VI Notes

Jinqi Luo

November 2019

1 VI Motivation

Data: Univariate Normal Distribution with unknown parameters μ, λ, which are to infer.

$$X \sim N(\mu, \lambda^{-1}) \tag{1}$$

Define the sample as $D = \{x_1, ..., x_N\}$:

$$P(D|\mu, \lambda) = \prod_{i=1}^{N} N(x_i|\mu, \lambda^{-1}) \tag{2}$$

Use a normal-gamma distribution as prior for μ and λ:

$$\begin{align} P(\mu, \lambda) &= P(\mu|\lambda) \cdot P(\lambda) \tag{3} \\ &= N(\mu|\mu_0, (\kappa_0\lambda)^{-1}) \cdot \Gamma(\lambda|a_0, b_0) \tag{4} \end{align}$$

To be specific:

$$P(\mu, \lambda) = exp(\frac{-k_0\lambda}{2}(\mu - \mu_0)^2) \cdot \lambda^{a_0-1}e^{-b_0\lambda} \tag{5}$$

The hyperparameters μ_0, κ_0, a_0, and b_0 are prior knowledge about μ and λ:

$$\begin{aligned} E[\mu] = \mu_0 \quad Var[\mu] = \frac{b_0}{\kappa_0(a_0 - 1)} \\ E[\lambda] = \frac{a_0}{b_0} \quad Var[\lambda] = \frac{a_0}{b_0^2} \end{aligned}$$

the posterior $P(\mu, \lambda|D)$ is:

$$P(\mu, \lambda|D) = \frac{P(\mu, \lambda, D)}{P(D)} \tag{6}$$
$$= \frac{P(D|\mu, \lambda)P(\mu, \lambda)}{P(D)} \tag{7}$$
$$\propto P(D|\mu, \lambda)P(\mu, \lambda) \tag{8}$$

Since the normal-gamma prior is conjugate to a normal distribution, the posterior $P(\mu, \lambda|D)$ follows the prior as a normal-gamma distribution.

$$P(D|\mu, \lambda)P(\mu, \lambda) \tag{9}$$
$$= \prod_{i=1}^{N} N(x_i|\mu, \lambda^{-1})N(\mu|\mu_0, (\kappa_0\lambda)^{-1})\Gamma(\lambda|a_0, b_0) \tag{10}$$
$$= N(\mu|\mu_N, (\kappa_N\lambda)^{-1})\Gamma(\lambda|a_N, b_N) \tag{11}$$

Because of conjugacy, the exact solution for posterior can be found where:

$$\mu_N = \frac{\kappa_0\mu_0 + N\bar{X}}{\kappa_0 + N} \tag{12}$$
$$\kappa_N = \kappa_0 + N \tag{13}$$
$$a_N = a_0 + N/2 \tag{14}$$
$$b_N = b_0 + \frac{1}{2}\sum_{i=1}^{N}(x_i - \bar{X})^2 + \frac{\kappa_0 n(\bar{X} - \mu_0)^2}{2(\lambda_0 + N)} \tag{15}$$

So Variational Inference here is verifiable and on-the-call to approximate $P(\mu, \lambda|D)$ by a simpler distribution $Q(\mu, \lambda)$.

2 VI for our Dataset

Firstly, it is important to figure out the loss funtion $J(Q(Z))$:

$$J(Q(Z)) \propto Distance(Q(Z)||P(X)) \tag{16}$$
$$Distance(B(x)||A(x)) = \int_x B(x) \log \frac{B(x)}{A(x)} dx \tag{17}$$

In our case, let's say the posterior Q is a fully factorized approximation and each latent variable in the factorization is independent (Mean-field). Then:

$$
\begin{aligned}
Q(Z) &= \prod_i q_i(z_i) && (18) \\
&= q_\mu(\mu) \cdot q_\lambda(\lambda) && (19)
\end{aligned}
$$

According to the slides, each component in Q, i.e. q_μ and q_λ, will be optimized when:

$$\log q_i(z_i) \propto E_{-q_i}[\log P(X)] \quad (20)$$

where $E_{-q_i}[P(X)]$ is the expectation of $P(X)$ using all factors in $Q(Z)$ excepts the factor itself $q_i(z_i)$.

In our case:

$$
\begin{aligned}
Q(Z) &= q_\mu(\mu) \cdot q_\lambda(\lambda) && (21) \\
P(X) &= P(\mu, \lambda, D) && (22)
\end{aligned}
$$

and thus:

$$
\begin{aligned}
\log q_\lambda(\lambda) &= E_{q_\mu}[\log P(\mu, \lambda, D)] && (23) \\
\log q_\mu(\mu) &= E_{q_\lambda}[\log P(\mu, \lambda, D)] && (24)
\end{aligned}
$$

By solving $\log q_\mu(\mu) = E_{q_\lambda}[\log P(\mu, \lambda, D)]$, it can be shown that

$$
\begin{aligned}
q_\mu(\mu) &= N(\mu|\mu_N, \kappa_N^{-1}) && (25) \\
\mu_N &= \frac{\kappa_0\mu_0 + N\bar{x}}{\kappa_0 + N} && (26) \\
\kappa_N &= (\kappa_0 + N)\frac{a_N}{b_N} && (27)
\end{aligned}
$$

By solving $\log q_\lambda(\lambda) = E_{q_\mu}[\log P(\mu, \lambda, D)]$, it can be shown that

$$q_\lambda(\lambda) = \text{Ga}(\lambda|a_N, b_N) \tag{28}$$

$$a_N = a_0 + \frac{N+1}{2} \tag{29}$$

$$b_N = b_0 + \kappa_0(\frac{1}{\kappa_N} + \mu_N^2 + \mu_0^2 - 2\mu_N\mu_0) + \frac{1}{2}\sum_{i=1}^{N}(x_i^2 + \frac{1}{\kappa_N} + \mu_N^2 - 2\mu_N x_i) \tag{30}$$

We can iteratively update two components of the Q distribution starting from a given initial state on μ_0, κ_0, a_0, b_0 from $P(\mu, \lambda)$, and then achieve a good approximation for it.

3 Python Demo

4 Reference

The book (starting from Page 742):
Machine Learning: a Probabilistic Perspective

Other materials that I went through:
[SydneyTech Lectures] Variational Inference Basics
https://github.com/roboticcam/machine-learning-notes/blob/master/files/variational.pdf
https://github.com/blei-lab/edward
[Jordan Boyd-Graber] Machine Learning: Variational Inference
https://github.com/vincentadam87/MFVI
[NIPS 2016 tutorial] Variational Inference: Foundations and Modern Methods
https://en.wikipedia.org/wiki/Gamma-distribution
https://en.wikipedia.org/wiki/Conjugate-prior