SVI

1. Frequentist & Bayesian

Frequentist: Maximum Likelihood Estimation

$$\begin{aligned}\theta_{ML} &= \arg\max p(x|\theta)\\ &= \arg\max \prod_{i=1}^{n} p(x_i|\theta)\\ &= \arg\max \sum_{i=1}^{n} \log p(x_i|\theta)\end{aligned}$$

Bayesian:

$$\begin{aligned}P(\theta|x) &= \frac{P(x|\theta)P(\theta)}{P(x)}\\ &= \frac{\prod_{i=1}^{n} p(x_i|\theta)p(\theta)}{\int \prod_{i=1}^{n} p(x_i|\theta)p(\theta)d\theta}\end{aligned}$$

2. Full bayesian inference:

train:

$$p(\theta|x_{\text{true}}, y_{\text{true}}) = \frac{p(y_{\text{true}}|x_{\text{true}}, \theta)p(\theta)}{\int p(y_{\text{true}}|x_{\text{true}}, \theta)p(\theta)d\theta}$$

test:

$$p(y|x, x_{\text{true}}, y_{\text{true}}) = \int p(y|x,\theta)p(\theta|x_{\text{true}}, y_{\text{true}})d\theta$$

For conjugate priors we can find analytical solutions and for low-dimensional problems we can perform numerical integration directly.

However, when faced with high-dimensional problems, the processing of this marginal distribution P(x) becomes difficult, making the posterior distribution difficult to solve.

VI (Variational Inference)

Then a solution is to use the simple distribution $q(\theta)$ to fit the complex posterior distribution $p(\theta|x)$. Then we can use KL divergence to measure the two distributions:

$$
\begin{aligned}
q^*(\theta) &= \arg\min_{q(\theta)} KL(q(\theta)||p(\theta|x)) \\
&= \arg\min_{q(\theta)} \int q(\theta) \log \frac{q(\theta)}{p(\theta|x)} d\theta
\end{aligned}
$$

However, there is still a posterior distribution in the optimization objective, so we need to make some transformations:

$$
\begin{aligned}
\int q(\theta) \log \frac{q(\theta)}{p(\theta|x)} d\theta &= \int q(\theta) \log \frac{q(\theta)p(x)}{p(x|\theta)p(\theta)} d\theta \\
&= \int q(\theta) \log \frac{q(\theta)p(x)}{p(x,\theta)} d\theta \\
&= \int q(\theta) \log \frac{q(\theta)}{p(x,\theta)} d\theta + \int q(\theta) \log p(x) d\theta
\end{aligned}
$$

We can get:

$$
\begin{aligned}
\mathbb{E}_q[\log p(x)] &= KL(q(\theta)||p(\theta|x)) + \int q(\theta) \log \frac{p(x,\theta)}{q(\theta)} d\theta \\
\rightarrow \log p(x) &= KL(q(\theta)||p(\theta|x)) + \mathcal{L}(q(\theta))
\end{aligned}
$$

We cannot solve $KL(q(\theta)||p(\theta|x))$explicitly, but because the KL divergence is always greater than or equal to 0. So we can maximize ELBO (evidence lower bound). Of course ELBO can also be further expressed as:

$$\begin{aligned}\mathcal{L}(q(\theta)) &= \int q(\theta)\log\frac{p(x,\theta)}{q(\theta)}d\theta \\ &= \int q(\theta)\log\frac{p(x|\theta)p(\theta)}{q(\theta)}d\theta \\ &= \mathbb{E}[\log(p(x|\theta))] - KL(q(\theta)||p(\theta))\end{aligned}$$

It can be seen that the first term of EBLO is equal to the expectation of likelihood, and the second term hopes that $q(\theta)p(\theta)$.

Mean field approximation

Then for maximizing ELBO, a simple way is to use the mean field approximation method, which decomposes $q(\theta)$ into multiple independent latent variables. So it can be expressed as

$$q(\theta) = \prod q_i(\theta_i)$$

And use CAVI (coodinate ascent variational inference) method to solve the problem. However, the problem with this method is that it is too hypothetical. It assumes that all variables are independent of each other, which cannot reflect the relationship between hidden variables. Additionally, this method underestimates the variance of the true distribution.

CAVI

This method only iterates one hidden variable in each calculation, and the other hidden variables are fixed.

$$\begin{aligned}\mathcal{L}(q(\theta)) &= \int_{j=1\ldots n}\prod q_j(\theta_j)\log p(x,\theta)d\theta - \int_{j=1\ldots n}\prod q_j(\theta_j)\log\prod q(\theta_j)d\theta \\ &= \mathbb{E}_{j=i}\mathbb{E}_{\prod_{j\neq i}}[\log p(x,\theta)] - \mathbb{E}_i[\log q_j(\theta_j)] + \mathbb{C}\end{aligned}$$

$$\begin{aligned}&\int_{j=1...n}\prod q_j(\theta_j)\log p(x,\theta)d\theta\\=&\int_{j=i}q_i(\theta_i)[\int_{j\neq i}\prod q_j(\theta_j)\log p(x,\theta)d\theta_j]d\theta_i\\=&\mathbb{E}_{j=i}\mathbb{E}_{\prod_{j\neq i}}[\log p(x,\theta)]\end{aligned}$$

$$\begin{aligned}&\int_{j=1...n}\prod q_j(\theta_j)\log\prod q(\theta_j)d\theta\\=&\int_{j=1...n}\prod q_j(\theta_j)\sum\log q(\theta_j)d\theta\\=&\sum\int_j q_j(\theta_j)\log q(\theta_j)d\theta_j\\=&\mathbb{E}_j[\log q_j(\theta_j)]+\mathbb{C}\end{aligned}$$

By iterating one at a time, it can eventually converge to a local optimal solution.

SVI (Stochastic Variational Inference)

For large-scale data, the SVI method can be used.

Our objective function is:

$$\begin{aligned}q(\theta)^* &= \arg\min KL(q(\theta)||p(\theta|x))\\&=\arg\max\mathcal{L}(q(\theta))\end{aligned}$$

Here we choose a parameterized distribution family to represent $q(\theta)$,This distribution is controlled by a set of variational parameters ϕ, so that our goal changes from seeking the optimal distribution to seeking the optimal distribution. parameters.

$$\begin{aligned}\phi^* &= \arg\max_{\phi}\mathcal{L}(\phi)\\&=\mathbb{E}_{q_\phi(\theta)}[\log p(x,\theta)]-\mathbb{E}_{q_\phi(\theta)}[\log q_\phi(\theta)]\end{aligned}$$

So we can find the gradient:

$$
\begin{aligned}
\nabla_\phi \mathcal{L}(\phi) &= \nabla_\phi (\mathbb{E}_{q_\phi(\theta)}[\log p(x,\theta)] - \mathbb{E}_{q_\phi(\theta)}[\log q_\phi(\theta)]) \\
&= \nabla_\phi \int q_\phi(\theta)(\log p(x,\theta) - \log q_\phi(\theta))d\theta \\
&= \int \underbrace{\nabla_\phi q_\phi}_{=q_\phi(\theta)\nabla_\phi \log q_\phi(\theta)(} (\theta)(\log p(x,\theta) - \log q_\phi(\theta))d\theta \\
&+ \int q_\phi(\theta)\nabla_\phi(\log p(x,\theta) - log q_\phi(\theta))d\theta \\
&= \int q_\phi(\theta)\nabla_\phi \log q_\phi(\theta)(\log p(x,\theta) - \log q_\phi(\theta))d\theta \\
&- \underbrace{\int q_\phi(\theta)\nabla_\phi \log q_\phi(\theta)d\theta}_{=\nabla_\phi \int q_\phi(\theta)d\theta = 0} \\
&= \mathbb{E}_{q_\phi(\theta)}[\nabla_\phi \log q_\phi(\theta)(\log p(x,\theta) - \log q_\phi(\theta)]
\end{aligned}
$$

So we can use the Monte Carlo method to approximate the gradient, and then use the stochastic gradient descent method to optimize the parameters.

$$
\begin{aligned}
\nabla_\phi \mathcal{L}(\phi) &= \mathbb{E}_{q_\phi(\theta)}[\nabla_\phi \log q_\phi(\theta)(\log p(x,\theta) - \log q_\phi(\theta)] \\
&= \frac{1}{L}\sum_{i=1}^{L}(\nabla_\phi \log q_\phi(\theta)(\log p(x,\theta) - \log q_\phi(\theta)) \\
\phi^{t+1} &= \phi^t + \lambda \nabla_\phi \mathcal{L}(\phi)
\end{aligned}
$$

However, $q_\phi(\theta)$ $q_\phi(\theta)\log q_\phi(\theta)$ will appear. Extremely high values lead to high variance. One solution is to use reparameterization techniques.

Reparameterization Trick

The expectation of the random variable $q(\theta)p(\epsilon)$ using the reparameterization technique, that is $\int q_\phi(\theta)d\theta = 1 = \int p(\epsilon)d\epsilon, \theta = g_\theta(\epsilon, x)$.

$$
\begin{aligned}
\nabla_\phi \mathcal{L}(\phi) &= \nabla_\phi \int q_\phi(\theta)(\log p(x,\theta) - \log q_\phi(\theta) d\theta \\
&= \nabla_\phi \int p(\epsilon)(\log p(x,\theta) - \log q_\phi(\theta) d\epsilon \\
&= \mathbb{E}_{p(\epsilon)}[\nabla_\phi(\log p(x,\theta) - \log q_\phi(\theta)] \\
&= \mathbb{E}_{p(\epsilon)}[\nabla_\theta(\log p(x,\theta) - \log q_\phi(\theta|x)\nabla_\phi\theta] \\
&= \mathbb{E}_{p(\epsilon)}[\nabla_\theta(\log p(x,\theta) - \log q_\phi(\theta|x)\nabla_\phi g_\phi(\epsilon,x)]
\end{aligned}
$$

so the pipeline is:

1. Initialize the parameters ϕ randomly
2. Sample a set of ϵ from the noise distribution $p(\epsilon)$
3. computer $\theta = g_\phi(\epsilon, x)$
4. computer gradietn
5. update ϕ