1 Notations

1.1 Model & Data

Let $f(x;\theta)$ a function of x with parameter θ. Let $y = f(x;\theta)$ an observable, thus the observed value obeys a Gaussian distribution. Let D denotes a set of observations, $D := \{(x_i, y_i, \sigma_i) : i = 1, ..., N_D\}$, wherein x_i is the ith input, y_i its observed value, and σ_i the observational error of y_i. We may employ mini-batch technique, thus denote $D_m := \{(x_i, y_i, \sigma_i) : i = 1, ..., N_m\} \subset D$ as a mini-batch, with batch-size $N_m \leqslant N_D$. We use $\mathbb{E}_{f(\theta)}[g(\theta)]$ represent the expectation of function g of a random variable obeys the p.d.f. f. Φ is for Gaussian p.d.f..

2 Bayesian

2.1 Prior-Posterior Iteration

2.2 Bayesian as Information Encoder

Comparing with the traditional method, what is the advantage of Bayesian way? The answer is, it encodes more information of data into model. Indeed, it does not encodes the value of peak of the posterior only, as traditional method does, but also much more information on the posterior. XXX

3 Neural Network for Posterior (nn4post)

3.1 The Model

Suppose we have some prior on θ, $p(\theta)$, we gain the unnormalized posterior $p(D|\theta)\, p(\theta)$. With D arbitrarily given, this unnormalized posterior is a function of θ, denoted by $p(\theta; D)$[1].[2]

We we are going to do is fit this $p(\theta; D)$ by ANN for any given D. To do so, we have to assume that $\mathrm{supp}\{p(\theta; D)\} = \mathbb{R}^d$ for some $d \in \mathbb{N}^+$ (i.e. has no compact support) but decrease exponentially fast as $\|\theta\| \to +\infty$. With this assumption, $\ln p(\theta; D)$ is well-defined. For ANN, we propose using Gaussian function as the activation-function. Thus, we have the fitting function

$$q(\theta; a, \mu, \zeta) := \sum_{i=1}^{N_c} c_i(a) \left\{ \prod_{\alpha=1}^{d} \Phi(\theta_\alpha - \mu_{i\alpha}, \sigma(\zeta_{i\alpha})) \right\}, \tag{1}$$

where

$$c_i(a) = \frac{\exp(a_i)}{\sum_{j=1}^{N_c} \exp(a_j)} = \mathrm{softmax}(i; a); \tag{2}$$

$$\sigma(\zeta_{i\alpha}) = \ln(1 + \exp(\zeta_{i\alpha})), \tag{3}$$

and $a_i, \mu_{i\alpha}, \zeta_{i\alpha} \in \mathbb{R}$ for $\forall i, \forall \alpha$ and

$$\Phi(x; \mu, \sigma) := \sqrt{\frac{1}{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \tag{4}$$

being the Gaussian p.d.f.. The introduction of ζ is for numerical consideration, see below.

---

1. This is why we use " ; " instead of " , ", indicating that D has been (arbitrarily) given and fixed.

2. The normalized posterior $p(\theta|D) = p(D|\theta)\, p(\theta)/p(D) = p(\theta; D)/p(D)$, by Bayes's rule.

3.1.1 Numerical Consideration

If, in q, we regard w, μ, and σ as independent variables, then the only singularity appears at $\sigma = 0$. Indeed, σ appears in Φ (as well as the derivatives of Φ) as denominator only, while others as numerators. However, once doing numerical iterations with a finite step-length of σ, the probability of reaching or even crossing 0 point cannot be surely absent. This is how we may encounter this singularity in practice.

Introducing the ζ is our trick of avoiding this singularity. Precisely, using a singular map that pushes the singularity to infinity solves the singularity. In this case, using softplus(.) that pushes $\sigma = 0$ to $\zeta \to -\infty$, so that, with finite steps of iteration, singularity (at $-\infty$) cannot be reached.

This trick (i.e. pushing a singularity to infinity) is the same as in avoiding the horizon-singularity of Schwarzschild solution of black hole.

3.2 Interpretation

3.2.1 As a Mixture Distribution

$q(\theta; a, \mu, \zeta)$ has a probabilistic interpretation. $\prod_{j=1}^{d} \Phi(\theta_j - \mu_{ij}, \sigma(\zeta_{ij}))$ corresponds to multi-dimensional Gaussian distribution (denote $\mathcal{N}$), with all dimensions independent with each other. The $\{c_i(a)\}$ is a categorical distribution, randomly choosing the Gaussian distributions. Thus $q(\theta; a, \mu, \zeta)$ is a composition: categorical $\to$ Gaussian. This is the *mixture distribution*.

3.2.2 As a Generalization

This model can also be interpreted as a direct generalization of mean-field variational inference. Indeed, let $N_c = 1$, this model reduces to mean-field variational inference. Remark that mean-field variational inference is a mature algorithm and has been successfully established on many practical applications.

3.2.3 As a Neural Network

3.3 Marginalization

This model can be marginalized easily. This then benefits the transfering of the model components. Precisely, for any dimension-index β given, we can marginalize all other dimensions directly, leaving

$$q(\theta_\beta; a, \mu, \zeta) \;=\; \prod_{\forall \gamma \neq \beta} \int \mathrm{d}\theta_\gamma \sum_{i=1}^{N_c} c_i(a) \left\{ \prod_{\alpha=1}^{d} \Phi(\theta_\alpha; \mu_{i\alpha}, \sigma(\zeta_{i\alpha})) \right\} \tag{5}$$

$$=\; \sum_{i=1}^{N_c} c_i(a) \, \Phi(\theta_\beta; \mu_{i\beta}, \sigma(\zeta_{i\beta})), \tag{6}$$

where employed the normalization of Φ.

3.4 Loss-Function

We employ "evidence of lower bound" (ELBO)[3]. It is ensured to have a unique global maximum, at which $p(\theta; D) = q(\theta; a, \mu, \zeta)$.

$$\mathrm{ELBO}(a, \mu, \zeta) := \mathbb{E}_{\theta \sim q(\theta; a, \mu, \zeta)}[\ln p(\theta; D) - \ln q(\theta; a, \mu, \zeta)].$$

---

3. The relation between ELBO and KL-divergence is that $\mathrm{ELBO} = -\mathrm{KL}(q \| p) + \mathrm{Const}$.

Since there's no compact support for both $p(\theta; D)$ and $q(\theta; a, \mu, \zeta)$, ELBO is well-defined. The loss-function (or say loss-function, performance, etc) of the fitting is then defined as $\mathcal{L} = -\text{ELBO}$, i.e.

$$\mathcal{L}(a, \mu, \zeta) = -\mathbb{E}_{\theta \sim q(\theta; a, \mu, \zeta)}[\ln p(\theta; D) - \ln q(\theta; a, \mu, \zeta)],$$

or, recall $\mathbb{H}[q] := -\mathbb{E}_q[\ln q]$ for any distribution q,

$$\mathcal{L}(a, \mu, \zeta) = -\mathbb{E}_{\theta \sim q(\theta; a, \mu, \zeta)}[\ln p(\theta; D)] - \mathbb{H}[q(\theta; a, \mu, \zeta)].$$

4 Optimization

4.1 ADVI

Automatic differentation variational inference (ADVI)[4] has the advantage that the variance of its Monte Carlo integral is orderly smaller than that of black box variational inference (i.e. optimization directly using ELBO without further reparameterization).

4.1.1 Derivation

Precisely, recall $\mathbb{E}$ for mean value, Φ for Gaussian p.d.f., $\sigma(.)$ for softplus function, $c(.)$ for softmax function, and

$$q(\theta; a, \mu, \zeta) = \sum_{i=1}^{N_c} c_i(a) \, \Phi(\theta; \mu_i, \sigma(\zeta_i)), \tag{7}$$

we have, for any function f,

$$\mathbb{E}_{q(\theta; a, \mu, \zeta)}[f(\theta)] = \int \mathrm{d}\theta \sum_{i=1}^{N_c} c_i(a) \, \Phi(\theta; \mu_i, \sigma(\zeta_i)) \, f(\theta) \tag{8}$$

$$= \sum_{i=1}^{N_c} c_i(a) \int \mathrm{d}\theta \, \Phi(\theta; \mu_i, \sigma(\zeta_i)) \, f(\theta) \tag{9}$$

$$= \sum_{i=1}^{N_c} c_i(a) \, \mathbb{E}_{\Phi(\theta; \mu_i, \sigma(\zeta_i))}[f(\theta)]. \tag{10}$$

With this general relation, we get

$$\mathcal{L}(a, \mu, \zeta) = -\{\mathbb{E}_{q(\theta; a, \mu, \zeta)}[\ln p(\theta; D)] - \mathbb{E}_{q(\theta; a, \mu, \zeta)}[\ln q(\theta; a, \mu, \zeta)]\} \tag{11}$$

$$= -\sum_{i}^{N_c} c_i(a) \, \mathbb{E}_{\Phi_i(\theta; \mu_i, \sigma(\zeta_i))}[\ln p(\theta; D) - \ln q(\theta; a, \mu, \zeta)] \tag{12}$$

Then, for $\forall i = 1, ..., N_c$, $\forall \alpha = 1, ..., N_d$, let

$$\eta_\alpha := \frac{\theta_\alpha - \mu_{i\alpha}}{\sigma(\zeta_{i\alpha})}, \tag{13}$$

we have

$$\theta_\alpha = \sigma(\zeta_{i\alpha}) \, \eta_\alpha + \mu_{i\alpha} \tag{14}$$

---

4. See, Kucukelbir, et al, 2016.

(or $\theta = \sigma(\zeta_i)\, \eta + \mu_i$ if hide the α index). So, for any i-component, we transform

$$\theta \;\to\; \sigma(\zeta_i)\, \eta + \mu_i; \tag{15}$$

$$\mathbb{E}_{\Phi(\theta;\mu_i,\sigma(\zeta_i))}[f(\theta)] \;\to\; \mathbb{E}_{\Phi(\eta;0,1)}[f(\sigma(\zeta_i)\, \eta + \mu_i)], \tag{16}$$

where function f is arbitrary, thus holds for both $\ln p(.;D)$ and $\ln q(.;a,\mu,\zeta)$.

With this setting, the derivatives to μ and to ζ are completely independent of $\mathbb{E}[.]$. And now, the loss function becomes

$$\mathcal{L}(a,\mu,\zeta) = -\sum_i^{N_c} c_i(a)\, \mathbb{E}_{\Phi(\eta;0,1)}[\ln p(\sigma(\zeta_i)\, \eta + \mu_i;D) - \ln q(\sigma(\zeta_i)\, \eta + \mu_i;a,\mu,\zeta)].$$

4.2 Redefination of $\partial\mathcal{L}/\partial a$

Let Δr the learning-rate. Then the updation of a_i at one iteration by gradient decent method is

$$\Delta a_i = \frac{\partial\mathcal{L}}{\partial a_i}(a,\mu,\zeta)\, \Delta r.$$

Notice that redefine the $\partial\mathcal{L}/\partial a$ by

$$\frac{\partial\mathcal{L}}{\partial a_i}(a,\mu,\zeta) \to \frac{\partial\mathcal{L}}{\partial a_i}(a,\mu,\zeta) + C,$$

where C can be any constant, leaves the updation of $c_i(a)$ invariant, since it makes

$$\Delta a_i \to \frac{\partial\mathcal{L}}{\partial a_i}(a,\mu,\zeta)\, \Delta r + C\, \Delta r,$$

thus

$$c_i(a + \Delta a) \to \frac{\exp\left(a_i + \Delta a_i + C\, \Delta r\right)}{\sum_j \exp\left(a_j + \Delta a_j + C\, \Delta r\right)} = \frac{\exp\left(a_i + \Delta a_i\right)}{\sum_j \exp\left(a_j + \Delta a_j\right)} = c_i(a + \Delta a).$$

This C thus provides an additional dof. We can tune the value of C so that the updation of a_i is numerically stable. Indeed, let C be the average of $\{\partial\mathcal{L}/\partial a_i: i = 1, ..., N_c\}$, we find a pretty stability of a as well as a pretty accuracy of c in the iteration process of optimization.

4.2.1 Re-scaling of a

4.3 Approximations

Comparing to the traditional MAP approach, using multi-peak mixture model makes the $\mathbb{H}(q)$ complicated, especially in the optimization process.

4.3.1 Entropy Lower Bound

Consider any mixture distribution with p.d.f. $\sum_i^{N_c} c_i\, q_i$ where cs are the categorical probabilities and q_i the p.d.f. of the component distributions of the mixture.

$$\mathbb{H}\left[\sum_i^{N_c} c_i\, q_i\right] \geqslant \sum_i^{N_c} c_i\, \mathbb{H}[q_i].$$

So, if define

$$\mathcal{L}'(a, \mu, \zeta) := -\sum_{i}^{N_c} c_i(a) \left\{ \mathbb{E}_{\Phi(\eta;0,1)}[\ln p(\sigma(\zeta_i)\, \eta + \mu_i; D)] + \mathbb{H}[\Phi(\theta, \mu_i, \sigma(\zeta_i))] \right\}, \tag{17}$$

then we have

$$\mathcal{L}' \geqslant \mathcal{L} \geqslant \min(\mathcal{L}) > -\infty,$$

thus $\mathcal{L}'$ has an global minimum. In this way, the entropy part becomes completely analytic (and simple).

4.3.2 Application

$$\frac{\partial \mathcal{L}'}{\partial a_i}(a, \mu, \zeta) = -c_i(a) \left\{ \mathbb{E}_{\Phi(\theta;\mu_i, \sigma(\zeta_i))}[\ln p(\theta; D)] + \mathbb{H}[\Phi(\theta, \mu_i, \sigma(\zeta_i))] + \mathcal{L}'(a, \mu, \zeta) \right\}; \tag{18}$$

$$\frac{\partial \mathcal{L}'}{\partial \mu_{i\alpha}}(a, \mu, \zeta) = \frac{\partial \mathcal{L}_p}{\partial \mu_{i\alpha}}(a, \mu, \zeta) + 0; \tag{19}$$

$$\frac{\partial \mathcal{L}'}{\partial \zeta_{i\alpha}}(a, \mu, \zeta) = \ldots \tag{20}$$

4.4 Stochastic Optimization

4.4.1 Difference between Bayesian and Traditional Methods

Suppose, instead of use the whole dataset, we employ mini-batch technique. Since all data are independent, if suppose that D_m is unbiased in D, then we have,

$$\ln p(D|\theta) = \sum_D p((x_i, y_i, \sigma_i)|\theta) \approx \frac{N_D}{N_m} \sum_{D_m} p((x_i, y_i, \sigma_i)|\theta) = \frac{N_D}{N_m} \ln p(D_m|\theta). \tag{21}$$

Then,

$$\ln p(\theta; D) = \ln p(D|\theta) + \ln p(\theta) = \frac{N_D}{N_m} \ln p(D_m|\theta) + \ln p(\theta), \tag{22}$$

thus as previous

$$\ln p(\theta; D) = \frac{N_D}{N_m} \sum_{(x_i, y_i, \sigma_i) \in D_m} \left\{ -\frac{1}{2} \ln(2\pi\sigma_i^2) - \frac{1}{2} \left(\frac{y_i - f(x_i; \theta)}{\sigma_i} \right)^2 \right\} + \ln p(\theta). \tag{23}$$

In this we meet one of the main differences between the Bayesian and the traditional. In the traditional method, N_D does not matters in training, being absent in the optimizer. However, in Bayesian, the number of data that are employed is encoded into Bayesian model, and has to, since the greater number of data gives more confidence. So, while using stochastic optimization in Bayesian mode, the factor N_D/N_m of likelihood has to be taken into account. We have to know how many data we actrually have, thus how confident we are.

5 Deep Learning

It cannot solve the vanishing gradient problem of deep neural network, since this problem is intrinsic to the posterior of deep neural network. Indeed, the posterior has the shape like $\exp(-x^2/\sigma^2)$ with $\sigma \to 0$, where x is the variable (argument) of the posterior. It has a sharp peak, located at a tiny area, with all other region extremely flat. The problem of find this peak, or equivalently, findng its tiny area, is intrinsically intactable.

So, even for Bayesian neural network, a layer by layer abstraction along depth cannot be absent.

6 Transfer Learning

7 Why not MCMC?