)\log(1-a^{[3]})$$

Backward Propagation: compute the derivative the loss with respect to the weights and bias.

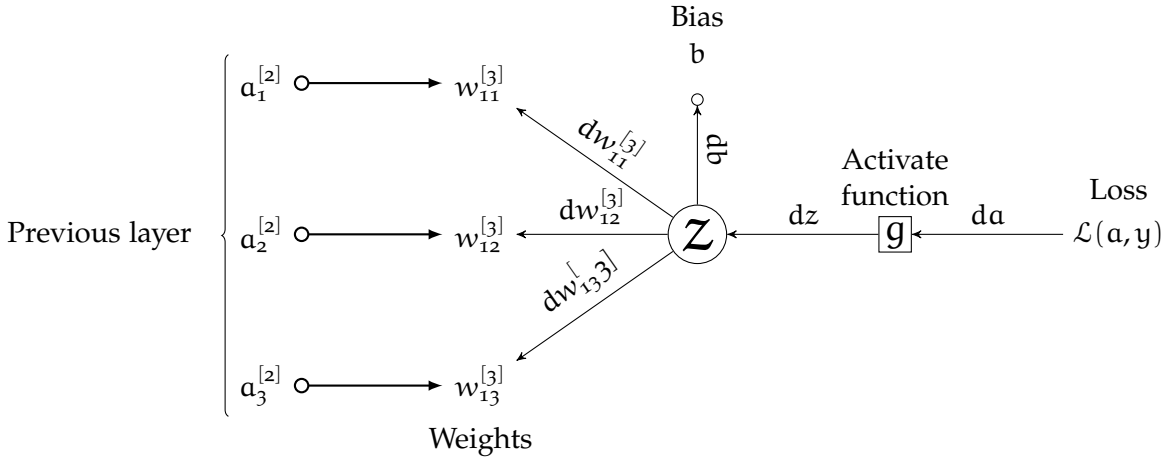

Figure 5.3: Demonstration of the example of backward propagation at output layer (the 3^{rd} layer)

1. At output layer, compute derivative of loss respect to the weights and bias

$$
\begin{aligned}
da^{[3]} &= \frac{\partial \mathcal{L}}{\partial a^{[3]}} = -\frac{y}{a^{[3]}} + \frac{1-y}{1-a^{[3]}} \\
dz^{[3]} &= da^{[3]} \cdot \frac{\partial g}{\partial z^{[3]}} = da^{[3]} \cdot \sigma(z^{[3]})(1-\sigma(z^{[3]}) = da^{[3]} \cdot a^{[3]}(1-a^{[3]}) \\
db^{[3]} &= dz^{[3]} \\
dW^{[3]} &= dz^{[3]} \cdot a^{[2]} \\
da^{[2]} &= dz^{[3]} \cdot (W^{[3]})^T
\end{aligned}
$$

2. At 2^{th} hidden layer,

5.3 Loss Function

Chapter 6

Energy-Based Models

The main purpose of statistical modeling and machine learning is to encode dependencies between variables (observed variables and latent variables or observed variables and target variables), which cause two basic algorithms in machine learning: *Inference* and *Learning*. Energy-Based Models (EBMs) capture dependencies by associating a scalar energy (a measure of compatibility) to each configuration of the variables [2].

Chapter 7

Restricted Boltzmann Machine

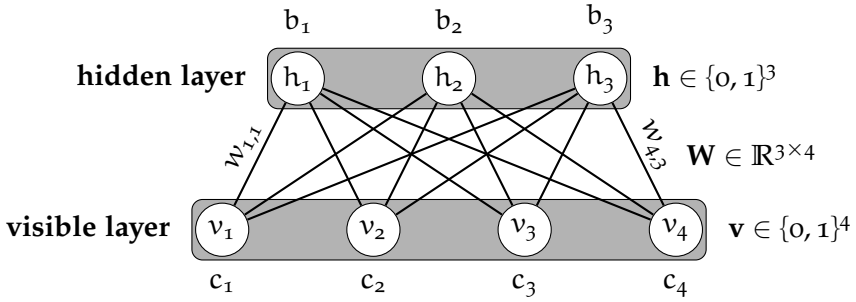

Figure 7.1: The restricted Boltzmann machine itself is an undirected graphical model based on a bipartite graph, with visible units in one part of the graph and hidden units in the other part. There are no connections among the visible units, nor any connections among the hidden units.

The Figure 7.1 presents a restricted Boltzmann machine with 4 visible units ($\mathbf{v}$) and 3 hidden units ($\mathbf{h}$), and $\mathbf{c}, \mathbf{b}$ are bias terms for visible layer and hidden layer respectively. The energy function defined in restricted Boltzmann machine is a joint configuration ($\mathbf{v}$, $\mathbf{h}$) of visible and hidden units,

$$\begin{aligned}\mathbf{E}(\mathbf{v},\mathbf{h}) &= -\mathbf{c}^\top\mathbf{v} - \mathbf{b}^\top\mathbf{h} - \mathbf{h}^\top\mathbf{W}\mathbf{v} \\ &= -\sum_{i\in visible} c_i v_i - \sum_{j\in hidden} b_j h_j - v_i h_j w_{ij}\end{aligned}$$

And the probability of joint configuration via energy function,

$$p(\mathbf{v},\mathbf{h}) = \frac{1}{Z} e^{-\mathbf{E}(\mathbf{v},\mathbf{h})}$$

where the Z is "partition function" given by,

$$Z = \sum_{\mathbf{v},\mathbf{h}} e^{-\mathbf{E}(\mathbf{v},\mathbf{h})}$$

And the "partition function" are normally intractable to compute, because it needs exponential sum all possible visible and hidden units. And the probability of visible units in

RBM is computed by summing over all hidden units,

$$p(\mathbf{v}) = \sum_{j \in hidden} p(\mathbf{v}, h_j)$$

7.1 Model Learning in Restricted Boltzmann Machine

Given N training examples or visible unites $\mathbf{v}$, the learning procedure is to maximize $\prod_{i=1}^{N} p(v_i)$. In other words, it's to minimize negative log-likelihood of $-\sum_{i=1}^{N} \log p(v_i)$. Basically, we compute derivative of $-\log p(v_i)$ respect to learning parameters (e.g., weights, and biases) $\theta = \{\mathbf{W}, \mathbf{b}, \mathbf{c}\}$ in RBM.

Suppose, there are N visible units and H hidden units in RBM. Then, the log likelihood function $\mathbf{L}(\theta)$ is defined as,

$$\begin{aligned}
\mathbf{L}(\theta) &= \sum_{i=1}^{N} \log(p(v_i)) \\
&= \sum_{i=1}^{N} \log(\sum_{\mathbf{h}} p(v_i, \mathbf{h})) \\
&= \sum_{i=1}^{N} \log(\sum_{\mathbf{h}} \frac{exp(-\mathbf{E}(v_i, \mathbf{h}))}{\mathbf{Z}}) \\
&= \sum_{i=1}^{N} \log(\sum_{\mathbf{h}} exp(-\mathbf{E}(v_i, \mathbf{h}))) - N\log(\mathbf{Z}) \\
&= \sum_{i=1}^{N} \log(\sum_{\mathbf{h}} exp(-\mathbf{E}(v_i, \mathbf{h}))) - N\log(\sum_{\mathbf{h},\mathbf{v}} exp(-\mathbf{E}(\mathbf{v}, \mathbf{h})))
\end{aligned}$$

Then, the derivative of $\mathbf{L}(\theta)$ respect to parameter θ is,

$$\begin{aligned}
\frac{\partial \mathbf{L}(\theta)}{\partial \theta} =& \frac{\partial\Big(\sum_{i=1}^{N} \log(\sum_{\mathbf{h}} exp(-\mathbf{E}(v_i, \mathbf{h})))\Big)}{\partial \theta} - \frac{\partial\Big(N\log(\sum_{\mathbf{h},\mathbf{v}} exp(-\mathbf{E}(\mathbf{v}, \mathbf{h})))\Big)}{\partial \theta} \\
=& \sum_{i=1}^{N} \frac{1}{\sum_{\mathbf{h}} exp(-\mathbf{E}(v_i, h))} \frac{\partial\Big(\sum_{\mathbf{h}} exp(-\mathbf{E}(v_i, \mathbf{h}))\Big)}{\partial \theta} \\
& - N \frac{1}{\sum_{\mathbf{h},\mathbf{v}} exp(-\mathbf{E}(\mathbf{v}, \mathbf{h}))} \frac{\partial\Big(\sum_{\mathbf{h},\mathbf{v}} exp(-\mathbf{E}(\mathbf{v}, \mathbf{h}))\Big)}{\partial \theta} \\
=& \sum_{i=1}^{N} \frac{1}{\sum_{\mathbf{h}} exp(-\mathbf{E}(v_i, \mathbf{h}))} \sum_{\mathbf{h}} \Big\{ exp(-\mathbf{E}(v_i, \mathbf{h})) \frac{\partial - \mathbf{E}(v_i, \mathbf{h})}{\partial \theta} \Big\} \\
& - N \frac{1}{\sum_{\mathbf{h},\mathbf{v}} exp(-\mathbf{E}(\mathbf{v}, \mathbf{h}))} \sum_{\mathbf{h},\mathbf{v}} \Big\{ exp(-\mathbf{E}(\mathbf{v}, \mathbf{h})) \frac{\partial - \mathbf{E}(\mathbf{v}, \mathbf{h})}{\partial \theta} \Big\} \\
=& \sum_{i=1}^{N} \sum_{\mathbf{h}} \frac{exp(-\mathbf{E}(v_i, \mathbf{h}))}{\sum_{\mathbf{h}} exp(-\mathbf{E}(v_i, h))} \frac{\partial - \mathbf{E}(v_i, \mathbf{h})}{\partial \theta} - N \sum_{\mathbf{h},\mathbf{v}} \frac{exp(-\mathbf{E}(\mathbf{v}, \mathbf{h}))}{\sum_{\mathbf{h},\mathbf{v}} exp(-\mathbf{E}(\mathbf{v}, \mathbf{h}))} \frac{\partial - \mathbf{E}(\mathbf{v}, \mathbf{h})}{\partial \theta}
\end{aligned}$$

$$
\begin{aligned}
&= \sum_{i=1}^{N} \sum_{\mathbf{h}} p(v_i|\mathbf{h}) \frac{\partial - \mathbf{E}(v_i, \mathbf{h})}{\partial \theta} - N \sum_{\mathbf{h},\mathbf{v}} p(\mathbf{v}, \mathbf{h}) \frac{\partial - \mathbf{E}(\mathbf{v}, \mathbf{h})}{\partial \theta} \\
&= \sum_{i=1}^{N} \mathbb{E}_{\mathbf{h}|v_i} \Big[\frac{\partial - \mathbf{E}(v_i, \mathbf{h})}{\partial \theta} \Big] - N \mathbb{E}_{\mathbf{v},\mathbf{h}} \Big[\frac{\partial - \mathbf{E}(\mathbf{v}, \mathbf{h})}{\partial \theta} \Big]
\end{aligned}
$$

Bibliography

[1] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Mar. 2003.

[2] Y. LeCun, S. Chopra, R. Hadsell, M. Ranzato, and F. Huang, "A tutorial on energy-based learning," *Predicting structured data*, vol. 1, no. 0, 2006.